Package 'CRISPRseek'

March 30, 2021

Type Package

Title Design of target-specific guide RNAs in CRISPR-Cas9, genome-editing systems

Version 1.30.1

Date 2021-01-11

Encoding UTF-8

Author
Lihua Julie Zhu, Benjamin R. Holmes, Hervé Pagès, Hui Mao, Michael Lawrence, Isana Veksler-Lublinsky, Victor Ambros, Neil Aronin and Michael Brodsky

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

Depends R (>= 3.0.1), BiocGenerics, Biostrings

Imports parallel, data.table, seqinr, S4Vectors (>= 0.9.25), IRanges, BSgenome, BiocParallel, hash, methods,reticulate,rhdf5

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

Description The package includes functions to find potential guide RNAs for the CRISPR editing system including Base Editors and the Prime Editor for input target sequences, optionally filter guide RNAs without restriction enzyme cut site, or without paired guide RNAs, genome-wide search for off-targets, score, rank, fetch flank sequence and indicate whether the target and off-targets are located in exon region or not. Potential guide RNAs are annotated with total score of the top5 and topN off-targets, detailed topN mismatch sites, restriction enzyme cut sites, and paired guide RNAs. The package also output indels and their frequencies for Cas9 targeted sites.

License GPL (>= 2)

LazyLoad yes

biocViews ImmunoOncology, GeneRegulation, SequenceMatching, CRISPR

RoxygenNote 7.1.0

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

git_branch RELEASE_3_12

git_last_commit 3b0270d

git_last_commit_date 2021-01-11

Date/Publication 2021-03-29

R topics documented:

`CRISPRseek-package` *Design of target-specific guide RNAs (gRNAs) in CRISPR-Cas9, genome-editing systems*

Description

Design of target-specific gRNAs for the CRISPR-Cas9 system by automatically finding potential gRNAs (paired/not paired), with/without restriction enzyme cut site(s) in a given sequence, searching for off targets with user defined maximum number of mismatches, calculating score of each off target based on mismatch positions in the off target and a penalty weight matrix, filtering off targets with user-defined criteria, and annotating off targets with flank sequences, whether located in exon or not. Summary report is also generated with gRNAs ranked by total topN off target score, annotated with restriction enzyme cut sites, gRNA efficacy and possible paired gRNAs. Detailed paired gRNAs information and restriction enzyme cut sites are stored in separate files in the output directory specified by the user. In total, four tab delimited files are generated in the output directory: OfftargetAnalysis.xls (off target details), Summary.xls (gRNA summary), REcutDetails.xls (restriction enzyme cut sites of each gRNA), and pairedgRNAs.xls (potential paired gRNAs).

Details

Package:	CRISPRseek
Type:	Package
Version:	1.0
Date:	2013-10-04
License:	GPL (>= 2)

Function offTargetAnalysis integrates all steps of off target analysis into one function call

Author(s)

Lihua Julie Zhu and Michael Brodsky Maintainer: julie.zhu@umassmed.edu

References

Mali P, Aach J, Stranges PB, Esvelt KM, Moosburner M, Kosuri S, Yang L, Church GM.CAS9 transcriptional activators for target specificity screening and paired nickases for cooperative genome engineering. Nat Biotechnol. 2013. 31(9):833-8 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. DNA targeting specificity of rNA-guided Cas9 nucleases. Nat Biotechnol. 2013. 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 Doench JG et al., Optimized sgRNA design to maximize activity and minimize off-target effe cts of CRISPR-Cas9. Nature Biotechnology Jan 18th 2016

See Also

offTargetAnalysis

Examples

```
    library(CRISPRseek)
    library("BSgenome.Hsapiens.UCSC.hg19")
    library(TxDb.Hsapiens.UCSC.hg19.knownGene)
    library(org.Hs.eg.db)
    outputDir <- getwd()
    inputFilePath <- system.file("extdata", "inputseq.fa", package = "CRISPRseek")
    REpatternFile <- system.file("extdata", "NEBenzymes.fa", package = "CRISPRseek")
######## Scenario 1. Target and off-target analysis for paired gRNAs with
######## one of the pairs overlap RE sites
    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly=TRUE,
        REpatternFile =REpatternFile,findPairedgRNAOnly=TRUE,
        BSgenomeName=Hsapiens, txdb=TxDb.Hsapiens.UCSC.hg19.knownGene,
        orgAnn = org.Hs.egSYMBOL,max.mismatch = 1, chromToSearch = "chrX",
        outputDir = outputDir,overwrite = TRUE)

######## Scenario 2. Target and off-target analysis for paired gRNAs with or
######## without RE sites
    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = FALSE,
        REpatternFile = REpatternFile,findPairedgRNAOnly = TRUE,
        BSgenomeName = Hsapiens, txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
        orgAnn = org.Hs.egSYMBOL,max.mismatch = 1, chromToSearch = "chrX",
        outputDir = outputDir, overwrite = TRUE)

######## Scenario 3. Target and off-target analysis for gRNAs overlap RE sites

    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = TRUE,
        REpatternFile = REpatternFile,findPairedgRNAOnly = FALSE,
        BSgenomeName = Hsapiens, txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
        orgAnn = org.Hs.egSYMBOL, max.mismatch = 1, chromToSearch = "chrX",
        outputDir = outputDir, overwrite = TRUE)
```

```
######## Scenario 4. Off-target analysis for all potential gRNAs, this will
########be the slowest among the aforementioned scenarios.

    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = FALSE,
        REpatternFile = REpatternFile,findPairedgRNAOnly = FALSE,
        BSgenomeName = Hsapiens, txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
         orgAnn = org.Hs.egSYMBOL, max.mismatch = 1, chromToSearch = "chrX",
        outputDir = outputDir,overwrite = TRUE)

######## Scenario 5. Target and off-target analysis for gRNAs input by user.
    gRNAFilePath <- system.file("extdata", "testHsap_GATA1_ex2_gRNA1.fa",
        package="CRISPRseek")
    results <- offTargetAnalysis(inputFilePath = gRNAFilePath, findgRNAs = FALSE,
        findgRNAsWithREcutOnly = FALSE, REpatternFile = REpatternFile,
        findPairedgRNAOnly = FALSE, BSgenomeName = Hsapiens,
        txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
         orgAnn = org.Hs.egSYMBOL, max.mismatch = 1, chromToSearch = "chrX",
        outputDir = outputDir, overwrite = TRUE)

####### Scenario 6. Quick gRNA finding  without target and off-target analysis
    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = TRUE,
        REpatternFile = REpatternFile,findPairedgRNAOnly = TRUE,
        chromToSearch = "", outputDir = outputDir, overwrite = TRUE)

####### Scenario 7. Quick gRNA finding  with gRNA efficacy analysis
    results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = TRUE,
        REpatternFile = REpatternFile,findPairedgRNAOnly = TRUE,
BSgenomeName = Hsapiens, annotateExon = FALSE,
        max.mismatch = 0, outputDir = outputDir, overwrite = TRUE)
```

`annotateOffTargets` *annotate off targets*

Description

annotate off targets to indicate whether it is inside an exon or intron, and the gene id if inside the gene.

Usage

```
annotateOffTargets(scores, txdb, orgAnn, ignore.strand = TRUE)
```

Arguments

`scores`	a data frame output from getOfftargetScore or filterOfftarget. It contains strand (strand of the off target, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name),gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (score of the off target), mismatch.distance2PAM (a comma separated distances of all mismatches to PAM,

	e.g., 14,11 means one mismatch is 14 bp away from PAM and the other mismatch is 11 bp away from PAM), alignment (alignment between gRNA and off target, e.g., ...G..C..... means that this off target aligns with gRNA except that G and C are mismatches),NGG (this off target contains canonical PAM or not, 1 for yes and 0 for no) mean.neighbor.distance.mismatch (mean distance between neighboring mismatches)
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#___An 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.en 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
ignore.strand	default to TRUE

Value

a data frame with off target annotation

Author(s)

Lihua Julie Zhu

References

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

See Also

offTargetAnalysis

Examples

```
library(CRISPRseek)
#library("BSgenome.Hsapiens.UCSC.hg19")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(org.Hs.eg.db)
hitsFile <-  system.file("extdata", "hits.txt", package="CRISPRseek")
hits <- read.table(hitsFile, sep = "\t", header = TRUE,
    stringsAsFactors = FALSE)
featureVectors <- buildFeatureVectorForScoring(hits)
scores <- getOfftargetScore(featureVectors)
outputDir <- getwd()
results <- annotateOffTargets(scores,
    txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
     orgAnn = org.Hs.egSYMBOL)
results
```

`buildFeatureVectorForScoring` *Build feature vectors*

Description

Build feature vectors for calculating scores of off targets

Usage

```
buildFeatureVectorForScoring(hits, gRNA.size = 20,
    canonical.PAM = "NGG",
    subPAM.position = c(22,23), PAM.size = 3, PAM.location = "3prime")
```

Arguments

`hits`	a data frame generated from searchHits, which contains IsMismatch.posX (Indicator variable indicating whether this position X is mismatch or not, 1 means yes and 0 means not, X = 1- gRNA.size) representing all positions in the guide RNA, abbreviated as gRNA),strand (strand of the off target, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name), gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (set to 100, and will be calculated in getOfftargetScore)
`gRNA.size`	gRNA size, default 20
`canonical.PAM`	Canonical PAM, default NGG for spCas9, TTTN for Cpf1
`subPAM.position`	The start and end positions of the sub PAM to fetch. Default to 22 and 23 for SP with 20bp gRNA and NGG as preferred PAM
`PAM.size`	Size of PAM, default to 3 for spCas9, 4 for Cpf1
`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

Value

A data frame with hits plus features used for calculating scores and for generating report, including IsMismatch.posX (Indicator variable indicating whether this position X is mismatch or not, 1 means yes and 0 means not, X = 1- gRNA.size) representing all positions in the gRNA), strand (strand of the off target, + for plus and - for minus strand),chrom (chromosome of the off target), chromStart (start position of the off target),chromEnd (end position of the off target),name (gRNA name), gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (score of the off target), mismatche.distance2PAM (a comma separated distances of all mismatches to PAM, e.g., 14,11 means one mismatch is 14 bp away from PAM and the other mismatch is 11 bp away from PAM), alignment (alignment between gRNA

and off target, e.g., ...G..C..... means that this off target aligns with gRNA except that G and C are mismatches), NGG (this off target contains canonical PAM or not, 1 for yes and 0 for no) mean.neighbor.distance.mismatch (mean distance between neighboring mismatches)

Author(s)

Lihua Julie Zhu

See Also

offTargetAnalysis

Examples

```
hitsFile <-  system.file("extdata", "hits.txt", package = "CRISPRseek")
hits <- read.table(hitsFile, sep= "\t", header = TRUE,
    stringsAsFactors = FALSE)
buildFeatureVectorForScoring(hits)
```

`calculategRNAEfficiency` *Calculate gRNA Efficiency*

Description

Calculate gRNA Efficiency for a given set of sequences and feature weight matrix

Usage

```
calculategRNAEfficiency(extendedSequence,
    baseBeforegRNA, featureWeightMatrix, gRNA.size = 20,
    enable.multicore = FALSE, n.cores.max = 6)
```

Arguments

`extendedSequence`	Sequences containing gRNA plus PAM plus flanking sequences. Each sequence should be long enough for building features specified in the featureWeightMatrix
`baseBeforegRNA`	Number of bases before gRNA used for calculating gRNA efficiency, default 4
`featureWeightMatrix`	a data frame with the first column containing significant features and the second column containing the weight of corresponding features. In the following example, DoenchNBT2014 weight matrix is used. Briefly, features include INTERCEPT,GC_LOW (penalty for low GC content in the gRNA sequence), GC_HIGH (penalty for high GC content in the gRNA sequence), G02 (means G at second position of the extendedSequence), GT02 (means GT di-nucleotides starting at 2nd position of the extendedSequence). To understand how is the feature weight matrix is identified, or how to use alternative feature weight matrix file, please see Doench et al., 2014 for details.
`gRNA.size`	The size of the gRNA, default 20

`enable.multicore`
: Indicate whether enable parallel processing, default FALSE. For super long sequences with lots of gRNAs, suggest set it to TRUE

`n.cores.max`
: 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

DNAStringSet consists of potential gRNAs that can be input to filtergRNAs function directly

Author(s)

Lihua Julie Zhu

References

Doench JG, Hartenian E, Graham DB, Tothova Z, Hegde M, Smith I, Sullender M, Ebert BL, Xavier RJ, Root DE. Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation. Nat Biotechnol. 2014 Sep 3. doi: 10.1038 nbt.3026 http://www.broadinstitute.org/rnai/public/analysis-tools/sgrna-design

See Also

offTargetAnalysis

Examples

```
  extendedSequence <- c("TGGATTGTATAATCAGCATGGATTTGGAAC",
"TCAACGAGGATATTCTCAGGCTTCAGGTCC",
"GTTACCTGAATTTGACCTGCTCGGAGGTAA",
"CTTGGTGTGGCTTCCTTTAAGACATGGAGC",
"CATACAGGCATTGAAGAAGAATTTAGGCCT",
"AGTACTATACATTTGGCTTAGATTTGGCGG",
"TTTTCCAGATAGCCGATCTTGGTGTGGCTT",
"AAGAAGGGAACTATTCGCTGGTGATGGAGT"
)
featureWeightMatrixFile <- system.file("extdata", "DoenchNBT2014.csv",
package = "CRISPRseek")
featureWeightMatrix <- read.csv(featureWeightMatrixFile, header=TRUE)
calculategRNAEfficiency(extendedSequence, baseBeforegRNA = 4,
featureWeightMatrix, gRNA.size = 20)
```

`compare2Sequences` *Compare 2 input sequences/sequence sets for possible guide RNAs (gRNAs)*

Description

Generate all possible guide RNAs (gRNAs) for two input sequences, or two sets of sequences and generate scores for potential off-targets in the other sequence.

