Package 'ExomeDepth'

July 20, 2014

Type Package

Title Calls CNV from exome sequence data

Version 1.0.7

Date 2014-7-20

Depends R (>= 2.15.0), methods, aod, VGAM (>= 0.8.4), GenomicRanges (>= 1.8.10), Rsamtools

Author Vincent Plagnol

Maintainer Vincent Plagnol <v.plagnol@ucl.ac.uk>

Description Calls copy number variants (CNVs) from targeted sequence data

License GPL-3

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-07-20 08:26:28

R topics documented:

ExomeDepth-package 2
AnnotateExtra 3
AnnotateExtra-methods 4
CallCNVs 4
Conrad.hg19.common.CNVs 5
count.everted.reads 6
countBam.everted 7
countBamInGRanges.exomeDepth 8
ExomeCount 9
ExomeDepth-class 10
exons.hg19 11
exons.hg19.X 12
genes.hg19 12

get.power.betabinom 13
getBamCounts 14
initialize-methods 15
plot-methods 15
qbetabinom 16
qbetabinom.ab 16
select.reference.set 17
show-methods 18
somatic.CNV.call 19
TestCNV 20
viterbi.hmm 20

Index **22**

`ExomeDepth-package` *Read depth based CNV calls for exome DNA sequence data*

Description

ExomeDepth uses read count data from exome or targeted sequencing experiments to call copy number variants.

Details

Package:	ExomeDepth
Type:	Package
Version:	0.1
Date:	2012-02-01
License:	What license is it under?
LazyLoad:	yes

The two key functions are: select.reference.set CallCNVs

Author(s)

Vincent Plagnol Maintainer: Vincent Plagnol <v.plagnol@ucl.ac.uk>

References

A robust model for read count data in exome sequencing experiments and implications for copy number variant calling, Bioinformatics, In Press

AnnotateExtra *Add annotations to a ExomeDepth object*

Description

Takes annotations in the GRanges format and adds these to the CNV calls in the ExomeDepth object.

Usage

```
AnnotateExtra(x, reference.annotation, min.overlap = 0.5, column.name = "overlap")
```

Arguments

x	An ExomeDepth object.
reference.annotation	The list of reference annotations in GRanges format.
min.overlap	The minimum fraction of the CNV call that is covered by the reference call to declare that there is a significant overlap.
column.name	The name of the column used to store the overlap (in the slot CNV.calls).

Details

A recent version of GenomicRanges (> 1.8.10) is required. Otherwise the function will return a warning and not update the ExomeDepth object.

Value

A ExomeDepth object with the relevant annotations added to the CNVcalls slot.

Author(s)

Vincent Plagnol

References

A robust model for read count data in exome sequencing experiments and implications for copy number variant calling, Plagnol et al, Bioinformatics

AnnotateExtra-methods *Additional annotations for ExomeDepth objects*

Description

Takes a GRanges object, typically a list of reference calls, and add these annotations to a ExomeDepth object.

Methods

```
signature(x = "ExomeDepth")
```

CallCNVs *Call CNV data from an ExomeDepth object.*

Description

The function must be called on an ExomeDepth object. Likelihood data must have been precomputed which should have been done by default when the ExomeDepth object was created.

Usage

```
CallCNVs(x, chromosome, start, end, name, transition.probability =
0.0001, expected.CNV.length = 50000)
```

Arguments

x	An ExomeDepth object
chromosome	Chromosome information for each exon (factor).
start	Start (physical position) of each exon (numeric, must have the same length as the chromosome argument).
end	End (physical position) of each exome (numeric, must have the same length as the chromosome argument).
name	Name of each exon (character or factor).
transition.probability	Transition probability of the hidden Markov Chain from the normal copy number state to either a deletion or a duplication. The default (0.0001) expect approximately 20 CNVs genome-wide.
expected.CNV.length	The expectation for the length of a CNV. This value factors into the Viterbi algorithm that is used to compte the transition from one state to the next, which depends on the distance between exons.

