Package 'oncoscanR'

October 21, 2019

Type Package

Title Meta-analyses of CNV segments

Version 0.1.0

Description Taking copy number segments as input (either ChAS export file or BED-like), identify chromosome arms that are globally altered and computes various scores (LST, TD, TDplus, LOH).

License MIT + file LICENSE

Date 2019-09-13

URL https://github.com/yannchristinat/oncoscanR

Encoding UTF-8

LazyData true

biocViews

Imports readr, S4Vectors, magrittr

RoxygenNote 6.1.1

Depends R (>= 3.3.0), IRanges (>= 2.6.1), GenomicRanges (>= 1.24.3)

Suggests testthat (>= 2.1.0), jsonlite

NeedsCompilation no

Author Yann Christinat [aut, cre]

Maintainer Yann Christinat <yann.christinat@hcuge.ch>

R topics documented:

`adjust_loh` *Trim LOH segments with respect to loss segments.*

Description

Trim LOH segments with respect to loss segments.

Usage

```
adjust_loh(segments)
```

Arguments

`segments`	A `GRanges` object containing the segments, their copy number and copy number types.

Details

LOH segments completely contained within (or equal to) a copy loss segment are deleted. LOH segments partially overlapping (on one end only) with a copy loss segment are trimmed to remove the overlap. If a copy loss segment is completely contained within (but not equal to) a LOH segment, then nothing is done; the overlap remains.

Value

A `GRanges` object containing the cleaned segments, their copy number and copy number types.

Examples

```
segs.adj <- adjust_loh(segs.chas_example)
```

---

armlevel_alt *Get all globally-altered chromosome arms.*

Description

Get all globally-altered chromosome arms.

Usage

```
armlevel_alt(segments, kit.coverage, threshold = 0.8)
```

Arguments

segments	A GRanges object containing the segments.
kit.coverage	A GRanges object containing the regions covered on each chromosome arm.
threshold	The minimum percentage of the arm to be considered as globally altered. Defaults to 80%.

Details

By default uses the sum of all alterations and set the arm as globally altered if \>80% of the arm is altered. Does not account for alteration type and copy number. Will run the function trim_to_coverage on the segments.

Value

A list of globally-altered chromosome arms with the percentage of arm altered.

Examples

```
arms <- armlevel_alt(segs.chas_example, oncoscan_na33.cov, 0.9)
```

`cntype.amp` *copy number type "Amplification" (3 additional copies or more).*

Description

copy number type "Amplification" (3 additional copies or more).

Usage

```
cntype.amp
```

Format

character

Source

```
cntype.amp <- "Amplification"
```

`cntype.gain` *copy number type "Gain" (one or more additional copies). In some contextes can be defined as the gain of 1-2 extra copies but not more.*

Description

copy number type "Gain" (one or more additional copies). In some contextes can be defined as the gain of 1-2 extra copies but not more.

Usage

```
cntype.gain
```

Format

character

Source

```
cntype.gain <- "Gain"
```

cntype.hetloss — *copy number type "Heterozygous loss" (loss of one of the two copies).*

Description

copy number type "Heterozygous loss" (loss of one of the two copies).

Usage

```
cntype.hetloss
```

Format

character

Source

```
cntype.hetloss <- "Heterozygous loss"
```

cntype.homloss — *copy number type "Homozygous loss" (loss of the two copies or only one on sexual chromosomes if the subject is male).*

Description

copy number type "Homozygous loss" (loss of the two copies or only one on sexual chromosomes if the subject is male).

Usage

```
cntype.homloss
```

Format

character

Source

```
cntype.homloss <- "Homozygous loss"
```

cntype.loh — *copy number type "LOH" (copy neutral loss of heterozygosity).*

Description

copy number type "LOH" (copy neutral loss of heterozygosity).

Usage

```
cntype.loh
```

Format

character

Source

```
cntype.loh <- "LOH"
```

cntype.loss — *copy number type "Loss" (loss of one or more copies; depends on gender for the sexual chromosomes).*

Description

copy number type "Loss" (loss of one or more copies; depends on gender for the sexual chromosomes).