Usage

```
compare2Sequences(inputFile1Path, inputFile2Path,
    inputNames=c("Seq1", "Seq2"),
    format = c("fasta", "fasta"), header=FALSE, findgRNAsWithREcutOnly = FALSE,
    searchDirection=c("both","1to2", "2to1"), BSgenomeName,
    baseEditing = FALSE, targetBase = "C", editingWindow = 4:8,
    editingWindow.offtargets = 4:8,
   REpatternFile=system.file("extdata", "NEBenzymes.fa", package = "CRISPRseek"),
   minREpatternSize = 6, findgRNAs = c(TRUE, TRUE), removegRNADetails = c(FALSE, FALSE),
    exportAllgRNAs = c("no", "all", "fasta", "genbank"), annotatePaired =  FALSE,
    overlap.gRNA.positions = c(17, 18), findPairedgRNAOnly = FALSE,
    min.gap = 0, max.gap = 20, gRNA.name.prefix = "_gR", PAM.size = 3,
    gRNA.size = 20, PAM = "NGG", PAM.pattern = "NNG$|NGN$",
    allowed.mismatch.PAM = 1, max.mismatch = 3,
    outputDir, upstream = 0, downstream = 0,
    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),
    overwrite = FALSE, baseBeforegRNA = 4,
    baseAfterPAM = 3, featureWeightMatrixFile = system.file("extdata",
       "DoenchNBT2014.csv", package = "CRISPRseek"), foldgRNAs = FALSE,
    gRNA.backbone =
"GUUUUAGAGCUAGAAAUAGCAAGUUAAAAUAAGGCUAGUCCGUUAUCAACUUGAAAAAGUGGCACCGAGUCGGUGCUUUUUU",
    temperature = 37,
    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",
   rule.set = c("Root_RuleSet1_2014", "Root_RuleSet2_2016"), mismatch.activity.file = system.file(
         "NatureBiot2016SuppTable19DoenchRoot.csv",
         package = "CRISPRseek")
    )
```

Arguments

inputFile1Path	Sequence input file 1 path that contains one of the two sequences to be searched for potential gRNAs. It can also be a DNAStringSet object with names field set. Please see examples below.

`inputFile2Path` Sequence input file 2 path that contains one of the two sequences to be searched for potential gRNAs. It can also be a DNAStringSet object with names field set. Please see examples below.

`inputNames` Name of the input sequences when inputFile1Path and inputFile2Path are DNAStringSet instead of file path

`format` Format of the input files, fasta, fastq and bed format are supported, default fasta

`header` Indicate whether the input file contains header, default FALSE, only applies to bed format

`findgRNAsWithREcutOnly` Indicate whether to find gRNAs overlap with restriction enzyme recognition pattern

`searchDirection` Indicate whether perfrom gRNA in both sequences and off-target search against each other (both) or search gRNA in input1 and off-target analysis in input2 (1to2), or vice versa (2to1)

`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

`baseEditing` Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.

`targetBase` Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.

`editingWindow` Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window.

`editingWindow.offtargets` Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

`REpatternFile` File path containing restriction enzyme cut patters

`minREpatternSize` Minimum restriction enzyme recognition pattern length required for the enzyme pattern to be searched for, default 6

`findgRNAs` Indicate whether to find gRNAs from the sequences in the input file or skip the step of finding gRNAs, default TRUE for both input sequences. Set it to FALSE if the input file contains user selected gRNAs plus PAM already.

`removegRNADetails` Indicate whether to remove the detailed gRNA information such as efficacy file and restriction enzyme cut sites, default false for both input sequences. Set it to TRUE if the input file contains the user selected gRNAs plus PAM already.

`exportAllgRNAs` Indicate whether to output all potential gRNAs to a file in fasta format, genbank format or both. Default to no.

`annotatePaired` Indicate whether to output paired information, default to FALSE

`overlap.gRNA.positions`
: The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18

`findPairedgRNAOnly`
: Choose whether to only search for paired gRNAs in such an orientation that the first one is on minus strand called reverse gRNA and the second one is on plus strand called forward gRNA. TRUE or FALSE, default FALSE

`min.gap`
: Minimum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 0

`max.gap`
: Maximum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 20

`gRNA.name.prefix`
: The prefix used when assign name to found gRNAs, default _gR, short for guided RNA.

`PAM.size`
: PAM length, default 3

`gRNA.size`
: The size of the gRNA, default 20

`PAM`
: PAM sequence after the gRNA, default NGG

`PAM.pattern`
: Regular expression of PAM, default NNG or NGN for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence

`allowed.mismatch.PAM`
: Maximum number of mismatches allowed to the PAM sequence, default to 1 for PAM.pattern NNG or NGN PAM

`max.mismatch`
: Maximum mismatch allowed to search the off targets in the other sequence, default 3

`outputDir`
: the directory where the sequence comparison results will be written to

`upstream`
: upstream offset from the bed input starts to search for gRNA and/or offtargets, default 0

`downstream`
: downstream offset from the bed input ends to search for gRNA and/or offtargets, default 0

`weights`
: 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) which is used in Hsu et al., 2013 cited in the reference section

`overwrite`
: overwrite the existing files in the output directory or not, default TRUE

`baseBeforegRNA`
: Number of bases before gRNA used for calculating gRNA efficiency, default 4 Please note, for PAM located on the 5 prime, need to specify the number of bases before the PAM sequence plus PAM size.

`baseAfterPAM`
: Number of bases after PAM used for calculating gRNA efficiency, default 3 for spCas9 Please note, for PAM located on the 5 prime, need to include the length of the gRNA plus the extended sequence on the 3 prime

`featureWeightMatrixFile`
: Feature weight matrix file used for calculating gRNA efficiency. By default DoenchNBT2014 weight matrix is used. To use alternative weight matrix file, please input a csv file with first column containing significant features and the second column containing the corresponding weights for the features. Please see Doench et al., 2014 for details.

`foldgRNAs`
: Default FALSE. If set to TRUE, summary file will contain minimum free energy of the secondary structure of gRNA with gRNA backbone from GeneRfold package provided that GeneRfold package has been installed.

gRNA.backbone gRNA backbone constant region sequence. Default to the sequence in Sp gRNA backbone.

temperature temperature in celsius. Default to 37 celsius.

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, spCas9 PAM is located on the 3 prime (3prime) while cpf1 PAM is located on the 5 prime (5prime)

rule.set Specify a rule set scoring system for calculating gRNA efficacy. Please note that Root_RuleSet2_2016 requires the following python packages with specified verion and python 2.7. 1. scikit-learn 0.16.1 2. pickle 3. pandas 4. numpy 5. scipy

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

Value

Return a data frame with all potential gRNAs from both sequences. In addition, a tab delimited file scoresFor2InputSequences.xls is also saved in the outputDir, sorted by scoreDiff descending.

name name of the gRNA

gRNAPlusPAM gRNA plus PAM sequence

targetInSeq1 target/off-target sequence including PAM in the 1st input sequence file

targetInSeq2 target/off-target sequence incuding PAM in the 2nd input sequence file

guideAlignment2Offtarget
alignment of gRNA to the other input sequence (off-target sequence)

offTargetStrand
strand of the other sequence (off-target sequence) the gRNA align to

scoreForSeq1 score for the target sequence in the 1st input sequence file

scoreForSeq2 score for the target sequence in the 1st input sequence file

mismatch.distance2PAM
distances of mismatch to PAM, e.g., 14 means the mismatch is 14 bp away from PAM

n.mismatch number of mismatches between the off-target and the gRNA

targetSeqName the name of the input sequence where the target sequence is located

scoreDiff scoreForSeq1 - scoreForSeq2

bracket.notation
folded gRNA in bracket notation

mfe.sgRNA	minimum free energy of sgRNA
mfe.diff	mfe.sgRNA-mfe.backbone
mfe.backbone	minimum free energy of the gRNA backbone by itself

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

See Also

CRISPRseek

Examples

```
    library(CRISPRseek)
    inputFile1Path <- system.file("extdata", "rs362331T.fa",
            package = "CRISPRseek")
    inputFile2Path <- system.file("extdata", "rs362331C.fa",
            package = "CRISPRseek")
    REpatternFile <- system.file("extdata", "NEBenzymes.fa",
            package = "CRISPRseek")
    seqs <- compare2Sequences(inputFile1Path, inputFile2Path,
        outputDir = getwd(),
        REpatternFile = REpatternFile, overwrite = TRUE)

    seqs2 <- compare2Sequences(inputFile1Path, inputFile2Path,
               inputNames=c("Seq1", "Seq2"),
               scoring.method = "CFDscore",
               outputDir = getwd(),
               overwrite = TRUE, baseEditing = TRUE)

    inputFile1Path <-
DNAStringSet(
"TAATATTTTAAAATCGGTGACGTGGGCCCAAAACGAGTGCAGTTCCAAAGGCACCCACCTGTGGCAG"
)
    ## when set inputFile1Path to a DNAStringSet object, it is important
    ## to call names
    names(inputFile1Path) <- "seq1"

    inputFile2Path <-
DNAStringSet(
"TAATATTTTAAAATCGGTGACGTGGGCCCAAAACGAGTGCAGTTCCAAAGGCACCCACCTGTGGCAG"
)
     ## when set inputFile2Path to a DNAStringSet object, it is important
    ## to call names

    names(inputFile2Path) <- "seq2"

    seqs <- compare2Sequences(inputFile1Path, inputFile2Path,
```

```
            inputNames=c("Seq1", "Seq2"),
            scoring.method = "CFDscore",
            outputDir = getwd(),
            overwrite = TRUE)

    seqs2 <- compare2Sequences(inputFile1Path, inputFile2Path,
                inputNames=c("Seq1", "Seq2"),
                scoring.method = "CFDscore",
                outputDir = getwd(),
                overwrite = TRUE, baseEditing = TRUE)
```

`filtergRNAs` *Filter gRNAs*

Description

Filter gRNAs containing restriction enzyme cut site

Usage

```
filtergRNAs(all.gRNAs, pairOutputFile = "",
    findgRNAsWithREcutOnly = FALSE,
    REpatternFile = system.file("extdata", "NEBenzymes.fa",
        package = "CRISPRseek"), format = "fasta",
  minREpatternSize = 4, overlap.gRNA.positions = c(17, 18),overlap.allpos = TRUE)
```

