

Package ‘MMAPPR2’

July 1, 2022

Title Mutation Mapping Analysis Pipeline for Pooled RNA-Seq

Version 1.11.0

Description MMAPPR2 maps mutations resulting from pooled RNA-seq data from the F2 cross of forward genetic screens. Its predecessor is described in a paper published in Genome Research (Hill et al. 2013). MMAPPR2 accepts aligned BAM files as well as a reference genome as input, identifies loci of high sequence disparity between the control and mutant RNA sequences, predicts variant effects using Ensembl's Variant Effect Predictor, and outputs a ranked list of candidate mutations.

Depends R (>= 3.6.0)

License GPL-3

Encoding UTF-8

RoxygenNote 6.1.1

VignetteBuilder knitr

Suggests testthat, mockery, roxygen2, knitr, rmarkdown, BiocStyle, MMAPPR2data

Imports ensemblVEP (>= 1.20.0), gmapR, Rsamtools, VariantAnnotation, BiocParallel, Biobase, BiocGenerics, dplyr, GenomeInfoDb, GenomicRanges, IRanges, S4Vectors, tidyrr, VariantTools, magrittr, methods, grDevices, graphics, stats, utils, stringr, data.table

SystemRequirements Ensembl VEP, Samtools

biocViews RNASeq, PooledScreens, DNASEq, VariantDetection

URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3613585/>,
<https://github.com/kjohnsen/MMAPPR2>

BugReports <https://github.com/kjohnsen/MMAPPR2/issues>

OS_type unix

git_url <https://git.bioconductor.org/packages/MMAPPR2>

git_branch master

git_last_commit 33bc9af

git_last_commit_date 2022-04-26

Date/Publication 2022-07-01

Author Kyle Johnsen [aut],
Nathaniel Jenkins [aut],
Jonathon Hill [cre]

Maintainer Jonathon Hill <jhill@byu.edu>

R topics documented:

calculateDistance	2
generateCandidates	3
loessFit	4
mmappr	5
MMAPPR2	6
MmapprData-class	6
MmapprData-getters	7
MmapprParam-class	8
MmapprParam-functions	10
outputMmapprData	13
peakRefinement	14
prePeak	14
tempOutputFolder	15
Index	17

calculateDistance	<i>Read BAM files and generate Euclidean distance data</i>
-------------------	--

Description

First step in the MMAPPR2 pipeline. Precedes the [loessFit](#) step.

Usage

```
calculateDistance(mmapprData)
```

Arguments

`mmapprData` The [MmapprData](#) object to be analyzed.

Value

A [MmapprData](#) object with the distance slot filled.

Examples

```

if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                              wtFiles = MMAPPR2data::exampleWTbam(),
                              mutFiles = MMAPPR2data::exampleMutBam(),
                              species = "danio_rerio",
                              outputFolder = tempOutputFolder())

  md <- new('MmapprData', param = mmappr_param)
  postCalcDistMD <- calculateDistance(md)
}

```

generateCandidates	<i>Generate potential causative mutations and consequences in peak regions</i>
--------------------	--

Description

Follows the [peakRefinement](#) step and produces a [MmapprData](#) object ready for [outputMmapprData](#).

Usage

```
generateCandidates(mmapprData)
```

Arguments

mmapprData The [MmapprData](#) object to be analyzed.

Value

A [MmapprData](#) object with the candidates slot filled with a [GRanges](#) object for each peak chromosome containing variants and predicted consequences from Ensembl's Variant Effect Predictor.

Examples

```

if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                              wtFiles = MMAPPR2data::exampleWTbam(),
                              mutFiles = MMAPPR2data::exampleMutBam(),
                              species = "danio_rerio",
                              outputFolder = tempOutputFolder())
}

## Not run:
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)
postLoessMD <- loessFit(postCalcDistMD)
postPrePeakMD <- prePeak(postLoessMD)

```

```
postPeakRefMD <- peakRefinement(postPrePeakMD)

postCandidatesMD <- generateCandidates(postPeakRefMD)

## End(Not run)
```

loessFit*Perform optimized Loess regression for each chromosome*

Description

Called after the [calculateDistance](#) step and before [prePeak](#).