Details

This function fits a hidden Markov model to the read depth data with three hidden states (normal, deletion, duplication).

Value

The same ExomeDepth object provided as input but with the slot CNVcalls containing a data frame with the output of the calling.

Author(s)

Vincent Plagnol

`Conrad.hg19.common.CNVs`
Conrad et al common CNVs

Description

Positions of common CNV calls (detected in a panel of 42 sample) from the Conrad et al paper (Nature 2010). This is build hg19 of the human genome.

Usage

```
data(Conrad.hg19)
```

Format

A data frame with common CNV calls.

Source

Conrad et al, Origins and functional impact of copy number variation in the human genome, Nature 2010

`count.everted.reads` *Count the number of everted reads for a set of BAM files.*

Description

This is the ExomeDepth high level function that takes a GenomicRanges object, a list of indexed/sorted BAM files, and compute the number of everted reads in each of the defined bins.

Usage

```
count.everted.reads(bed.frame = NULL, bed.file = NULL,
                    bam.files, index.files = bam.files,
                    min.mapq = 20, include.chr = FALSE)
```

Arguments

`bed.frame`	`data.frame` containing the definition of the regions. The first three columns must be chromosome, start, end.
`bed.file`	`character` file name. Target BED file with the definition of the regions. This file will only be used if no bed.frame argument is provided. No headers are assumed so remove them if they exist. Either a bed.file or a bed.frame must be provided for this function to run.
`bam.files`	`character`, list of BAM files to extract read count data from.
`index.files`	Optional `character` argument with the list of indexes for the BAM files, without the '.bai' suffix. If the indexes are simply obtained by adding .bai to the BAM files, this argument does not need to be specified.
`min.mapq`	`numeric`, minimum mapping quality to include a read.
`include.chr`	`logical`, if set to TRUE, this function will remove the string 'chr' from the chromosome names of the target BED file.

Details

Everted reads are characteristic of the presence of duplications in a BAM files. This routine will parse a BAM files and the suggested use is to provide relatively large bins (for example gene based, and ExomeDepth has a genes.hg19 object that is appropriate for this) to flag the genes that contain such reads suggestive of a duplication. A manual check of the data using IGV is recommended to confirm that these reads are all located in the same DNA region, which would confirm the presence of a copy number variant.

Value

A data frame that contains the region and the number of identified reads in each bin.

Note

This function calls a lower level function called XXX that works on each single BAM file.

Author(s)

Vincent Plagnol

References

Computational methods for discovering structural variation with next-generation sequencing, Medvedev P, Stanciu M, Brudno M., Nature Methods 2009

See Also

getBAMCounts

Examples

```
## Not run:   test <- count.everted.reads (bed.frame = genes.hg19,
  bed.file = NULL,
  bam.files = bam.files,
  min.mapq = 20,
  include.chr = FALSE)

## End(Not run)
```

---

countBam.everted *Counts everted reads from a single BAM file*

Description

This is a utility function that is called by the higher level count.everted.reads. It processes each BAM file individually to generate the count data.

Usage

```
countBam.everted(bam.file, granges, index = bam.file, min.mapq = 1)
```

Arguments

bam.file	BAM file that needs to be parsed
granges	Genomic Ranges object with the location of the bins for which we want to count the everted reads.
index	Index for the BAM files.
min.mapq	Minimum mapping quality to include reads.

Details

Most users will not use this function, and it will only be called by the higher level count.everted.reads. Nevertheless it may be useful on its own in some cases.

Value

A list with the number of reads in each bin.

Author(s)

Vincent Plagnol

See Also

count.everted.reads

`countBamInGRanges.exomeDepth`

Compute read count data from BAM files.

Description

Parses a BAM file and count reads that are located within a target region defined by a GenomicRanges object.