Arguments

`all.gRNAs`	gRNAs as DNAStringSet, such as the output from findgRNAs
`pairOutputFile`	File path with paired gRNAs
`findgRNAsWithREcutOnly`	Indicate whether to find gRNAs overlap with restriction enzyme recognition pattern
`REpatternFile`	File path containing restriction enzyme cut patters
`format`	Format of the REpatternFile, default as fasta
`minREpatternSize`	Minimum restriction enzyme recognition pattern length required for the enzyme pattern to be searched for, default 4
`overlap.gRNA.positions`	The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18
`overlap.allpos`	Default TRUE, meaning that only gRNAs overlap with all the positions are retained FALSE, meaning that gRNAs overlap with one or both of the positions are retained

Value

`gRNAs.withRE`	gRNAs as DNAStringSet that passed the filter criteria
`gRNAREcutDetails`	a data frame that contains a set of gRNAs annotated with restriction enzyme cut details

Author(s)

Lihua Julie Zhu

See Also

offTargetAnalysis

Examples

```
all.gRNAs <- findgRNAs(
    inputFilePath = system.file("extdata", "inputseq.fa",
    package = "CRISPRseek"),
    pairOutputFile = "testpairedgRNAs.xls",
    findPairedgRNAOnly = TRUE)

gRNAs.RE <- filtergRNAs(all.gRNAs = all.gRNAs,
    pairOutputFile = "testpairedgRNAs.xls",minREpatternSize = 6,
    REpatternFile = system.file("extdata", "NEBenzymes.fa",
    package = "CRISPRseek"), overlap.allpos = TRUE)

gRNAs  <- gRNAs.RE$gRNAs.withRE
restriction.enzyme.cut.sites <- gRNAs.RE$gRNAREcutDetails
```

---

filterOffTarget *filter off targets and generate reports.*

Description

filter off targets that meet the criteria set by users such as minimum score, topN. In addition, off target was annotated with flank sequence, gRNA cleavage efficiency and whether it is inside an exon or not if fetchSequence is set to TRUE and annotateExon is set to TRUE

Usage

```
filterOffTarget(scores, min.score = 0.01, topN = 200,
    topN.OfftargetTotalScore = 20,
    annotateExon = TRUE, txdb, orgAnn,ignore.strand = TRUE,
    outputDir, oneFilePergRNA = FALSE,
    fetchSequence = TRUE, upstream = 200, downstream = 200, BSgenomeName,
    baseBeforegRNA = 4, baseAfterPAM = 3,
    featureWeightMatrixFile = system.file("extdata", "DoenchNBT2014.csv",
package = "CRISPRseek"),
    rule.set = c("Root_RuleSet1_2014", "Root_RuleSet2_2016", "CRISPRscan"),
    calculategRNAefficacyForOfftargets = TRUE)
```

Arguments

scores	a data frame output from getOfftargetScore. It contains strand (strand of the off target, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name),gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the

off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (score of the off target), mismatch.distance2PAM (a comma separated distances of all mismatches to PAM, e.g., 14,11 means one mismatch is 14 bp away from PAM and the other mismatch is 11 bp away from PAM), alignment (alignment between gRNA and off target, e.g., ...G..C..... means that this off target aligns with gRNA except that G and C are mismatches),NGG (this off target contains canonical PAM or not, 1 for yes and 0 for no) mean.neighbor.distance.mismatc (mean distance between neighboring mismatches)

`min.score` minimum score of an off target to included in the final output, default 0.5

`topN` top N off targets to be included in the final output, default 100

`topN.OfftargetTotalScore`
top N off target used to calculate the total off target score, default 10

`annotateExon` Choose whether or not to indicate whether the off target is inside an exon or not, default TRUE

`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#___An 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.en 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

`ignore.strand` default to TRUE

`outputDir` the directory where the off target analysis and reports will be written to

`oneFilePergRNA` write to one file for each gRNA or not, default to FALSE

`fetchSequence` Fetch flank sequence of off target or not, default TRUE

`upstream` upstream offset from the off target start, default 200

`downstream` downstream offset from the off target end, default 200

`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, and BSgenome.Dmelanogaster.UCSC.dm3 for dm3

`baseBeforegRNA` Number of bases before gRNA used for calculating gRNA efficiency, default 4

`baseAfterPAM` Number of bases after PAM used for calculating gRNA efficiency, default 3

`featureWeightMatrixFile`
Feature weight matrix file used for calculating gRNA efficiency. By default DoenchNBT2014 weight matrix is used. To use alternative weight matrix file, please input a csv file with first column containing significant features and the second column containing the corresponding weights for the features. Please see Doench et al., 2014 for details.

`rule.set` Specify a rule set scoring system for calculating gRNA efficacy.

`calculategRNAefficacyForOfftargets`
Default to TRUE to output gRNA efficacy for offtargets as well as ontargets. Set it to FALSE if only need gRNA efficacy calculated for ontargets only to speed up the analysis. Please refer to https://support.bioconductor.org/p/133538/#133661 for potential use cases of offtarget efficacies.

Value

offtargets	a data frame with off target analysis results
summary	a data frame with summary of the off target analysis results

Author(s)

Lihua Julie Zhu

References

Doench JG, Hartenian E, Graham DB, Tothova Z, Hegde M, Smith I, Sullender M, Ebert BL, Xavier RJ, Root DE. Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation. Nat Biotechnol. 2014 Sep 3. doi: 10.1038 nbt.3026 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

See Also

offTargetAnalysis

Examples

```
library(CRISPRseek)
library("BSgenome.Hsapiens.UCSC.hg19")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(org.Hs.eg.db)
hitsFile <-  system.file("extdata", "hits.txt", package="CRISPRseek")
hits <- read.table(hitsFile, sep = "\t", header = TRUE,
    stringsAsFactors = FALSE)
featureVectors <- buildFeatureVectorForScoring(hits)
scores <- getOfftargetScore(featureVectors)
outputDir <- getwd()
results <- filterOffTarget(scores, BSgenomeName = Hsapiens,
    txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
     orgAnn = org.Hs.egSYMBOL, outputDir = outputDir,
    min.score = 0.1, topN = 10, topN.OfftargetTotalScore = 5)
results$offtargets
results$summary
```

---

findgRNAs *Find potential gRNAs*

Description

Find potential gRNAs for an input file containing sequences in fasta format

Usage

```
findgRNAs(inputFilePath, baseEditing = FALSE, targetBase = "C", editingWindow = 4:8,
    format = "fasta", PAM = "NGG", PAM.size = 3,
    findPairedgRNAOnly = FALSE, annotatePaired = TRUE,
    paired.orientation = c("PAMout", "PAMin"), enable.multicore = FALSE,
    n.cores.max = 6, gRNA.pattern = "", gRNA.size = 20,
    overlap.gRNA.positions = c(17,18),
    primeEditing = FALSE,
        PBS.length = 13L,
        RT.template.length = 8:28,
        RT.template.pattern = "D$",
        corrected.seq,
        targeted.seq.length.change,
        bp.after.target.end = 15L,
        target.start,
        target.end,
        primeEditingPaired.output = "pairedgRNAsForPE.xls",
    min.gap = 0, max.gap = 20,
    pairOutputFile, name.prefix = "",
    featureWeightMatrixFile = system.file("extdata",
        "DoenchNBT2014.csv", package = "CRISPRseek"), baseBeforegRNA = 4,
        baseAfterPAM = 3, calculategRNAEfficacy = FALSE, efficacyFile,
    PAM.location = "3prime",
    rule.set = c("Root_RuleSet1_2014", "Root_RuleSet2_2016", "CRISPRscan"))
```

Arguments

inputFilePath	Sequence input file path or a DNAStringSet object that contains sequences to be searched for potential gRNAs
baseEditing	Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.
targetBase	Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.
editingWindow	Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window.
format	Format of the input file, fasta and fastq are supported, default fasta
PAM	protospacer-adjacent motif (PAM) sequence after the gRNA, default NGG
PAM.size	PAM length, default 3
findPairedgRNAOnly	Choose whether to only search for paired gRNAs in such an orientation that the first one is on minus strand called reverse gRNA and the second one is on plus strand called forward gRNA. TRUE or FALSE, default FALSE
annotatePaired	Indicate whether to output paired information, default TRUE
paired.orientation	PAMin orientation means the two adjacent PAMs on the sense and antisense strands face inwards towards each other like N21GG and CCN21 whereas PAMout orientation means they face away from each other like CCN21 and N21GG

enable.multicore
: Indicate whether enable parallel processing, default FALSE. For super long sequences with lots of gRNAs, suggest set it to TRUE

n.cores.max
: 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.

gRNA.pattern
: Regular expression or IUPAC Extended Genetic Alphabet to represent gRNA pattern, default is no restriction. To specify that the gRNA must start with GG for example, then set it to ^GG. Please see help(translatePattern) for a list of IUPAC Extended Genetic Alphabet.

gRNA.size
: The size of the gRNA, default 20

overlap.gRNA.positions
: The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18

primeEditing
: Indicate whether to design gRNAs for prime editing. Default to FALSE. If true, please set PBS.length, RT.template.length, RT.template.pattern, targeted.seq.length.change, bp.after.target.end, target.start, and target.end accordingly

PBS.length
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to ouput for primer binding site.

RT.template.length
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases required for RT template, default to 8 to 18. Please increase the length if the edit is large insertion. Only gRNAs with calculated RT.template.length falling into the specified range will be in the output. It is calculated as the following. RT.template.length = target.start – cut.start + (target.end - target.start) + targeted.seq.length.change + bp.after.target.end

RT.template.pattern
: Applicable only when primeEditing is set to TRUE. It is used to specify the RT template sequence pattern, default to not ending with C according to https://doi.org/10.1038/s41586-019-1711-4

corrected.seq
: Applicable only when primeEditing is set to TRUE. It is used to specify the mutated or inserted sequences after successful editing.

targeted.seq.length.change
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of targeted sequence length change. Please set it to 0 for base changes, positive numbers for insersion, and negative number for deletion. For example, 10 means that the corrected sequence will have 10bp insertion, -10 means that the corrected sequence will have 10bp deletion, and 0 means only bases have been changed and the sequence length remains the same

bp.after.target.end
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to add after the target change end site as part of RT template. Please refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

target.start
: Applicable only when primeEditing is set to TRUE. It is used to specify the start location in the input sequence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`target.end` Applicable only when primeEditing is set to TRUE. It is used to specify the end location in the input sequnence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`primeEditingPaired.output` Applicable only when primeEditing is set to TRUE. It is used to specify the file path to save pegRNA and the second gRNA with PBS, RT.template, gRNA sequences, default pairedgRNAsForPE.xls

`min.gap` Minimum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 0

`max.gap` Maximum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 20

`pairOutputFile` The output file for writing paired gRNA information to

`name.prefix` The prefix used when assign name to found gRNAs, default gRNA, short for guided RNA.

`baseBeforegRNA` Number of bases before gRNA used for calculating gRNA efficiency, default 4 for spCas9 Please note, for PAM located on the 5 prime, need to specify the number of bases before the PAM sequence plus PAM size.

`baseAfterPAM` Number of bases after PAM used for calculating gRNA efficiency, default 3 for spCas9 Please note, for PAM located on the 5 prime, need to include the length of the gRNA plus the extended sequence on the 3 prime

`featureWeightMatrixFile` Feature weight matrix file used for calculating gRNA efficiency. By default DoenchNBT2014 weight matrix is used. To use alternative weight matrix file, please input a csv file with first column containing significant features and the second column containing the corresponding weights for the features. Please see Doench et al., 2014 for details.

`calculategRNAEfficacy` Default to FALSE, not to calculate gRNA efficacy

`efficacyFile` File path to write gRNA efficacies

`PAM.location` PAM location relative to gRNA. For example, spCas9 PAM is located on the 3 prime while cpf1 PAM is located on the 5 prime

`rule.set` Specify a rule set scoring system for calculating gRNA efficacy.

Details

If users already has a fasta file that contains a set of potential gRNAs, then users can call filergRNAs directly although the easiest way is to call the one-stop-shopping function OffTargetAnalysis with findgRNAs set to FALSE.

Value

DNAStringSet consists of potential gRNAs that can be input to filtergRNAs function directly

Note

If the input sequence file contains multiple >300 bp sequences, suggest create one input file for each sequence and run the OffTargetAnalysis separately.