Usage

```
loessFit(mmapprData)
```

Arguments

mmapprData The [MmapprData](#) object to be analyzed.

Value

A [MmapprData](#) object with the \$loess element of the distance slot list filled.

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                              wtFiles = MMAPPR2data::exampleWTbam(),
                              mutFiles = MMAPPR2data::exampleMutBam(),
                              species = "danio_rerio",
                              outputFolder = tempOutputFolder())
}
## Not run:
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)

postLoessMD <- loessFit(postCalcDistMD)

## End(Not run)
```

Description

MMAPPR2 is designed to map the causative mutation in a forward genetics screen. It analyzes aligned sequence files, calculates the per-base Euclidean distance between the mutant and wild-type pools, performs a Loess regression on that distance, and generates candidate variants in regions of peak distance.

Usage

```
mmappr(mmapprParam)
```

Arguments

`mmapprParam` A [MmapprParam](#) object containing desired parameters.

Value

A [MmapprData](#) object containing results and/or intermediate data.

See Also

[calculateDistance](#), [loessFit](#), [prePeak](#), [peakRefinement](#), [generateCandidates](#), [outputMmapprData](#)

Examples

```
if (requireNamespace('MMAPPR2data', quietly = TRUE)
    & all(Sys.which(c('vep', 'samtools')) != '')) {

  # Specify parameters:
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                              wtFiles = MMAPPR2data::exampleWTbam(),
                              mutFiles = MMAPPR2data::exampleMutBam(),
                              species = "danio_rerio",
                              outputFolder = tempOutputFolder())

  # Run pipeline:
  mmapprData <- mmappr(mmappr_param)

}

## Not run:
### Alternately, you can navigate the pipeline step by step.
### This may be helpful for debugging.
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)
postLoessMD <- loessFit(postCalcDistMD)
postPrePeakMD <- prePeak(postLoessMD)
```

```

postPeakRefMD <- peakRefinement(postPrePeakMD)
postCandidatesMD <- generateCandidates(postPeakRefMD)
outputMmapprData(postCandidatesMD)

## End(Not run)

```

MMAPPR2

Mutation Mapping Analysis Pipeline for Pooled RNA-seq

Description

The main functionality of this package is described in the [mmappr](#) function.

MmapprData-class

MmapprData Class

Description

Stores data from each step of the MMAPPR2 pipeline.

Slots

param [MmapprParam](#) object storing parameters used in analysis.

distance List containing raw counts and Euclidean distance data for each chromosome. After [calculateDistance](#), chromosomes with sufficient data should have \$wtCounts, \$mutCounts, and \$distanceDf populated. After [loessFit](#), the \$distanceDf element for each chromosome list is replaced with a \$loess element.

peaks List of chromosomes containing peak regions. Initialized after [prePeak](#) and populated with density function after [peakRefinement](#).

candidates List containing [GRanges](#) object for each peak, resulting from [generateCandidates](#) function. VEP data, including gene symbol and consequence for each variant, are included in metacolumns.

See Also

[mmappr](#), [MmapprData-getters](#)

Description

Access slots of [MmapprData](#) object.

Usage

```
## S4 method for signature 'MmapprData'  
param(obj)
```

```
## S4 method for signature 'MmapprData'  
distance(obj)
```

```
## S4 method for signature 'MmapprData'  
peaks(obj)
```

```
## S4 method for signature 'MmapprData'  
candidates(obj)
```

Arguments

obj Desired [MmapprData](#) object.

Value

Desired attribute.

See Also

[MmapprData](#)

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)  
    & all(Sys.which(c("samtools", "vep")) != "")) {  
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),  
                              wtFiles = MMAPPR2data::exampleWTbam(),  
                              mutFiles = MMAPPR2data::exampleMutBam(),  
                              species = "danio_rerio",  
                              outputFolder = tempOutputFolder())  
  
  md <- new('MmapprData', param = mmappr_param)  
  
  param(md)  
  distance(md)  
  peaks(md)  
  candidates(md)
```

```
}

```

MmapprParam-class *MmapprParam Class and Constructor*

Description

MmapprParam stores parameters for running `mmappr`.