Usage

```
cntype.loss
```

Format

character

Source

```
cntype.loss <- "Loss"
```

cntype.strongamp *copy number type "Strong amplification" (8 additional copies or more).*

Description

copy number type "Strong amplification" (8 additional copies or more).

Usage

```
cntype.strongamp
```

Format

character

Source

```
cntype.strongamp <- "Strong amplification"
```

cntype.weakamp *copy number type "Weak amplification" (3 to 7 additional copies).*

Description

copy number type "Weak amplification" (3 to 7 additional copies).

Usage

```
cntype.weakamp
```

Format

character

Source

```
cntype.weakamp <- "Weak amplification"
```

get_cn_subtype *Get the detailed copy number types.*

Description

Get the detailed copy number types.

Usage

```
get_cn_subtype(segments, gender)
```

Arguments

segments	A GRanges object containing the segments, their copy number (field cn) and their copy number types (field cn.type). The cn.type is expected to be either "Gain", "Loss" or "LOH".
gender	Character to indicate whether the sample is male ("M") or female ("F"). If NULL then the field cn.type is set to NA for the chromosomes X and Y.

Details

The copy number subtypes are defined as follow: - "Gain": 1-2 extra copies - "Weak amplification": 3-7 extra copies - "Strong amplification": 8 or more extra copies - "Heterozygote loss": Loss of one copy out of two - "Homozygote loss": Loss of all copies - "LOH": copy-neutral loss of one parental allele

Value

A list of copy number types (one for each segment). Raises an error if the cn.type, cn and gender are not compatible.

Examples

```
subtypes <- get_cn_subtype(segs.chas_example, 'F')
```

get_oncoscan_coverage_from_probes *Load the oncoscan annotation file and infer the covered regions from it.*

Description

Load the oncoscan annotation file and infer the covered regions from it.

Usage

```
get_oncoscan_coverage_from_probes(filename = system.file("extdata",
  "OncoScan.na33.r1.annot.csv.zip", package = "oncoscanR"))
```

Arguments

filename	Path to the ChAS annotation file (either compressed or not). Defaults to the Oncoscan na33.r1 file present in the package.

Details

Expects the following columns from the annotation file (as the first line of non-comment): - "Chromosome": chromosome name - "Physical Position": genomic position of the SNP - "Cytoband": cytoband name (e.g. 3q22.1)

Value

A GRanges object containing the regions covered on each chromosome arm. If the file does not respect the format specifications, then an error is raised.

Examples

```
oncoscan_na33.cov <- get_oncoscan_coverage_from_probes()
```

---

is.cn_segment *Test if the object is a CNV segment*

Description

Test if the object is a CNV segment

Usage

```
is.cn_segment(obj, raise_error = TRUE)
```

Arguments

obj	Object to test
raise_error	Boolean. If TRUE then raises an error if the test fails.

Value

Boolean. TRUE if the object is of class GRanges with fields cn and cn.type.

Examples

```
library(GenomicRanges)
library(IRanges)
s1 <- GRanges(seqnames = '1p', ranges = IRanges(start = 1, end = 10),
              cn = 6, cn.type = cntype.gain, cn.subtype = cntype.weakamp)
is.cn_segment(s1)
s2 <- GRanges(seqnames = '1p', ranges = IRanges(start = 1, end = 10),
              cn = 6)
is.cn_segment(s2, raise_error = FALSE)
```

`load_bed` *Load a CNV file in a BED-like format.*

Description

Load a CNV file in a BED-like format.

Usage

```
load_bed(filename, gender)
```

Arguments

`filename`	Path to the CNV file.
`gender`	Character to indicate whether the sample is male ("M") or female ("F").

Details

The file is expected to contain the following columns: chromosome name, segment start, segment end, copy number. An fifth column with the copy number type is optional and if present should contain the values "Gain", "Loss" or "LOH". If the gender is not specified, then the segments on the sexual chromosomes (X and Y) are dropped as the copy number type cannot be determined for those.