Author(s)

Lihua Julie Zhu

See Also

offTargetAnalysis

Examples

```
findgRNAs(inputFilePath = system.file("extdata",
    "inputseq.fa", package = "CRISPRseek"),
    pairOutputFile = "testpairedgRNAs.xls",
    findPairedgRNAOnly = TRUE)


##### predict gRNA efficacy using CRISPRscan
featureWeightMatrixFile <- system.file("extdata", "Morenos-Mateo.csv",
        package = "CRISPRseek")

findgRNAs(inputFilePath = system.file("extdata",
    "inputseq.fa", package = "CRISPRseek"),
    pairOutputFile = "testpairedgRNAs.xls",
    findPairedgRNAOnly = FALSE,
    calculategRNAEfficacy= TRUE,
    rule.set = "CRISPRscan",
    baseBeforegRNA = 6, baseAfterPAM = 6,
    featureWeightMatrixFile = featureWeightMatrixFile,
    efficacyFile = "testCRISPRscanEfficacy.xls"
 )

 findgRNAs(inputFilePath = system.file("extdata",
    "testCRISPRscan.fa", package = "CRISPRseek"),
    pairOutputFile = "testpairedgRNAs.xls",
    findPairedgRNAOnly = FALSE,
    calculategRNAEfficacy= TRUE,
    rule.set = "CRISPRscan",
    baseBeforegRNA = 6, baseAfterPAM = 6,
    featureWeightMatrixFile = featureWeightMatrixFile,
    efficacyFile = "testCRISPRscanEfficacy.xls"
 )

findgRNAs(inputFilePath = system.file("extdata",
    "cpf1.fa", package = "CRISPRseek"),
    findPairedgRNAOnly=FALSE,
    pairOutputFile = "testpairedgRNAs-cpf1.xls",
    PAM="TTTN", PAM.location = "5prime", PAM.size = 4,
    overlap.gRNA.positions = c(19,23),
    baseBeforegRNA = 8, baseAfterPAM = 23,
    calculategRNAEfficacy= TRUE,
    featureWeightMatrixFile = system.file("extdata",
        "DoenchNBT2014.csv", package = "CRISPRseek"),
   efficacyFile = "testcpf1Efficacy.xls")

findgRNAs(inputFilePath = system.file("extdata",
         "cpf1.fa", package = "CRISPRseek"),
         findPairedgRNAOnly=FALSE,
```

```
              pairOutputFile = "testpairedgRNAs-cpf1.xls",
              PAM="TTTN", PAM.location = "5prime", PAM.size = 4,
              overlap.gRNA.positions = c(19,23),
              baseBeforegRNA = 8, baseAfterPAM = 23,
              calculategRNAEfficacy= TRUE,
              featureWeightMatrixFile = system.file("extdata",
                  "DoenchNBT2014.csv", package = "CRISPRseek"),
             efficacyFile = "testcpf1Efficacy.xls", baseEditing =  TRUE,
             editingWindow=20, targetBase = "X")

    findgRNAs(inputFilePath = system.file("extdata",
              "cpf1.fa", package = "CRISPRseek"),
              findPairedgRNAOnly=FALSE,
              pairOutputFile = "testpairedgRNAs-cpf1.xls",
              PAM="TTTN", PAM.location = "5prime", PAM.size = 4,
              overlap.gRNA.positions = c(19,23),
              baseBeforegRNA = 8, baseAfterPAM = 23,
              calculategRNAEfficacy= TRUE,
              featureWeightMatrixFile = system.file("extdata",
                  "DoenchNBT2014.csv", package = "CRISPRseek"),
             efficacyFile = "testcpf1Efficacy.xls", baseEditing =  TRUE,
             editingWindow=20, targetBase = "C")

     inputSeq <-  DNAStringSet(paste(
"CCAGTTTGTGGATCCTGCTCTGGTGTCCTCCACACCAGAATCAGGGATCGAAAACTCA",
"TCAGTCGATGCGAGTCATCTAAATTCCGATCAATTTCACACTTTAAACG", sep =""))
     gRNAs <-  findgRNAs(inputFilePath =  inputSeq,
         pairOutputFile = "testpairedgRNAs1.xls",
         PAM.size = 3L,
         gRNA.size = 20L,
         overlap.gRNA.positions = c(17L,18L),
         PBS.length = 15,
         corrected.seq = "T",
         RT.template.pattern = "D$",
         RT.template.length = 8:30,
         targeted.seq.length.change = 0,
         bp.after.target.end = 15,
         target.start = 46,
         target.end = 46,
         paired.orientation = "PAMin", min.gap = 20, max.gap = 90,
         primeEditing = TRUE, findPairedgRNAOnly = TRUE)
```

`getOfftargetScore` *Calculate score for each off target*

Description

Calculate score for each off target with given feature vectors and weights vector

Usage

```
getOfftargetScore(featureVectors,
    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))
```

Arguments

featureVectors	a data frame generated from buildFeatureVectorForScoring. It contains IsMismatch.posX (Indicator variable indicating whether this position X is mismatch or not, 1 means yes and 0 means not, X = 1- gRNA.size) representing all positions in the gRNA), strand (strand of the off target, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target), name (gRNA name),gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (score of the off target), mismatch.distance2PAM (a comma separated distances of all mismatches to PAM, e.g., 14,11 means one mismatch is 14 bp away from PAM and the other mismatch is 11 bp away from PAM), alignment (alignment between gRNA and off target, e.g., ...G..C..... means that this off target aligns with gRNA except that G and C are mismatches),NGG (this off target contains canonical PAM or not, 1 for yes and 0 for no) mean.neighbor.distance.mismatch (mean distance between neighboring mismatches)
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) which is used in Hsu et al., 2013 cited in the reference section

Details

score is calculated using the weights and algorithm by Hsu et al., 2013 cited in the reference section

Value

a data frame containing strand (strand of the match, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name), gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (score of the off target), mismatch.distance2PAM (a comma separated distances of all mismatches to PAM, e.g., 14,11 means one mismatch is 14 bp away from PAM and the other mismatch is 11 bp away from PAM), alignment (alignment between gRNA and off target, e.g., ...G..C..... means that this off target aligns with gRNA except that G and C are mismatches), NGG (this off target contains canonical PAM or not, 1 for yes and 0 for no) mean.neighbor.distance.mismatch (mean distance between neighboring mismatches)

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

See Also

offTargetAnalysis

Examples

```
hitsFile <-  system.file("extdata", "hits.txt",
    package = "CRISPRseek")
hits <- read.table(hitsFile, sep = "\t", header = TRUE,
    stringsAsFactors = FALSE)
featureVectors <- buildFeatureVectorForScoring(hits)
getOfftargetScore(featureVectors)
```

---

isPatternUnique *Output whether the input patterns occurs only once in the sequence*

Description

Input a sequence and a list of patterns and determine if the patterns occurs only once in the sequence. Used for determining whether a RE site in gRNA also occurs in the flanking region.

Usage

```
isPatternUnique(seq, patterns)
```

Arguments

seq	flanking sequence of a gRNA
patterns	patterns as DNAStringSet, such as a list of RE sites

Value

returns a character vectors containing the uniqueness of each pattern/RE site

Author(s)

Lihua Julie Zhu

Examples

```
seq <- "TGGATTGTATAATCAGCATGGATTTGGAAC"
patterns <- DNAStringSet(c("TGG", "TGGA", "TGGATA", "TTGGAAC", ""))
isPatternUnique(seq, patterns)
isPatternUnique(seq)
isPatternUnique(patterns)
```

offTargetAnalysis *Design of target-specific guide RNAs for CRISPR-Cas9 system in one function*

Description

Design of target-specific guide RNAs (gRNAs) and predict relative indel fequencies for CRISPR-Cas9 system by automatically calling findgRNAs, filtergRNAs, searchHits, buildFeatureVectorForScoring, getOfftargetScore, filterOfftarget, calculating gRNA cleavage efficiency, and predict gRNA efficacy, indels and their frequencies.

Usage

```
offTargetAnalysis(inputFilePath, format = "fasta", header = FALSE,
    gRNAoutputName, findgRNAs = TRUE,
    exportAllgRNAs = c("all", "fasta", "genbank", "no"),
    findgRNAsWithREcutOnly = FALSE,
    REpatternFile = system.file("extdata", "NEBenzymes.fa",
        package = "CRISPRseek"), minREpatternSize = 4,
    overlap.gRNA.positions = c(17, 18), findPairedgRNAOnly = FALSE,
    annotatePaired = TRUE, paired.orientation = c("PAMout","PAMin"),
    enable.multicore = FALSE, n.cores.max = 6,
    min.gap = 0, max.gap = 20, gRNA.name.prefix = "", PAM.size = 3,
    gRNA.size = 20,  PAM = "NGG",  BSgenomeName, chromToSearch = "all",
    chromToExclude = c("chr17_ctg5_hap1","chr4_ctg9_hap1", "chr6_apd_hap1",
"chr6_cox_hap2", "chr6_dbb_hap3", "chr6_mann_hap4", "chr6_mcf_hap5","chr6_qbl_hap6",
"chr6_ssto_hap7"),
    max.mismatch = 3, PAM.pattern = "NNG$|NGN$", allowed.mismatch.PAM = 1,
    gRNA.pattern = "",
    baseEditing = FALSE, targetBase = "C", editingWindow = 4:8,
    editingWindow.offtargets = 4:8,
    primeEditing = FALSE,
        PBS.length = 13L,
        RT.template.length = 8:28,
        RT.template.pattern = "D$",
        corrected.seq,
        targeted.seq.length.change,
        bp.after.target.end = 15L,
        target.start,
        target.end,
    primeEditingPaired.output = "pairedgRNAsForPE.xls",
    min.score = 0, topN = 1000,
    topN.OfftargetTotalScore = 10, annotateExon = TRUE,
    txdb, orgAnn, ignore.strand = TRUE,
    outputDir, fetchSequence = TRUE, upstream = 200, downstream = 200,
    upstream.search = 0, downstream.search = 0,
    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),
    baseBeforegRNA = 4, baseAfterPAM = 3,
    featureWeightMatrixFile = system.file("extdata", "DoenchNBT2014.csv",
package = "CRISPRseek"), useScore = TRUE, useEfficacyFromInputSeq = FALSE,
```

```
    outputUniqueREs = TRUE, foldgRNAs = FALSE,
    gRNA.backbone =
"GUUUUAGAGCUAGAAAUAGCAAGUUAAAAUAAGGCUAGUCCGUUAUCAACUUGAAAAAGUGGCACCGAGUCGGUGCUUUUUU",
    temperature = 37,
    overwrite = FALSE,
    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",
     rule.set = c("Root_RuleSet1_2014", "Root_RuleSet2_2016", "CRISPRscan"),
     calculategRNAefficacyForOfftargets = TRUE,
     mismatch.activity.file = system.file("extdata",
         "NatureBiot2016SuppTable19DoenchRoot.csv",
         package = "CRISPRseek"),
     predIndelFreq = FALSE,
     predictIndelFreq.onTargetOnly = TRUE,
     method.indelFreq = "Lindel",
     baseBeforegRNA.indelFreq = 13,
     baseAfterPAM.indelFreq = 24
    )
```

Arguments

inputFilePath	Sequence input file path or a DNAStringSet object that contains sequences to be searched for potential gRNAs
format	Format of the input file, fasta, fastq and bed are supported, default fasta
header	Indicate whether the input file contains header, default FALSE, only applies to bed format
gRNAoutputName	Specify the name of the gRNA outupt file when inputFilePath is DNAStringSet object instead of file path
findgRNAs	Indicate whether to find gRNAs from the sequences in the input file or skip the step of finding gRNAs, default TRUE. Set it to FALSE if the input file contains user selected gRNAs plus PAM already.
exportAllgRNAs	Indicate whether to output all potential gRNAs to a file in fasta format, genbank format or both. Default to both.

`findgRNAsWithREcutOnly`
: Indicate whether to find gRNAs overlap with restriction enzyme recognition pattern

`REpatternFile`
: File path containing restriction enzyme cut patterns

`minREpatternSize`
: Minimum restriction enzyme recognition pattern length required for the enzyme pattern to be searched for, default 4

`overlap.gRNA.positions`
: The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18

`findPairedgRNAOnly`
: Choose whether to only search for paired gRNAs in such an orientation that the first one is on minus strand called reverse gRNA and the second one is on plus strand called forward gRNA. TRUE or FALSE, default FALSE

`annotatePaired`
: Indicate whether to output paired information, default TRUE

`paired.orientation`
: PAMin orientation means the two adjacent PAMs on the sense and antisense strands face inwards towards each other like N21GG and CCN21 whereas PAMout orientation means they face away from each other like CCN21 and N21GG

`min.gap`
: Minimum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 0

`enable.multicore`
: Indicate whether enable parallel processing, default FALSE. For super long sequences with lots of gRNAs, suggest set it to TRUE

`n.cores.max`
: 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.