Usage

```
MmapprParam(refFasta, wtFiles, mutFiles, species, vepFlags = NULL,
  refGenome = NULL, outputFolder = NULL, distancePower = 4,
  peakIntervalWidth = 0.95, minDepth = 10, homozygoteCutoff = 0.95,
  minBaseQuality = 20, minMapQuality = 30,
  loessOptResolution = 0.001, loessOptCutFactor = 0.1, naCutoff = 0,
  fileAggregation = c("sum", "mean"))
```

Arguments

refFasta	The path to the fasta file genome, which will be referenced in reading the BAM files.
wtFiles	Character vector, BamFile , or BamFileList containing BAM files for the wild-type pool to be analyzed.
mutFiles	Character vector, BamFile , or BamFileList containing BAM files for the mutant pool to be analyzed.
species	Length-one character vector of name of species under analysis. Used only in generating default VEPFlags object.
vepFlags	Optional VEPFlags object containing runtime options for Ensembl's Variant Effect Predictor. See vignette for details. Generated by default.
refGenome	GmapGenome object storing reference genome to be used in variant calling. Make sure it is the same genome aligned to and used installed with VEP. Generated by default.
outputFolder	Length-one character vector specifying where to save output, including a MmapprData stored as <code>mmappr_data.RDS</code> , <code>mmappr2.log</code> , a <code>.tsv</code> file for each peak chromosome containing candidate mutations, and PDF plots of both the entire genome and peak chromosomes. Defaults to an automatically generated <code>mmappr2_<timestamp></code> .
distancePower	Length-one numeric vector determining to what power Euclidean distance values are raised before fitting. Higher powers tend to increase high values and decrease low values, exaggerating the variation in the data. Default of 4.
peakIntervalWidth	Length-one numeric vector between 0 and 1 specifying desired width of linkage region(s). The default value of 0.95, for example, yields peak regions defined as including the top 95% of SNPs in the peak region, as determined by the peak resampling distribution.

MmapprParam-functions *MmapprParam Getters and Setters*

Description

Access and assign slots of [MmapprParam](#) object.