Usage

```
countBamInGRanges.exomeDepth(bam.file, index = bam.file, granges,
min.mapq = 1, read.width = 1, force.single.end = FALSE)
```

Arguments

`bam.file`	BAM file to be parsed
`index`	Index of the BAM file, without the '.bai' suffix.
`granges`	Genomic ranges object defining the bins
`min.mapq`	Minimum read mapping quality (Phred scaled).
`read.width`	For single end reads, an estimate of the frament size. For paired reads, the fragment size can be directly computed from the paired alignment and this value is ignored.
`force.single.end`	Treat all paired reads as single end. It is generally a bad idea but may be useful in some rare situations. Only use this option if you really know what you are doing.

Details

Largely derived from its equivalent function in the exomeCopy package.

Value

A GRanges object with count data.

ExomeCount *Example dataset for ExomeDepth*

Description

An example dataset of 4 exome samples, chromosome 1 only.

Usage

```
data(ExomeCount)
```

Format

A data frame with 25592 observations on the following 9 variables.

`chromosome` a character vector with chromosome names (only chromosome 1 in that case)

`start` Start of the exons

`end` End of the exons

`exons` a character vector with the exon names.

`camfid.032KA_sorted_unique.bam` a numeric vector of read count data.

`camfid.033ahw_sorted_unique.bam` a numeric vector of read count data.

`camfid.035if_sorted_unique.bam` a numeric vector of read count data.

`camfid.034pc_sorted_unique.bam` a numeric vector of read count data.

`GC` a numeric vector with the GC content.

Source

Dataset generated in collaboration with Sergey Nejentsev, University of Cambridge.

References

Paper currently under review.

ExomeDepth-class *Class* ExomeDepth

Description

A class to hold the read count data that is used by ExomeDepth to call CNVs.

Objects from the Class

Objects can be created by calls of the form `new("ExomeDepth", data = NULL, test, reference, formula = 'cbind(1', subset.for.speed = NULL)`. `data` is optional and is only used if the `formula` argument refers to covariates (in which case these covariates must be included in the data frame). `test` and `reference` refer to the read count data for the test and reference samples. Creating a ExomeDepth object will automatically fit the beta-binomial model (using routines from the `aod` package) and compute the likelihood for the three copy number states (normal, deletion and duplication).

Slots

`test`: `numeric`, read count data for the test sample.

`reference`: `numeric`, read count data for the reference sample (usually a combination of samples).

`formula`: `character`, a character string describing the linear model linking test and reference. Typically this would be `cbind(test, reference) ~ 1`.

`expected`: The expected read count data for the test sample assuming normal copy number.

`phi`: The over-dispersion parameter of the binomial model. See the `aod` package for more details.

`likelihood`: A matrix of likelihood values, one column per copy number (deletion, normal, duplication).

`annotations`: A `data.frame` specifying the chromosome, start and end for the bins used in the read count computation.

`CNV.calls`: A `data.frame` describing the output of the CNV calling procedure.

Methods

CallCNVs `signature(x = "ExomeDepth",transition.probability = "numeric", chromosome = "factor", start = "numeric", end = "numeric", name = "character")` : Uses the pre-computed likelihood values and fits a hidden Markov Chain to the data to generated merged CNV calls.

AddAnnotations `signature(object = "ExomeDepth", name = "character", chromosome = "factor", start = "numeric")`: This method is unlikely to be directly used but it can include the exon names, chromosome, start, end into the ExomeDepth object.

TestCNV `signature(x = "ExomeDepth", chromosome = "factor", start = "numeric", end = "numeric", type = "character")`: type must be either deletion of duplication. This function takes an ExomeDepth object and returns the Bayes factor in favor of a CNV at the specified location.

Author(s)

Vincent Plagnol

References

Paper recently submitted

See Also

`select.reference.set` `CallCNVs` `aod`

Examples

```
showClass("ExomeDepth")
```

`exons.hg19` *Positions of exons on build hg19 of the human genome*

Description

Exon position extracted from the ensembl database version 71.

Usage

```
data(exons.hg19)
```

Format

A data frame with 192,379 observations on the following 4 variables.