`max.gap`
: Maximum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 20

`gRNA.name.prefix`
: The prefix used when assign name to found gRNAs, default gRNA, short for guided RNA.

`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

`chromToSearch`
: Specify the chromosome to search, default to all, meaning search all chromosomes. For example, chrX indicates searching for matching in chromosome X only

`chromToExclude`
: Specify the chromosome not to search. If specified as "", meaning to search chromosomes specified by chromToSearch. By default, to exclude haplotype blocks from offtarget search in hg19, i.e., chromToExclude = c("chr17_ctg5_hap1","chr4_ctg9_hap1" "chr6_apd_hap1", "chr6_cox_hap2", "chr6_dbb_hap3", "chr6_mann_hap4", "chr6_mcf_hap5","chr6_ "chr6_ssto_hap7")

`max.mismatch` Maximum mismatch allowed in off target search, default 3. Warning: will be considerably slower if set > 3

`PAM.pattern` Regular expression of protospacer-adjacent motif (PAM), default NNG$|NGN$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence

`allowed.mismatch.PAM`
Maximum number of mismatches allowed in the PAM sequence for offtarget search, default to 1 to allow NGN and NNG PAM pattern for offtarget identification.

`gRNA.pattern` Regular expression or IUPAC Extended Genetic Alphabet to represent gRNA pattern, default is no restriction. To specify that the gRNA must start with GG for example, then set it to ^GG. Please see help(translatePattern) for a list of IUPAC Extended Genetic Alphabet.

`baseEditing` Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.

`targetBase` Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.

`editingWindow` Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window.

`editingWindow.offtargets`
Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 (1 means the most distal site from the 3' PAM, the most proximla site from the 5' PAM), which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

`primeEditing` Indicate whether to design gRNAs for prime editing. Default to FALSE. If true, please set PBS.length, RT.template.length, RT.template.pattern, targeted.seq.length.change, bp.after.target.end, target.start, and target.end accordingly

`PBS.length` Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to ouput for primer binding site.

`RT.template.length`
Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases required for RT template, default to 8 to 18. Please increase the length if the edit is large insertion. Only gRNAs with calculated RT.template.length falling into the specified range will be in the output. It is calculated as the following. RT.template.length = target.start – cut.start + (target.end - target.start) + targeted.seq.length.change + bp.after.target.end

`RT.template.pattern`
Applicable only when primeEditing is set to TRUE. It is used to specify the RT template sequence pattern, default to not ending with C according to https://doi.org/10.1038/s41586-019-1711-4

`corrected.seq` Applicable only when primeEditing is set to TRUE. It is used to specify the mutated or inserted sequences after successful editing.

`targeted.seq.length.change`
Applicable only when primeEditing is set to TRUE. It is used to specify the number of targeted sequence length change. Please set it to 0 for base changes,

positive numbers for insersion, and negative number for deletion. For example, 10 means that the corrected sequence will have 10bp insertion, -10 means that the corrected sequence will have 10bp deletion, and 0 means only bases have been changed and the sequence length remains the same

`bp.after.target.end`
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to add after the target change end site as part of RT template. Please refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`target.start`
: Applicable only when primeEditing is set to TRUE. It is used to specify the start location in the input sequence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`target.end`
: Applicable only when primeEditing is set to TRUE. It is used to specify the end location in the input sequnence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`primeEditingPaired.output`
: Applicable only when primeEditing is set to TRUE. It is used to specify the file path to save pegRNA and the second gRNA with PBS, RT.template, gRNA sequences, default pairedgRNAsForPE.xls

`min.score`
: minimum score of an off target to included in the final output, default 0

`topN`
: top N off targets to be included in the final output, default 1000

`topN.OfftargetTotalScore`
: top N off target used to calculate the total off target score, default 10

`annotateExon`
: Choose whether or not to indicate whether the off target is inside an exon or not, default TRUE

`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#___An 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.en 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

`ignore.strand`
: default to TRUE when annotating to gene

`outputDir`
: the directory where the off target analysis and reports will be written to

`fetchSequence`
: Fetch flank sequence of off target or not, default TRUE

`upstream`
: upstream offset from the off target start, default 200

`downstream`
: downstream offset from the off target end, default 200

`upstream.search`
: upstream offset from the bed input starts to search for gRNAs, default 0

`downstream.search`
: downstream offset from the bed input ends to search for gRNAs, default 0

`weights` Applicable only when scoring.method is set to Hsu-Zhang 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) which is used in Hsu et al., 2013 cited in the reference section

`baseBeforegRNA` Number of bases before gRNA used for calculating gRNA efficiency, default 4 Please note, for PAM located on the 5 prime, need to specify the number of bases before the PAM sequence plus PAM size.

`baseAfterPAM` Number of bases after PAM used for calculating gRNA efficiency, default 3 for spCas9 Please note, for PAM located on the 5 prime, need to include the length of the gRNA plus the extended sequence on the 3 prime

`featureWeightMatrixFile`
Feature weight matrix file used for calculating gRNA efficiency. By default DoenchNBT2014 weight matrix is used. To use alternative weight matrix file, please input a csv file with first column containing significant features and the second column containing the corresponding weights for the features. Please see Doench et al., 2014 for details.

`useScore` Default TRUE, display in gray scale with the darkness indicating the gRNA efficacy. The taller bar shows the Cas9 cutting site. If set to False, efficacy will not show. Instead, gRNAs in plus strand will be colored red and gRNAs in negative strand will be colored green.

`useEfficacyFromInputSeq`
Default FALSE. If set to TRUE, summary file will contain gRNA efficacy calculated from input sequences instead of from off-target analysis. Set it to TRUE if the input sequence is from a different species than the one used for off-target analysis.

`outputUniqueREs`
Default TRUE. If set to TRUE, summary file will contain REs unique to the cleavage site within 100 or 200 bases surrounding the gRNA sequence.

`foldgRNAs` Default FALSE. If set to TRUE, summary file will contain minimum free energy of the secondary structure of gRNA with gRNA backbone from GeneRfold package provided that GeneRfold package has been installed.

`gRNA.backbone` gRNA backbone constant region sequence. Default to the sequence in Sp gRNA backbone.

`temperature` temperature in celsius. Default to 37 celsius.

`overwrite` overwrite the existing files in the output directory or not, default FALSE

`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 spCas9 with 20bp gRNA and NGG as preferred PAM. For Cpf1, it could be c(1,2).

`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

`rule.set` Specify a rule set scoring system for calculating gRNA efficacy. Please note that Root_RuleSet2_2016 requires the following python packages with specified verion and python 2.7. 1. scikit-learn 0.16.1 2. pickle 3. pandas 4. numpy 5. scipy

`calculategRNAefficacyForOfftargets`
Default to TRUE to output gRNA efficacy for offtargets as well as ontargets. Set it to FALSE if only need gRNA efficacy calculated for ontargets only to speed up the analysis. Please refer to https://support.bioconductor.org/p/133538/#133661 for potential use cases of offtarget efficacies.

`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

`predIndelFreq` Default to FALSE. Set it to TRUE to output the predicted indels and their frequencies.

`predictIndelFreq.onTargetOnly`
Default to TRUE, indicating that indels and their frequencies will be predicted for ontargets only. Usually, researchers are only interested in predicting the editing outcome for the ontargets since any editing in the offtargets are unwanted. Set it to FALSE if you are interested in predicting indels and their frequencies for offtargets. It will take longer time to run if you set it to FALSE.

`method.indelFreq`
Currently only Lindel method has been implemented. Please let us know if you think additional methods should be made available. Lindel is compatible with both Python2.7 and Python3.5 or higher. Please type help(predictRelativeFreqIndels) to get more details.

`baseBeforegRNA.indelFreq`
Default to 13 for Lindel method.

`baseAfterPAM.indelFreq`
Default to 24 for Lindel method.

Value

Four tab delimited files are generated in the output directory: OfftargetAnalysis.xls (detailed information of off targets), Summary.xls (summary of the gRNAs), REcutDetails.xls (restriction enzyme cut sites of each gRNA), and pairedgRNAs.xls (potential paired gRNAs)

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

Doench JG, Hartenian E, Graham DB, Tothova Z, Hegde M, Smith I, Sullender M, Ebert BL, Xavier RJ, Root DE. Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation. Nat Biotechnol. 2014 Sep 3. doi: 10.1038 nbt.3026

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

Moreno-Mateos, M., Vejnar, C., Beaudoin, J. et al. CRISPRscan: designing highly efficient sgRNAs for CRISPR-Cas9 targeting in vivo. Nat Methods 12, 982–988 (2015) doi:10.1038/nmeth.3543

Doench JG et al., Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9. Nature Biotechnology Jan 18th 2016

Anzalone et al., Search-and-replace genome editing without double-strand breaks or donor DNA. Nature October 2019 https://www.nature.com/articles/s41586-019-1711-4

Wei Chen, Aaron McKenna, Jacob Schreiber et al., Massively parallel profiling and predictive modeling of the outcomes of CRISPR/Cas9-mediated double-strand break repair, Nucleic Acids Research, Volume 47, Issue 15, 05 September 2019, Pages 7989–8003, https://doi.org/10.1093/nar/gkz487