Usage

```
## S4 method for signature 'MmapprParam'  
refFasta(obj)
```

```
## S4 method for signature 'MmapprParam'  
wtFiles(obj)
```

```
## S4 method for signature 'MmapprParam'  
mutFiles(obj)
```

```
## S4 method for signature 'MmapprParam'  
species(obj)
```

```
## S4 method for signature 'MmapprParam'  
vepFlags(obj)
```

```
## S4 method for signature 'MmapprParam'  
refGenome(obj)
```

```
## S4 method for signature 'MmapprParam'  
homozygoteCutoff(obj)
```

```
## S4 method for signature 'MmapprParam'  
distancePower(obj)
```

```
## S4 method for signature 'MmapprParam'  
peakIntervalWidth(obj)
```

```
## S4 method for signature 'MmapprParam'  
minDepth(obj)
```

```
## S4 method for signature 'MmapprParam'  
minBaseQuality(obj)
```

```
## S4 method for signature 'MmapprParam'  
minMapQuality(obj)
```

```
## S4 method for signature 'MmapprParam'  
loessOptResolution(obj)
```

```
## S4 method for signature 'MmapprParam'
loessOptCutFactor(obj)

## S4 method for signature 'MmapprParam'
naCutoff(obj)

## S4 method for signature 'MmapprParam'
outputFolder(obj)

## S4 method for signature 'MmapprParam'
fileAggregation(obj)

## S4 replacement method for signature 'MmapprParam'
refFasta(obj) <- value

## S4 replacement method for signature 'MmapprParam'
wtFiles(obj) <- value

## S4 replacement method for signature 'MmapprParam'
mutFiles(obj) <- value

## S4 replacement method for signature 'MmapprParam'
vepFlags(obj) <- value

## S4 replacement method for signature 'MmapprParam'
refGenome(obj) <- value

## S4 replacement method for signature 'MmapprParam'
species(obj) <- value

## S4 replacement method for signature 'MmapprParam'
homozygoteCutoff(obj) <- value

## S4 replacement method for signature 'MmapprParam'
distancePower(obj) <- value

## S4 replacement method for signature 'MmapprParam'
peakIntervalWidth(obj) <- value

## S4 replacement method for signature 'MmapprParam'
minDepth(obj) <- value

## S4 replacement method for signature 'MmapprParam'
minBaseQuality(obj) <- value

## S4 replacement method for signature 'MmapprParam'
loessOptResolution(obj) <- value
```

```
## S4 replacement method for signature 'MmapprParam'  
loessOptCutFactor(obj) <- value  
  
## S4 replacement method for signature 'MmapprParam'  
naCutoff(obj) <- value  
  
## S4 replacement method for signature 'MmapprParam'  
outputFolder(obj) <- value  
  
## S4 replacement method for signature 'MmapprParam'  
minMapQuality(obj) <- value  
  
## S4 replacement method for signature 'MmapprParam'  
fileAggregation(obj) <- value
```

Arguments

obj	Desired MmapprParam object.
value	Value to replace desired attribute.

Value

The desired [MmapprParam](#) attribute.

See Also

[MmapprParam](#)

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)  
    & all(Sys.which(c("samtools", "vep")) != "")) {  
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),  
                              wtFiles = MMAPPR2data::exampleWTbam(),  
                              mutFiles = MMAPPR2data::exampleMutBam(),  
                              species = "danio_rerio")  
  
  outputFolder(mmappr_param) <- 'mmappr2_test_1'  
  minBaseQuality(mmappr_param) <- 25  
  vepFlags(mmappr_param)  
}
```

outputMmapprData	<i>Generate plots and tables from MMAPPR2 data</i>
------------------	--

Description

Generate plots and tables from MMAPPR2 data

Usage

```
outputMmapprData(mmapprData)
```

Arguments

mmapprData The [MmapprData](#) object to be output

Value

A [MmapprData](#) object after writing output files to the folder specified in the outputFolder slot of the link{MmapprParam} used.

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                             wtFiles = MMAPPR2data::exampleWTbam(),
                             mutFiles = MMAPPR2data::exampleMutBam(),
                             species = "danio_rerio",
                             outputFolder = tempOutputFolder())
}
## Not run:
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)
postLoessMD <- loessFit(postCalcDistMD)
postPrePeakMD <- prePeak(postLoessMD)
postPeakRefMD <- peakRefinement(postPrePeakMD)
postCandidatesMD <- generateCandidates(postPeakRefMD)

outputMmapprData(postCandidatesMD)

## End(Not run)
```

peakRefinement	<i>Characterize Euclidean distance peaks using resampling simulation</i>
----------------	--

Description

Follows the [prePeak](#) step and precedes [generateCandidates](#).

Usage

```
peakRefinement(mmapprData)
```

Arguments

`mmapprData` The [MmapprData](#) object to be analyzed.

Value

A [MmapprData](#) object with the peaks slot filled and populated.