`chromosome` a factor with levels `1` `10` `11` `12` `13` `14` `15` `16` `17` `18` `19` `2` `20` `21` `22` `3` `4` `5` `6` `7` `8` `9`

`start` a numeric vector

`end` a numeric vector

`name` A character vector of names for the exon(s)

Source

Ensemble database version 71.

exons.hg19.X — *Positions of exons on build hg19 of the human genome and on chromosome X*

Description

Exon position extracted from the ensembl database version 61 and on chromosome X only.

Usage

```
data(exons.hg19)
```

Format

A data frame of exons with the following 4 variables.

`chromosome` a factor with levels `X Y`

`start` a numeric vector

`end` a numeric vector

`name` A character vector of names for the exons

Source

Ensemble database version 71.

genes.hg19 — *Positions of genes on build hg19 of the human genome*

Description

Exon position extracted from the ensembl database version 71.

Usage

```
data(genes.hg19)
```

Format

A data frame with 18,033 observations on the following 4 variables.

`chromosome` a factor with levels `1 10 11 12 13 14 15 16 17 18 19 2 20 21 22 3 4 5 6 7 8 9`

`start` a numeric vector

`end` a numeric vector

`name` A character vector of names for the gene(s)

Source

Ensemble database version 71.

`get.power.betabinom` *Estimate the power to compare two beta-binomial distributions.*

Description

A power study useful in the context of ExomeDepth.

Usage

```
get.power.betabinom(size,
my.phi,
my.p,
my.alt.p,
theory = FALSE,
frequentist = FALSE,
limit = FALSE)
```

Arguments

`size`	Number of samples from the beta-binomial distribution.
`my.phi`	Over-dispersion parameter.
`my.p`	Expected p under the null.
`my.alt.p`	Expected p under the alternative.
`theory`	`logical`, should a theoretical limit (large sample size) be used? Defaults to FALSE.
`frequentist`	`logical`, should a frequentist version be used? Defaults to FALSE.
`limit`	`logical`, should another large sample size limit be used? Defaults to FALSE.

Value

An expected Bayes factor.

Author(s)

Vincent Plagnol

getBamCounts *Get count data for multiple exomes*

Description

Essentially a wrapper for the accessory function countBamInGRanges which only considers a single BAM file at a time.

Usage

```
getBamCounts(bed.frame = NULL,
bed.file = NULL,
bam.files,
index.files = bam.files,
min.mapq = 20,
read.width = 300,
include.chr = FALSE,
referenceFasta = NULL,
force.single.end = FALSE)
```

Arguments

bed.frame	data.frame containing the definition of the regions. The first three columns must be chromosome, start, end.
bed.file	character file name. Target BED file with the definition of the regions. This file will only be used if no bed.frame argument is provided. No headers are assumed so remove them if they exist. Either a bed.file or a bed.frame must be provided for this function to run.
bam.files	character, list of BAM files to extract read count data from.
index.files	Optional character argument with the list of indexes for the BAM files, without the '.bai' suffix. If the indexes are simply obtained by adding .bai to the BAM files, this argument does not need to be specified.
min.mapq	numeric, minimum mapping quality to include a read.
read.width	numeric, maximum distance between the side of the target region and the middle of the paired read to include the paired read into that region.
include.chr	logical, if set to TRUE, this function will remove the string 'chr' from the chromosome names of the target BED file.
referenceFasta	character, file name for the reference genome in fasta format. If available, GC content will be computed and added to the output.
force.single.end	Treat all paired reads as single end. It is generally a bad idea but may be useful in some rare situations. Do not use this option if you do not have a very good reason and you know what you are doing. The main issue is that paired reads mapping to the same region will be counted twice.

Details

This function is largely a copy of a similar one available in the exomeCopy package.

Value

A GenomicRanges object that stores the read count data for the BAM files listed as argument.

Author(s)

Vincent Plagnol

References

exomeCopy R package.