See Also

CRISPRseek

Examples

```
library(CRISPRseek)
library("BSgenome.Hsapiens.UCSC.hg19")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(org.Hs.eg.db)
outputDir <- getwd()
inputFilePath <- system.file("extdata", "inputseq.fa",
            package = "CRISPRseek")
REpatternFile <- system.file("extdata", "NEBenzymes.fa",
            package = "CRISPRseek")
results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = TRUE,
            REpatternFile = REpatternFile, findPairedgRNAOnly = FALSE,
            annotatePaired = FALSE,
            BSgenomeName = Hsapiens, chromToSearch = "chrX",
            txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
    orgAnn = org.Hs.egSYMBOL, max.mismatch = 1,
            outputDir = outputDir, overwrite = TRUE)

       #### predict indels and their frequecies for target sites

       if (interactive())
       {
          results <- offTargetAnalysis(inputFilePath,findgRNAsWithREcutOnly = TRUE,
            findPairedgRNAOnly = FALSE,
            annotatePaired = FALSE,
            BSgenomeName = Hsapiens, chromToSearch = "chrX",
            txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
    orgAnn = org.Hs.egSYMBOL, max.mismatch = 1,
            outputDir = outputDir, overwrite = TRUE,
            predIndelFreq=TRUE, predictIndelFreq.onTargetOnly= TRUE)

          names(results$indelFreq)
          head(results$indelFreq[[1]])
      ### save the indel frequences to tab delimited files, one file for each target/offtarget site.
      mapply(write.table, results$indelFreq, file=paste0(names(results$indelFreq), '.xls'), sep = "\t", row.
       }
       #### predict gRNA efficacy using CRISPRscan
```

```
featureWeightMatrixFile <- system.file("extdata", "Morenos-Mateo.csv",
     package = "CRISPRseek")

results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly = TRUE,
     REpatternFile = REpatternFile, findPairedgRNAOnly = FALSE,
     annotatePaired = FALSE,
     BSgenomeName = Hsapiens, chromToSearch = "chrX",
     txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
     orgAnn = org.Hs.egSYMBOL, max.mismatch = 1,
     rule.set = "CRISPRscan",
     baseBeforegRNA = 6, baseAfterPAM = 6,
     featureWeightMatrixFile = featureWeightMatrixFile,
     outputDir = outputDir, overwrite = TRUE)

######## PAM is on the 5 prime side
results <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly =  FALSE,
     REpatternFile = REpatternFile, findPairedgRNAOnly = FALSE,
     annotatePaired = FALSE,
     BSgenomeName = Hsapiens, chromToSearch = "chrX",
     txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
     orgAnn = org.Hs.egSYMBOL, max.mismatch = 4,
     outputDir = outputDir, overwrite = TRUE, PAM.location = "5prime",
     PAM = "TGT", PAM.pattern = "^T[A|G]N", allowed.mismatch.PAM = 2,
     subPAM.position = c(1,2))

 results1 <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly =  FALSE,
          REpatternFile = REpatternFile, findPairedgRNAOnly = FALSE,
          annotatePaired = FALSE,
          BSgenomeName = Hsapiens, chromToSearch = "chrX",
          txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
          orgAnn = org.Hs.egSYMBOL, max.mismatch = 4,
          outputDir = outputDir, overwrite = TRUE, PAM.location = "5prime",
          PAM = "TGT", PAM.pattern = "^T[A|G]N", allowed.mismatch.PAM = 2,
        subPAM.position = c(1,2), baseEditing = TRUE, editingWindow =20, targetBase = "G")

results.testBE <- offTargetAnalysis(inputFilePath, findgRNAsWithREcutOnly =  FALSE,
          REpatternFile = REpatternFile, findPairedgRNAOnly = FALSE,
          annotatePaired = FALSE,
          BSgenomeName = Hsapiens, chromToSearch = "chrX",
          txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
          orgAnn = org.Hs.egSYMBOL, max.mismatch = 4,
          outputDir = outputDir, overwrite = TRUE, PAM.location = "5prime",
          PAM = "TGT", PAM.pattern = "^T[A|G]N", allowed.mismatch.PAM = 2,
          subPAM.position = c(1,2), baseEditing = TRUE,
          editingWindow = 10:20, targetBase = "A")

 inputFilePath <- DNAStringSet(paste(
"CCAGTTTGTGGATCCTGCTCTGGTGTCCTCCACACCAGAATCAGGGATCGAAAA",
"CTCATCAGTCGATGCGAGTCATCTAAATTCCGATCAATTTCACACTTTAAACG", sep =""))
 names(inputFilePath) <- "testPE"
 results3 <- offTargetAnalysis(inputFilePath,
     gRNAoutputName = "testPEgRNAs",
     BSgenomeName = Hsapiens, chromToSearch = "chrX",
     txdb = TxDb.Hsapiens.UCSC.hg19.knownGene,
     orgAnn = org.Hs.egSYMBOL, max.mismatch = 1,
     outputDir = outputDir, overwrite = TRUE,
     PAM.size = 3L,
```

```
    gRNA.size = 20L,
    overlap.gRNA.positions = c(17L,18L),
    PBS.length = 15,
    corrected.seq = "T",
    RT.template.pattern = "D$",
    RT.template.length = 8:30,
    targeted.seq.length.change = 0,
    bp.after.target.end = 15,
    target.start = 20,
    target.end = 20,
    paired.orientation = "PAMin", min.gap = 20, max.gap = 90,
    primeEditing = TRUE, findPairedgRNAOnly = TRUE)
```

---

offTargetAnalysisWithoutBSgenome

Design of target-specific guide RNAs for CRISPR-Cas9 system in one function without BSgenome

---

Description

Design of target-specific guide RNAs (gRNAs) for CRISPR-Cas9 system without BSgenome by automatically calling findgRNAs, filtergRNAs, searchHits, buildFeatureVectorForScoring, getOfftargetScore, filterOfftarget, calculating gRNA cleavage efficiency and generate reports.

Usage

```
offTargetAnalysisWithoutBSgenome(inputFilePath, format = "fasta", header = FALSE,
    gRNAoutputName, findgRNAs = TRUE,
    exportAllgRNAs = c("all", "fasta", "genbank", "no"),
    findgRNAsWithREcutOnly = FALSE,
    REpatternFile = system.file("extdata", "NEBenzymes.fa",
        package = "CRISPRseek"), minREpatternSize = 4,
    overlap.gRNA.positions = c(17, 18), findPairedgRNAOnly = FALSE,
    annotatePaired = TRUE, paired.orientation = c("PAMout","PAMin"),
    enable.multicore = FALSE, n.cores.max = 6,
    min.gap = 0, max.gap = 20, gRNA.name.prefix = "", PAM.size = 3,
    gRNA.size = 20,  PAM = "NGG",  BSgenomeName, chromToSearch = "all",
    chromToExclude = c("chr17_ctg5_hap1","chr4_ctg9_hap1", "chr6_apd_hap1",
"chr6_cox_hap2", "chr6_dbb_hap3", "chr6_mann_hap4", "chr6_mcf_hap5","chr6_qbl_hap6",
"chr6_ssto_hap7"),
    max.mismatch = 3, PAM.pattern = "NNG$|NGN$", allowed.mismatch.PAM = 1,
    gRNA.pattern = "",
    baseEditing = FALSE, targetBase = "C", editingWindow = 4:8,
    editingWindow.offtargets = 4:8,
    primeEditing = FALSE,
        PBS.length = 13L,
        RT.template.length = 8:28,
        RT.template.pattern = "D$",
        corrected.seq,
        targeted.seq.length.change,
        bp.after.target.end = 15L,
        target.start,
```

```
        target.end,
      primeEditingPaired.output = "pairedgRNAsForPE.xls",
      min.score = 0, topN = 1000,
      topN.OfftargetTotalScore = 10, annotateExon = TRUE,
      txdb, orgAnn, ignore.strand = TRUE,
      outputDir, fetchSequence = TRUE, upstream = 200, downstream = 200,
      upstream.search = 0, downstream.search = 0,
      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),
      baseBeforegRNA = 4, baseAfterPAM = 3,
      featureWeightMatrixFile = system.file("extdata", "DoenchNBT2014.csv",
  package = "CRISPRseek"), useScore = TRUE, useEfficacyFromInputSeq = FALSE,
      outputUniqueREs = TRUE, foldgRNAs = FALSE,
      gRNA.backbone =
  "GUUUUAGAGCUAGAAAUAGCAAGUUAAAAUAAGGCUAGUCCGUUAUCAACUUGAAAAAGUGGCACCGAGUCGGUGCUUUUUU",
      temperature = 37,
      overwrite = FALSE,
      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",
       rule.set = c("Root_RuleSet1_2014", "Root_RuleSet2_2016", "CRISPRscan"),
       mismatch.activity.file = system.file("extdata",
           "NatureBiot2016SuppTable19DoenchRoot.csv",
           package = "CRISPRseek"),
       useBSgenome = FALSE, genomeSeqFile,
       predIndelFreq = FALSE,
       predictIndelFreq.onTargetOnly = TRUE,
       method.indelFreq = "Lindel",
       baseBeforegRNA.indelFreq = 13,
       baseAfterPAM.indelFreq = 24
      )
```

Arguments

inputFilePath	Sequence input file path or a DNAStringSet object that contains sequences to be searched for potential gRNAs

format Format of the input file, fasta, fastq and bed are supported, default fasta

header Indicate whether the input file contains header, default FALSE, only applies to bed format

gRNAoutputName Specify the name of the gRNA outupt file when inputFilePath is DNAStringSet object instead of file path

findgRNAs Indicate whether to find gRNAs from the sequences in the input file or skip the step of finding gRNAs, default TRUE. Set it to FALSE if the input file contains user selected gRNAs plus PAM already.

exportAllgRNAs Indicate whether to output all potential gRNAs to a file in fasta format, genbank format or both. Default to both.

findgRNAsWithREcutOnly
Indicate whether to find gRNAs overlap with restriction enzyme recognition pattern

REpatternFile File path containing restriction enzyme cut patterns

minREpatternSize
Minimum restriction enzyme recognition pattern length required for the enzyme pattern to be searched for, default 4

overlap.gRNA.positions
The required overlap positions of gRNA and restriction enzyme cut site, default 17 and 18

findPairedgRNAOnly
Choose whether to only search for paired gRNAs in such an orientation that the first one is on minus strand called reverse gRNA and the second one is on plus strand called forward gRNA. TRUE or FALSE, default FALSE

annotatePaired Indicate whether to output paired information, default TRUE

paired.orientation
PAMin orientation means the two adjacent PAMs on the sense and antisense strands face inwards towards each other like N21GG and CCN21 whereas PAMout orientation means they face away from each other like CCN21 and N21GG

min.gap Minimum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 0

enable.multicore
Indicate whether enable parallel processing, default FALSE. For super long sequences with lots of gRNAs, suggest set it to TRUE

n.cores.max 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.

max.gap Maximum distance between two oppositely oriented gRNAs to be valid paired gRNAs. Default 20

gRNA.name.prefix
The prefix used when assign name to found gRNAs, default gRNA, short for guided RNA.

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

`chromToSearch` Specify the chromosome to search, default to all, meaning search all chromosomes. For example, chrX indicates searching for matching in chromosome X only

`chromToExclude` Specify the chromosome not to search. If specified as "", meaning to search chromosomes specified by chromToSearch. By default, to exclude haplotype blocks from offtarget search in hg19, i.e., chromToExclude = c("chr17_ctg5_hap1","chr4_ctg9_hap1" "chr6_apd_hap1", "chr6_cox_hap2", "chr6_dbb_hap3", "chr6_mann_hap4", "chr6_mcf_hap5","chr6_ "chr6_ssto_hap7")

`max.mismatch` Maximum mismatch allowed in off target search, default 3. Warning: will be considerably slower if set > 3

`PAM.pattern` Regular expression of protospacer-adjacent motif (PAM), default NNG$|NGN$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence

`allowed.mismatch.PAM` Maximum number of mismatches allowed in the PAM sequence for offtarget search, default to 1 for NNG or NGN PAM pattern for offtarget finding.

`gRNA.pattern` Regular expression or IUPAC Extended Genetic Alphabet to represent gRNA pattern, default is no restriction. To specify that the gRNA must start with GG for example, then set it to ^GG. Please see help(translatePattern) for a list of IUPAC Extended Genetic Alphabet.

`baseEditing` Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.

`targetBase` Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.

`editingWindow` Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window.

`editingWindow.offtargets` Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 (1 means the most distal site from the 3' PAM, the most proximla site from the 5' PAM), which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

`primeEditing` Indicate whether to design gRNAs for prime editing. Default to FALSE. If true, please set PBS.length, RT.template.length, RT.template.pattern, targeted.seq.length.change, bp.after.target.end, target.start, and target.end accordingly

`PBS.length` Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to ouput for primer binding site.

`RT.template.length` Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases required for RT template, default to 8 to 18. Please increase the length if the edit is large insertion. Only gRNAs with calculated RT.template.length falling into the specified range will be in the output. It is calculated as the following. RT.template.length = target.start – cut.start + (target.end - target.start) + targeted.seq.length.change + bp.after.target.end

`RT.template.pattern`
: Applicable only when primeEditing is set to TRUE. It is used to specify the RT template sequence pattern, default to not ending with C according to https://doi.org/10.1038/s41586-019-1711-4

`corrected.seq`
: Applicable only when primeEditing is set to TRUE. It is used to specify the mutated or inserted sequences after successful editing.

`targeted.seq.length.change`
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of targeted sequence length change. Please set it to 0 for base changes, positive numbers for insersion, and negative number for deletion. For example, 10 means that the corrected sequence will have 10bp insertion, -10 means that the corrected sequence will have 10bp deletion, and 0 means only bases have been changed and the sequence length remains the same

`bp.after.target.end`
: Applicable only when primeEditing is set to TRUE. It is used to specify the number of bases to add after the target change end site as part of RT template. Please refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`target.start`
: Applicable only when primeEditing is set to TRUE. It is used to specify the start location in the input sequence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`target.end`
: Applicable only when primeEditing is set to TRUE. It is used to specify the end location in the input sequnence to make changes, which will be used to obtain the RT template sequence. Please also refer to RT.template.length for how this parameter influences the RT.template.length calculation which is used as a filtering criteria in pregRNA selection.