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                             wtFiles = MMAPPR2data::exampleWTbam(),
                             mutFiles = MMAPPR2data::exampleMutBam(),
                             species = "danio_rerio",
                             outputFolder = tempOutputFolder())
}
## Not run:
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)
postLoessMD <- loessFit(postCalcDistMD)
postPrePeakMD <- prePeak(postLoessMD)

postPeakRefMD <- peakRefinement(postPrePeakMD)

## End(Not run)
```

prePeak	<i>Identify chromosomes containing peaks</i>
---------	--

Description

Follows the [loessFit](#) step and precedes [peakRefinement](#).

Usage

```
prePeak(mmapprData)
```

Arguments

mmapprData The `MmapprData` object to be analyzed.

Value

A `MmapprData` object with the peaks slot initialized.

Examples

```
if (requireNamespace('MMAPPR2data', quietly=TRUE)
    & all(Sys.which(c("samtools", "vep")) != "")) {
  mmappr_param <- MmapprParam(refFasta = MMAPPR2data::goldenFasta(),
                              wtFiles = MMAPPR2data::exampleWTbam(),
                              mutFiles = MMAPPR2data::exampleMutBam(),
                              species = "danio_rerio",
                              outputFolder = tempOutputFolder())
}
## Not run:
md <- new('MmapprData', param = mmappr_param)
postCalcDistMD <- calculateDistance(md)
postLoessMD <- loessFit(postCalcDistMD)

postPrePeakMD <- prePeak(postLoessMD)

## End(Not run)
```

tempOutputFolder *Generate temporary output folder*

Description

Conveniently creates a timestamp-named temporary directory

Usage