Examples

```
## Not run:
load(exons.hg19)

my.counts <- getBamCounts(bed.frame = exonpos,
                          bam.files = my.bam,
                          referenceFasta = 'human_g1k_v37.fasta')

## End(Not run)
```

---

initialize-methods ~~ *Methods for Function* initialize ~~

Description

~~ Methods for function initialize ~~

Methods

signature(.Object = "ExomeDepth")

---

plot-methods ~~ *Methods for Function* plot ~~

Description

~~ Methods for function plot ~~

Methods

signature(x = "ExomeDepth", sequence, count.threshold = 100, xlim = NULL, ylim = NULL, ylab = 'Obse

Plot the read depth data for a CNV call.

qbetabinom *Quantile for betabin function*

Description

Quantile function for the betabinomal distribution using the p/phi parameterisation.

Usage

```
qbetabinom(p, size, phi, prob)
```

Arguments

p	Point of the distribution from which one is looking for the quantile
size	Sample size of the random variable
phi	Over-dispersion parameter
prob	Mean probability of the binomial distribution

Details

Filling a gap in the VGAM package.

Value

A real number corresponding to the quantile p.

Author(s)

Vincent Plagnol

See Also

VGAM R package.

qbetabinom.ab *Quantile function for the beta-binomial distribution*

Description

Standard qbetabinomial.ab function which is missing from the VGAM package.

Usage

```
qbetabinom.ab(p, size, shape1, shape2)
```

Arguments

p	Mean value of the beta-binomial distribution.
size	Size of the beta-binomial.
shape1	First parameter of the beta distribution for p.
shape2	Second parameter of the beta distribution for p.

Value

A quantile of the distribution.

See Also

VGAM package.

---

select.reference.set *Combine multiple samples to optimize the reference set in order to maximise the power to detect CNV.*

Description

The power to detect copy number variant (CNVs) from targeted sequence data can be maximised if the most appropriate set of sequences is used as reference. This function is designed to combine multiple reference exomes in order to build the best reference set.

Usage

```
select.reference.set(test.counts, reference.counts, bin.length = NULL,
n.bins.reduced = 0, data = NULL, formula = 'cbind(test, reference) ~ 1',
phi.bins = 1)
```

Arguments

test.counts	Read count data for the test sample (numeric, typically a vector of integer values).
reference.counts	Matrix of read count data for a set of additional samples that can be used as a comparison point for the test sample.
bin.length	Length (in bp) of each of the regions (often exons, but not necessarily) that were used to compute the read count data (i.e. what is provided in the argument test.counts of this function). If not provided all bins are assumed to have equal length.
n.bins.reduced	This optimization function can be slow when applied genome-wide. For the purpose of building the reference sample, it is not necessary to use the full data. The number provided by this argument specifies the number of regions (typically exons) that will be sub-sampled (using a grid) to optimise the referenceset. I find that 10,000 is largely sufficient for exome data.

data	Defaults to NULL: A data frame of covariates that can be included in the model.
formula	Defaults to 'cbind(test, reference) ~ 1'. This formula will be used to fit the read count data. Covariates present in the data frame (for example GC content) can be included in the right hand side of the equation'. If covariates are provided they must be provided as arguments (in the data frame "data").
phi.bins	Numeric integer (typically 1, 2, or 3) that specifies the number of windows where the over-dispersion parameter phi can vary. It defaults to 1, i.e. a single over-dispersion parameter, independently of read depth.

Value

reference.choice	character: list of samples selected as optimum reference set.
summary.stats	A data frame summarizing the output of this computation, including expected Bayes factor, Rs statistic (see reference for explanation) for multiple choices of reference set.

Author(s)

Vincent Plagnol

References

Key paper currently under review.

---

show-methods *~~ Methods for Function* show *~~*

Description

~~ Methods for function show ~~

Methods

```
signature(object = "ExomeDepth")
```

somatic.CNV.call *Call somatic variants between healthy and disease tissues.*

Description

Use read depth data from targeted sequencing experiments to call CNV between a tumor and matched healthy tissue.