`primeEditingPaired.output`
: Applicable only when primeEditing is set to TRUE. It is used to specify the file path to save pegRNA and the second gRNA with PBS, RT.template, gRNA sequences, default pairedgRNAsForPE.xls

`min.score`
: minimum score of an off target to included in the final output, default 0

`topN`
: top N off targets to be included in the final output, default 1000

`topN.OfftargetTotalScore`
: top N off target used to calculate the total off target score, default 10

`annotateExon`
: Choose whether or not to indicate whether the off target is inside an exon or not, default TRUE

`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#___An 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.en 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

`ignore.strand`
: default to TRUE when annotating to gene

`outputDir`
: the directory where the off target analysis and reports will be written to

`fetchSequence` Fetch flank sequence of off target or not, default TRUE

`upstream` upstream offset from the off target start, default 200

`downstream` downstream offset from the off target end, default 200

`upstream.search`
upstream offset from the bed input starts to search for gRNAs, default 0

`downstream.search`
downstream offset from the bed input ends to search for gRNAs, default 0

`weights` Applicable only when scoring.method is set to Hsu-Zhang 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) which is used in Hsu et al., 2013 cited in the reference section

`baseBeforegRNA` Number of bases before gRNA used for calculating gRNA efficiency, default 4 Please note, for PAM located on the 5 prime, need to specify the number of bases before the PAM sequence plus PAM size.

`baseAfterPAM` Number of bases after PAM used for calculating gRNA efficiency, default 3 for spCas9 Please note, for PAM located on the 5 prime, need to include the length of the gRNA plus the extended sequence on the 3 prime

`featureWeightMatrixFile`
Feature weight matrix file used for calculating gRNA efficiency. By default DoenchNBT2014 weight matrix is used. To use alternative weight matrix file, please input a csv file with first column containing significant features and the second column containing the corresponding weights for the features. Please see Doench et al., 2014 for details.

`useScore` Default TRUE, display in gray scale with the darkness indicating the gRNA efficacy. The taller bar shows the Cas9 cutting site. If set to False, efficacy will not show. Instead, gRNAs in plus strand will be colored red and gRNAs in negative strand will be colored green.

`useEfficacyFromInputSeq`
Default FALSE. If set to TRUE, summary file will contain gRNA efficacy calculated from input sequences instead of from off-target analysis. Set it to TRUE if the input sequence is from a different species than the one used for off-target analysis.

`outputUniqueREs`
Default TRUE. If set to TRUE, summary file will contain REs unique to the cleavage site within 100 or 200 bases surrounding the gRNA sequence.

`foldgRNAs` Default FALSE. If set to TRUE, summary file will contain minimum free energy of the secondary structure of gRNA with gRNA backbone from GeneRfold package provided that GeneRfold package has been installed.

`gRNA.backbone` gRNA backbone constant region sequence. Default to the sequence in Sp gRNA backbone.

`temperature` temperature in celsius. Default to 37 celsius.

`overwrite` overwrite the existing files in the output directory or not, default FALSE

`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 spCas9 with 20bp gRNA and NGG as preferred PAM. For Cpf1, it could be c(1,2).

`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

`rule.set`
: Specify a rule set scoring system for calculating gRNA efficacy. Please note that Root_RuleSet2_2016 requires the following python packages with specified verion and python 2.7. 1. scikit-learn 0.16.1 2. pickle 3. pandas 4. numpy 5. scipy

`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

`useBSgenome`
: Specify whether BSgenome is avaialbe for searching for gRNA and offtargets, default to FALSE. If set it to TRUE, the results should be the same as when using offtargetAnalysis function.

`genomeSeqFile`
: Specify the genome sequence file in fasta format. It is only applicable and required when useBSgenome is set to FALSE.

`predIndelFreq`
: Default to FALSE. Set it to TRUE to output the predicted indels and their frequencies.

`predictIndelFreq.onTargetOnly`
: Default to TRUE, indicating that indels and their frequencies will be predicted for ontargets only. Usually, researchers are only interested in predicting the editing outcome for the ontargets since any editing in the offtargets are unwanted. Set it to FALSE if you are interested in predicting indels and their frequencies for offtargets. It will take longer time to run if you set it to FALSE.

`method.indelFreq`
: Currently only Lindel method has been implemented. Please let us know if you think additional methods should be made available. Lindel is compatible with both Python2.7 and Python3.5 or higher. Please type help(predictRelativeFreqIndels) to get more details.

`baseBeforegRNA.indelFreq`
: Default to 13 for Lindel method.

`baseAfterPAM.indelFreq`
: Default to 24 for Lindel method.

Value

Four tab delimited files are generated in the output directory: OfftargetAnalysis.xls (detailed information of off targets), Summary.xls (summary of the gRNAs), REcutDetails.xls (restriction enzyme cut sites of each gRNA), and pairedgRNAs.xls (potential paired gRNAs)

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

Doench JG, Hartenian E, Graham DB, Tothova Z, Hegde M, Smith I, Sullender M, Ebert BL, Xavier RJ, Root DE. Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation. Nat Biotechnol. 2014 Sep 3. doi: 10.1038 nbt.3026

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

Moreno-Mateos, M., Vejnar, C., Beaudoin, J. et al. CRISPRscan: designing highly efficient sgRNAs for CRISPR-Cas9 targeting in vivo. Nat Methods 12, 982–988 (2015) doi:10.1038/nmeth.3543

Doench JG et al., Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9. Nature Biotechnology Jan 18th 2016

Anzalone et al., Search-and-replace genome editing without double-strand breaks or donor DNA. Nature October 2019 https://www.nature.com/articles/s41586-019-1711-4

Wei Chen, Aaron McKenna, Jacob Schreiber et al., Massively parallel profiling and predictive modeling of the outcomes of CRISPR/Cas9-mediated double-strand break repair, Nucleic Acids Research, Volume 47, Issue 15, 05 September 2019, Pages 7989–8003, https://doi.org/10.1093/nar/gkz487

See Also

CRISPRseek

Examples

```
library(CRISPRseek)
inputFilePath <- system.file("extdata", "inputseqWithoutBSgenome.fa",
            package = "CRISPRseek")
########### genomeSeq.fasta contains the genomic sequence in fasta format for gRNA and offtarget search########
        genomeSeqFile <- system.file("extdata", "genomeSeq.fasta",
             package = "CRISPRseek")
        library(hash)
        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)
        featureWeightMatrixFile <- system.file("extdata", "Morenos-Mateo.csv",
            package = "CRISPRseek")
results <- offTargetAnalysisWithoutBSgenome(inputFilePath = inputFilePath,
            format = "fasta",
            useBSgenome = FALSE,
            genomeSeqFile = genomeSeqFile,
            findgRNAs = TRUE,
            exportAllgRNAs = "fasta",
            fetchSequence = FALSE,
            findgRNAsWithREcutOnly = FALSE,
            findPairedgRNAOnly = FALSE,
            annotatePaired = FALSE,
    max.mismatch = 1,
            annotateExon = FALSE,
            scoring.method = "CFDscore",
```

```
min.score = 0.01,
PAM = "NGG",
PAM.pattern <- "NNN",
rule.set = "CRISPRscan",
featureWeightMatrixFile = featureWeightMatrixFile,
subPAM.activity  = subPAM.activity,
outputDir = "gRNAoutput-CRISPRseek-CRISPRscan-CFDscore", overwrite = TRUE)
```

`predictRelativeFreqIndels` *Predict insertions and deletions induced by CRISPR/Cas9 editing*

Description

Predict insertions and deletions, and associated reletive frequecies induced by CRISPR/Cas9 editing

Usage

```
predictRelativeFreqIndels(extendedSequence, method = "Lindel")
```

Arguments

`extendedSequence`	A vector of DNA sequences of length 60bp. It consists 30bp before the cut site and 30bp after the cut site.
`method`	the prediction method. default to Lindel. Currently only Lindel method are implemented.

Details

Predict relative indel frequency around target sites of CRISPR/Cas9 system. Currently only Lindel method using logistic regression is implemented in CRISPRseek.

Lindel is compatible with both Python2.7 and Python3.5 or higher.

By default, reticulate uses the version of Python found on your PATH (i.e. Sys.which("python")).

Use the function use_python in reticulate library to set the python path to a specific version. For example, use_python('/opt/anaconda3/lib/python3.7')

This function implements the Lindel method

Value

A list with the same length as the input extendedSequence.

Each list item either contains a warning message, or a predicted fraction of frameshift in the mutational outcomes plus a data frame with three columns.

The three columns are the alignment of predicted indel sequence to the original unedited sequence, predicted indel frequency, and the location of the predicted indels. The warning message for the Lindel method is as follows.

Warning: No PAM sequence is identified. Please check your sequence and try again.

Author(s)

Hui Mao and Lihua Julie Zhu

References

Wei Chen, Aaron McKenna, Jacob Schreiber et al., Massively parallel profiling and predictive modeling of the outcomes of CRISPR/Cas9-mediated double-strand break repair, Nucleic Acids Research, Volume 47, Issue 15, 05 September 2019, Pages 7989–8003, https://doi.org/10.1093/nar/gkz487

Examples

```
extendedSequence <- c("AAA", "TAACGTTATCAACGCCTATATTAAAGCGACCGTCGGTTGAACTGCGTGGATCAATGCGTC")
if (interactive())
    indelFreq <- predictRelativeFreqIndels(extendedSequence, method = "Lindel")
```

---

searchHits *Search for off targets in a sequence as DNAString*

Description

Search for off targets for given gRNAs, sequence and maximum mismatches

Usage

```
searchHits(gRNAs, seqs, seqname, max.mismatch = 3, PAM.size = 3,
                 gRNA.size = 20, PAM = "NGG", PAM.pattern = "NNN$",
                 allowed.mismatch.PAM = 2, PAM.location = "3prime",
                 outfile,
                 baseEditing = FALSE, targetBase = "C", editingWindow = 4:8)
```

Arguments

gRNAs	DNAStringSet object containing a set of gRNAs. Please note the sequences must contain PAM appended after gRNAs, e.g., ATCGAAATTCGAGCCAATCCCGG where ATCGAAATTCGAGCCAATCC is the gRNA and CGG is the PAM
seqs	DNAString object containing a DNA sequence.
seqname	Specify the name of the sequence
max.mismatch	Maximum mismatch allowed in off target search, default 3. Warning: will be considerably slower if it is set to greater than 3
PAM.size	Size of PAM, default 3
gRNA.size	Size of gRNA, default 20
PAM	PAM as regular expression for appending to the gRNA, default NGG for SpCas9, change to TTTN for cpf1.
PAM.pattern	Regular expression of PAM, default N[A|G]G$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence
allowed.mismatch.PAM	Maximum number of mismatches allowed in the offtargets comparing to the PAM sequence. Default to 2 for NGG PAM
PAM.location	PAM location relative to gRNA. For example, spCas9 PAM is located on the 3 prime while cpf1 PAM is located on the 5 prime

outfile	File path to temporarily store the search results
baseEditing	Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.
targetBase	Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.
editingWindow	Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

Value

a data frame contains IsMismatch.posX (indicator variable indicating whether this position X is mismatch or not, 1 means yes and 0 means not, X = 1 to gRNA.size) representing all positions in the gRNA),strand (strand of the match, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name), gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTarget-Sequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (set to 100, and will be updated in getOfftargetScore)

Author(s)

Lihua Julie Zhu

See Also

offTargetAnalysis

Examples

```
 if(interactive())
 {
    all.gRNAs <- findgRNAs(inputFilePath =
        system.file("extdata", "inputseq.fa", package = "CRISPRseek"),
        pairOutputFile = "pairedgRNAs.xls",
findPairedgRNAOnly = TRUE)

    library("BSgenome.Hsapiens.UCSC.hg19")
    ### for speed reason, use max.mismatch = 0 for finding all targets with
    ### all variants of PAM
    hits <- searchHits(all.gRNAs[1], BSgenomeName = Hsapiens,
        max.mismatch = 0, chromToSearch = "chrX")
    colnames(hits)

    ### test PAM located at 5 prime
    all.gRNAs <- findgRNAs(inputFilePath =
             system.file("extdata", "inputseq.fa", package = "CRISPRseek"),
             pairOutputFile = "pairedgRNAs.xls",
             findPairedgRNAOnly = FALSE,
             PAM = "TGT", PAM.location = "5prime")
```

```
    library("BSgenome.Hsapiens.UCSC.hg19")
         ### for speed reason, use max.mismatch = 0 for finding all targets with
         ### all variants of PAM
    hits <- searchHits(all.gRNAs[1], BSgenomeName = Hsapiens, PAM.size = 3,
        max.mismatch = 0, chromToSearch = "chrX", PAM.location = "5prime",
        PAM = "^T[A|G]N", allowed.mismatch.PAM = 2)
    colnames(hits)
}
```