```
tempOutputFolder()
```

Value

The path to the temporary directory

Index

BamFile, 8
BamFileList, 8

calculateDistance, 2, 4–6
candidates (MmapprData-getters), 7
candidates, MmapprData-method
 (MmapprData-getters), 7

distance (MmapprData-getters), 7
distance, MmapprData-method
 (MmapprData-getters), 7
distancePower (MmapprParam-functions),
 10
distancePower, MmapprParam-method
 (MmapprParam-functions), 10
distancePower<-
 (MmapprParam-functions), 10
distancePower<-, MmapprParam-method
 (MmapprParam-functions), 10

fileAggregation
 (MmapprParam-functions), 10
fileAggregation, MmapprParam-method
 (MmapprParam-functions), 10
fileAggregation<-
 (MmapprParam-functions), 10
fileAggregation<-, MmapprParam-method
 (MmapprParam-functions), 10

generateCandidates, 3, 5, 6, 14
GmapGenome, 8
GRanges, 3, 6

homozygoteCutoff
 (MmapprParam-functions), 10
homozygoteCutoff, MmapprParam-method
 (MmapprParam-functions), 10
homozygoteCutoff<-
 (MmapprParam-functions), 10
homozygoteCutoff<-, MmapprParam-method
 (MmapprParam-functions), 10

loessFit, 2, 4, 5, 6, 14
loessOptCutFactor
 (MmapprParam-functions), 10
loessOptCutFactor, MmapprParam-method
 (MmapprParam-functions), 10
loessOptCutFactor<-
 (MmapprParam-functions), 10
loessOptCutFactor<-, MmapprParam-method
 (MmapprParam-functions), 10
loessOptResolution
 (MmapprParam-functions), 10
loessOptResolution, MmapprParam-method
 (MmapprParam-functions), 10
loessOptResolution<-
 (MmapprParam-functions), 10
loessOptResolution<-, MmapprParam-method
 (MmapprParam-functions), 10

minBaseQuality (MmapprParam-functions),
 10
minBaseQuality, MmapprParam-method
 (MmapprParam-functions), 10
minBaseQuality<-
 (MmapprParam-functions), 10
minBaseQuality<-, MmapprParam-method
 (MmapprParam-functions), 10
minDepth (MmapprParam-functions), 10
minDepth, MmapprParam-method
 (MmapprParam-functions), 10
minDepth<- (MmapprParam-functions), 10
minDepth<-, MmapprParam-method
 (MmapprParam-functions), 10
minMapQuality (MmapprParam-functions),
 10
minMapQuality, MmapprParam-method
 (MmapprParam-functions), 10
minMapQuality<-
 (MmapprParam-functions), 10
minMapQuality<-, MmapprParam-method
 (MmapprParam-functions), 10

- mmappr, [5](#), [6](#), [8](#)
- MMAPPR2, [6](#)
- MMAPPR2-package (MMAPPR2), [6](#)
- MmapprData, [2–5](#), [7](#), [8](#), [13–15](#)
- MmapprData (MmapprData-class), [6](#)
- MmapprData-class, [6](#)
- MmapprData-getters, [6](#), [7](#)
- MmapprParam, [5](#), [6](#), [10](#), [12](#)
- MmapprParam (MmapprParam-class), [8](#)
- MmapprParam-class, [8](#)
- MmapprParam-functions, [10](#)
- mutFiles (MmapprParam-functions), [10](#)
- mutFiles, MmapprParam-method
(MmapprParam-functions), [10](#)
- mutFiles<- (MmapprParam-functions), [10](#)
- mutFiles<-, MmapprParam-method
(MmapprParam-functions), [10](#)

- naCutoff (MmapprParam-functions), [10](#)
- naCutoff, MmapprParam-method
(MmapprParam-functions), [10](#)
- naCutoff<- (MmapprParam-functions), [10](#)
- naCutoff<-, MmapprParam-method
(MmapprParam-functions), [10](#)

- outputFolder (MmapprParam-functions), [10](#)
- outputFolder, MmapprParam-method
(MmapprParam-functions), [10](#)
- outputFolder<- (MmapprParam-functions),
[10](#)
- outputFolder<-, MmapprParam-method
(MmapprParam-functions), [10](#)
- outputMmapprData, [3](#), [5](#), [13](#)

- param (MmapprData-getters), [7](#)
- param, MmapprData-method
(MmapprData-getters), [7](#)
- peakIntervalWidth
(MmapprParam-functions), [10](#)
- peakIntervalWidth, MmapprParam-method
(MmapprParam-functions), [10](#)
- peakIntervalWidth<-
(MmapprParam-functions), [10](#)
- peakIntervalWidth<-, MmapprParam-method
(MmapprParam-functions), [10](#)
- peakRefinement, [3](#), [5](#), [6](#), [14](#), [14](#)
- peaks (MmapprData-getters), [7](#)
- peaks, MmapprData-method
(MmapprData-getters), [7](#)

- prePeak, [4–6](#), [14](#), [14](#)

- refFasta (MmapprParam-functions), [10](#)
- refFasta, MmapprParam-method
(MmapprParam-functions), [10](#)
- refFasta<- (MmapprParam-functions), [10](#)
- refFasta<-, MmapprParam-method
(MmapprParam-functions), [10](#)
- refGenome (MmapprParam-functions), [10](#)
- refGenome, MmapprParam-method
(MmapprParam-functions), [10](#)
- refGenome<- (MmapprParam-functions), [10](#)
- refGenome<-, MmapprParam-method
(MmapprParam-functions), [10](#)

- species (MmapprParam-functions), [10](#)
- species, MmapprParam-method
(MmapprParam-functions), [10](#)
- species<- (MmapprParam-functions), [10](#)
- species<-, MmapprParam-method
(MmapprParam-functions), [10](#)

- tempOutputFolder, [15](#)

- VEPFlags, [8](#)
- vepFlags (MmapprParam-functions), [10](#)
- vepFlags, MmapprParam-method
(MmapprParam-functions), [10](#)
- vepFlags<- (MmapprParam-functions), [10](#)
- vepFlags<-, MmapprParam-method
(MmapprParam-functions), [10](#)

- wtFiles (MmapprParam-functions), [10](#)
- wtFiles, MmapprParam-method
(MmapprParam-functions), [10](#)
- wtFiles<- (MmapprParam-functions), [10](#)
- wtFiles<-, MmapprParam-method
(MmapprParam-functions), [10](#)