Usage

```
somatic.CNV.call(normal, tumor, prop.tumor = 1, chromosome, start, end, names)
```

Arguments

normal	Read count data (numeric vector) for the normal tissue.
tumor	Read count data (numeric vector) for the tumor.
prop.tumor	Proportion of the tumour DNA in the tumour sample (between 0 and 1, and less than 1 if there is normal tissue in the tumor sample).
chromosome	Chromosome information for the bins.
start	Start position of each bin (typically in bp).
end	End position of each bin.
names	Names for each bin (tyically exon names but any way to track the bins will do).

Details

Experimental function at this stage.

Value

An ExomeDepth object with CNV calls.

Note

Absolutely experimental, not the main function from the package.

Author(s)

V. Plagnol

References

A robust model for read count data in exome sequencing experiments and implications for copy number variant calling, Bioinformatics, In Press

`TestCNV` *Computes the Bayes Factor in favour of a CNV defined by position and type.*

Description

Test what evidence supports the presence of a CNV in an ExomeDepth object.

Usage

```
TestCNV(x, chromosome, start, end, type)
```

Arguments

`x`	ExomeDepth object containing the likelihood information.
`chromosome`	Chromosome data (`factor`)
`start`	Start of the putative CNV
`end`	End of the putative CNV
`type`	`character`, Should be either 'deletion' or 'duplication'

Value

A Bayes factor assessing the evidence in favoor of the CNV.

`viterbi.hmm` *Computes the Viterbi path for a hidden markov model*

Description

Estimates the most likely path for a hidden Markov Chain using the maximum likelihood Viterbi algorithm.

Usage

```
viterbi.hmm(transitions, loglikelihood, positions, expected.CNV.length)
```

Arguments

`transitions`	Transition matrix
`loglikelihood`	numeric matrix containing the loglikelihood of the data under the possible states
`positions`	Positions of the exons
`expected.CNV.length`	Expected length of CNV calls, which has an impact on the transition matrix between CNV states.

Details

Standard forward-backward Viterbi algorithm using a precomputed matrix of likelihoods.

Value

comp1	Description of 'comp1'
comp2	Description of 'comp2'

Author(s)

Vincent Plagnol

Index

∗Topic **\textasciitilde\textasciitilde other possible keyword(s) \textasciitilde\textasciitilde**
  initialize-methods, 15
  plot-methods, 15
  show-methods, 18
∗Topic **classes**
  ExomeDepth-class, 10
∗Topic **datasets**
  Conrad.hg19.common.CNVs, 5
  ExomeCount, 9
  exons.hg19, 11
  exons.hg19.X, 12
  genes.hg19, 12
∗Topic **methods**
  AnnotateExtra-methods, 4
  initialize-methods, 15
  plot-methods, 15
  show-methods, 18
∗Topic **package**
  ExomeDepth-package, 2

AddAnnotations,ExomeDepth-method *(*ExomeDepth-class*)*, 10
AnnotateExtra, 3
AnnotateExtra,ExomeDepth-method *(*AnnotateExtra-methods*)*, 4
AnnotateExtra-methods, 4

CallCNVs, 4
CallCNVs,ExomeDepth-method *(*ExomeDepth-class*)*, 10
Conrad.hg19.common.CNVs, 5
count.everted.reads, 6
countBam.everted, 7
countBamInGRanges.exomeDepth, 8

ExomeCount, 9
ExomeDepth *(*ExomeDepth-package*)*, 2
ExomeDepth-class, 10
ExomeDepth-package, 2
exons.hg19, 11
exons.hg19.X, 12

genes.hg19, 12
get.power.betabinom, 13
getBamCounts, 14

initialize,ExomeDepth-method *(*initialize-methods*)*, 15
initialize-methods, 15

plot,ANY-method *(*plot-methods*)*, 15
plot,ExomeDepth-method *(*plot-methods*)*, 15
plot-methods, 15

qbetabinom, 16
qbetabinom.ab, 16

select.reference.set, 17
show,ExomeDepth-method *(*show-methods*)*, 18
show-methods, 18
somatic.CNV.call, 19

TestCNV, 20
TestCNV,ExomeDepth-method *(*ExomeDepth-class*)*, 10

viterbi.hmm, 20