---

searchHits2 *Search for off targets*

Description

Search for off targets for given gRNAs, BSgenome and maximum mismatches

Usage

```
searchHits2(gRNAs, BSgenomeName, chromToSearch = "all", chromToExclude = "",
    max.mismatch = 3,
    PAM.size = 3, gRNA.size = 20, PAM = "NGG", PAM.pattern = "N[A|G]G$",
    allowed.mismatch.PAM = 1, PAM.location = "3prime",
    baseEditing = FALSE, targetBase = "C", editingWindow = 4:8)
```

Arguments

gRNAs	DNAStringSet object containing a set of gRNAs. Please note the sequences must contain PAM appended after gRNAs, e.g., ATCGAAATTCGAGCCAATCCCGG where ATCGAAATTCGAGCCAATCC is the gRNA and CGG is the PAM
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, and BSgenome.Dmelanogaster.UCSC.dm3 for dm3
chromToSearch	Specify the chromosome to search, default to all, meaning search all chromosomes. For example, chrX indicates searching for matching in chromosome X only
chromToExclude	Specify the chromosome not to search, default to none, meaning to search chromosomes specified by chromToSearch. For example, to exclude haplotype blocks from offtarget search in hg19, set chromToExclude to c(""chr17_ctg5_hap1","chr4_ctg9_hap1", "chr6_apd_hap1", "chr6_cox_hap2", "chr6_dbb_hap3", "chr6_mann_hap4", "chr6_mcf_hap5","chr6_ "chr6_ssto_hap7")
max.mismatch	Maximum mismatch allowed in off target search, default 3. Warning: will be considerably slower if it is set to greater than 3
PAM.size	Size of PAM, default 3
gRNA.size	Size of gRNA, default 20
PAM	Regular expression of protospacer-adjacent motif (PAM), default NGG for spCas9. For cpf1, ^TTTN
PAM.pattern	Regular expression of PAM, default N[A|G]G$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence

`allowed.mismatch.PAM`
: Number of degenerative bases in the PAM sequence, default to 1 for N[A|G]G PAM

`PAM.location`
: PAM location relative to gRNA. For example, spCas9 PAM is located on the 3 prime while cpf1 PAM is located on the 5 prime

`baseEditing`
: Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.

`targetBase`
: Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.

`editingWindow`
: Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

Value

a data frame contains IsMismatch.posX (indicator variable indicating whether this position X is mismatch or not, 1 means yes and 0 means not, X = 1 to gRNA.size) representing all positions in the gRNA),strand (strand of the match, + for plus and - for minus strand), chrom (chromosome of the off target), chromStart (start position of the off target), chromEnd (end position of the off target),name (gRNA name), gRNAPlusPAM (gRNA sequence with PAM sequence concatenated), OffTargetSequence (the genomic sequence of the off target), n.mismatch (number of mismatches between the off target and the gRNA), forViewInUCSC (string for viewing in UCSC genome browser, e.g., chr14:31665685-31665707), score (set to 100, and will be updated in getOfftargetScore)

Author(s)

Lihua Julie Zhu

See Also

offTargetAnalysis

Examples

```
    all.gRNAs <- findgRNAs(inputFilePath =
        system.file("extdata", "inputseq.fa", package = "CRISPRseek"),
        pairOutputFile = "pairedgRNAs.xls",
findPairedgRNAOnly = TRUE)

    library("BSgenome.Hsapiens.UCSC.hg19")
    ### for speed reason, use max.mismatch = 0 for finding all targets with
    ### all variants of PAM
    hits <- searchHits2(all.gRNAs[1], BSgenomeName = Hsapiens,
        max.mismatch = 0, chromToSearch = "chrX")
    colnames(hits)

    ### test PAM located at 5 prime
    all.gRNAs <- findgRNAs(inputFilePath =
             system.file("extdata", "inputseq.fa", package = "CRISPRseek"),
             pairOutputFile = "pairedgRNAs.xls",
             findPairedgRNAOnly = FALSE,
```

```
          PAM = "TGT", PAM.location = "5prime")

library("BSgenome.Hsapiens.UCSC.hg19")
     ### for speed reason, use max.mismatch = 0 for finding all targets with
     ### all variants of PAM
hits <- searchHits2(all.gRNAs[1], BSgenomeName = Hsapiens, PAM.size = 3,
    max.mismatch = 0, chromToSearch = "chrX", PAM.location = "5prime",
    PAM = "NGG",
    PAM.pattern = "^T[A|G]N", allowed.mismatch.PAM = 2)
colnames(hits)
```

`translatePattern` *translate pattern from IUPAC Extended Genetic Alphabet to regular expression*

Description

translate pattern containing the IUPAC nucleotide ambiguity codes to regular expression. For example,Y->[C|T], R-> [A|G], S-> [G|C], W-> [A|T], K-> [T|U|G], M-> [A|C], B-> [C|G|T], D-> [A|G|T], H-> [A|C|T], V-> [A|C|G] and N-> [A|C|T|G].

Usage

```
translatePattern(pattern)
```

Arguments

`pattern`	a character vector with the IUPAC nucleotide ambiguity codes

Value

a character vector with the pattern represented as regular expression

Author(s)

Lihua Julie Zhu

Examples

```
pattern1 <- "AACCNWMK"
translatePattern(pattern1)
```

uniqueREs *Output restriction enzymes that recognize only the gRNA cleavage sites*

Description

For each identified gRNA, output restriction enzymes that recognize only the gRNA cleavage sites.

Usage

```
uniqueREs(REcutDetails, summary, offTargets, scanUpstream = 100,
   scanDownstream = 100, BSgenomeName)
```

Arguments

REcutDetails	REcutDetails stored in the REcutDetails.xls
summary	summary stored in the summary.xls
offTargets	offTargets stored in the offTargets.xls
scanUpstream	upstream offset from the gRNA start, default 100
scanDownstream	downstream offset from the gRNA end, default 100
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

Value

returns the RE sites that recognize only the gRNA cleavage sites for each gRNA.

Author(s)

Lihua Julie Zhu

Examples

```
    library("BSgenome.Hsapiens.UCSC.hg19")
    load(system.file("extdata", "ForTestinguniqueREs.RData",
            package = "CRISPRseek"))
    uniqueREs(results$REcutDetails, results$summary, results$offtarget,
scanUpstream = 50,
        scanDownstream = 50, BSgenomeName = Hsapiens)
```

writeHits *Write the hits of sequence search from a sequence to a file*

Description

write the hits of sequence search from a sequence instead of BSgenome to a file, internal function used by searchHits

Usage

```
writeHits(gRNA, seqname, matches, strand, file, gRNA.size = 20L,
     PAM = "NGG", PAM.pattern = "N[A|G]G$", max.mismatch = 4L,
     chrom.len, append = FALSE, PAM.location = "3prime",
     PAM.size = 3L, allowed.mismatch.PAM = 1L,
     seqs,
    baseEditing = FALSE, targetBase = "C", editingWindow = 4:8)
```

Arguments

gRNA	DNAString object with gRNA sequence with PAM appended immediately after,e.g., ACGTACGTACGTACTGACGTCGG with 20bp gRNA sequence plus 3bp PAM sequence CGG
seqname	sequence name as character
matches	XStringViews object storing matched chromosome locations
strand	strand of the match, + for plus strand and - for minus strand
file	file path where the hits is written to
gRNA.size	gRNA size, default 20
PAM	PAM as regular expression for appending to the gRNA, default NGG for SpCas9, change to TTTN for cpf1.
PAM.pattern	PAM as regular expression for filtering the hits, default N[A|G]G$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence.
max.mismatch	maximum mismatch allowed within the gRNA (excluding PAM portion) for filtering the hits, default 4
chrom.len	length of the matched chromosome
append	TRUE if append to existing file, false if start a new file
PAM.location	PAM location relative to gRNA. For example, spCas9 PAM is located on the 3 prime while cpf1 PAM is located on the 5 prime
PAM.size	Size of PAM, default 3
allowed.mismatch.PAM	Maximum number of mismatches allowed in the offtargets comparing to the PAM sequence. Default to 1 for NGG PAM
seqs	DNAString object containing a DNA sequence.
baseEditing	Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.
targetBase	Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.

editingWindow Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

Value

results are saved in the file specified by file

Author(s)

Lihua Julie Zhu

References

http://bioconductor.org/packages/2.8/bioc/vignettes/BSgenome/inst/doc/ GenomeSearching.pdf

See Also

offTargetAnalysis

Examples

```
if(interactive())
{
   gRNAPlusPAM <- DNAString("ACGTACGTACGTACTGACGTCGG")
   x <- DNAString("AAGCGCGATATGACGTACGTACGTACTGACGTCGG")
   chrom.len <- nchar(as.character(x))
   m <- matchPattern(gRNAPlusPAM, x)
   names(m) <- "testing"
   writeHits(gRNA = gRNAPlusPAM, seqname = "chr1",
       matches = m, strand = "+", file = "exampleWriteHits.txt",
       chrom.len = chrom.len, append = FALSE)
}
```

---

writeHits2 *Write the hits of sequence search to a file*

Description

write the hits of sequence search to a file, internal function used by searchHits

Usage

```
writeHits2(gRNA, seqname, matches, strand, file, gRNA.size = 20,
    PAM = "NGG", PAM.pattern = "N[A|G]G$",
    max.mismatch = 4, chrom.len, append = FALSE,
    PAM.location = "3prime", PAM.size = 3,
    allowed.mismatch.PAM = 1L,
    BSgenomeName, baseEditing = FALSE, targetBase = "C", editingWindow = 4:8)
```

Arguments

gRNA	DNAString object with gRNA sequence with PAM appended immediately after,e.g., ACGTACGTACGTACTGACGTCGG with 20bp gRNA sequence plus 3bp PAM sequence CGG
seqname	chromosome name as character, e.g., chr1
matches	XStringViews object storing matched chromosome locations
strand	strand of the match, + for plus strand and - for minus strand
file	file path where the hits is written to
gRNA.size	gRNA size, default 20
PAM	PAM as regular expression for filtering the hits, default NGG for spCas9. For cpf1, TTTN.
PAM.pattern	Regular expression of protospacer-adjacent motif (PAM), default N[A|G]G$ for spCas9. For cpf1, ^TTTN since it is a 5 prime PAM sequence
max.mismatch	maximum mismatch allowed within the gRNA (excluding PAM portion) for filtering the hits, default 4
chrom.len	length of the matched chromosome
append	TRUE if append to existing file, false if start a new file
PAM.location	PAM location relative to gRNA. For example, spCas9 PAM is located on the 3 prime while cpf1 PAM is located on the 5 prime
PAM.size	Size of PAM, default 3
allowed.mismatch.PAM	Number of degenerative bases in the PAM sequence, default to 1 for N[A|G]G PAM
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
baseEditing	Indicate whether to design gRNAs for base editing. Default to FALSE If TRUE, please set baseEditing = TRUE, targetBase and editingWidow accordingly.
targetBase	Applicable only when baseEditing is set to TRUE. It is used to indicate the target base for base editing systems, default to C for converting C to T in the CBE system. Please change it to A if you intend to use the ABE system.
editingWindow	Applicable only when baseEditing is set to TRUE. It is used to indicate the effective editing window to consider for the offtargets search only, default to 4 to 8 which is for the original CBE system. Please change it accordingly if the system you use have a different editing window, or you would like to include offtargets with the target base in a larger editing window.

Value

results are saved in the file specified by file

Author(s)

Lihua Julie Zhu

References

http://bioconductor.org/packages/2.8/bioc/vignettes/BSgenome/inst/doc/ GenomeSearching.pdf

See Also

offTargetAnalysis

Examples

```
library("BSgenome.Hsapiens.UCSC.hg19")
gRNAPlusPAM <- DNAString("ACGTACGTACGTACTGACGTCGG")
x <- DNAString("AAGCGCGATATGACGTACGTACGTACTGACGTCGG")
chrom.len <- nchar(as.character(x))
m <- matchPattern(gRNAPlusPAM, x)
names(m) <- "testing"
writeHits2(gRNA = gRNAPlusPAM, seqname = "chr1",
    PAM = "NGG", PAM.pattern = "NNN$", allowed.mismatch.PAM = 2,
    matches = m, strand = "+", file = "exampleWriteHits.txt",
    chrom.len = chrom.len, append = FALSE, BSgenomeName = Hsapiens)
```

Index