Value

A `GRanges` object containing the segments, their copy number and copy number types. If the file contains twice the same segment or does not respect the format specifications, then an error is raised.

Examples

```
segs.filename <- system.file("extdata", "cnv_example.bed", package = "oncoscanR")
segs.cnv_example <- load_bed(segs.filename, "F")
```

load_chas *Load a ChAS text export file.*

Description

Load a ChAS text export file.

Usage

```
load_chas(filename, kit.coverage)
```

Arguments

filename	Path to the ChAS file.
kit.coverage	A GRanges object containing the regions covered on each chromosome arm by the kit.

Details

The ChAS file is expected to have the following column names: "CN State" (number or empty), "Type" (expected value: "Gain", "Loss" or "LOH") and "Full Location" (in the format "chr:start-end").

Value

A GRanges object containing the segments, their copy number (field cn), their copy number types (field cn.type). cn.type contains either "Gain", "Loss" or "LOH". If the file contains twice the same segment or does not respect the format specifications, then an error is raised. NB. The chromosome name is in the format "1" and not "chr1" and will be transformed if needed.

Examples

```
segs.filename <- system.file("extdata", "chas_example.txt", package = "oncoscanR")
segs.chas_example <- load_chas(segs.filename, oncoscan_na33.cov)
```

merge_segments *Merge segments with respect to the kit resolution and the copy number.*

Description

Merge segments with respect to the kit resolution and the copy number.

Usage

```
merge_segments(segments, kit.resolution = 300)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.
kit.resolution	Number >0 indicating the minimum segment size detectable by the technique (in kilobases). Defaults to the Oncoscan assay resolution outside of cancer genes: 300Kb.

Details

If two segments are at a distance smaller than the resolution, then the segments are merged if the share the same cn value. Note that the function does not look at the copy number type or subtype but only at the actual copy number to decide whether segments can be merged.

Value

A GRanges object containing the cleaned segments, their copy number and copy number types.

Examples

```
segs.merged <- merge_segments(segs.chas_example)
segs.merged_50k <- merge_segments(segs.chas_example, 50)
```

---

oncoscan_na33.cov *Arm coverage of the Oncoscan na33.r1 assay*

Description

Arm coverage of the Oncoscan na33.r1 assay

Usage

```
oncoscan_na33.cov
```

Format

A GRanges object containing the regions covered on each chromosome arm. Arms not covered by the kit are absent from the object.

Source

Processing of the file OncoScan.na33.r1.annot.csv (obtained on Affymetrix website) by the function `oncoscan_na33.cov <- get_oncoscan_coverage_from_probes()`.

Examples

```
## Not run:
 oncoscan_na33.cov

## End(Not run)
```

`ploidy` *Compute the total number of chromosomes (ploidy).*

Description

Compute the total number of chromosomes (ploidy).

Usage

```
ploidy(segments, kit.coverage)
```

Arguments

`segments`	A `GRanges` object containing the segments, their copy number and copy number types.
`kit.coverage`	A `GRanges` object containing the regions covered on each chromosome arm.

Details

For each chromosome get the maximum copy number variation (positive or negative) that cover 80 of each arm of the chromosome. If an arm is not part of the kit coverage, then it is assumed that the other arm is equally altered.

Value

the total number of chromosomes

Examples

```
ploidy(segs.chas_example, kit.coverage = oncoscan_na33.cov)
```

`prune_by_size` *Remove segments smaller than the kit resolution.*

Description

Remove segments smaller than the kit resolution.

Usage

```
prune_by_size(segments, threshold = 300)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.
threshold	Number indicating the minimum segment size to be kept (in kilobases). Defaults to the Oncoscan assay resolution outside of cancer genes: 300Kb.

Value

A GRanges object containing the cleaned segments, their copy number and copy number types.

Examples

```
segs.300k <- prune_by_size(segs.chas_example)
segs.50k <- prune_by_size(segs.chas_example, 50)
```

---

same_segments *Compare two CNV segments and test if equal*

Description

Compare two CNV segments and test if equal

Usage

```
same_segments(x, y)
```

Arguments

x	A GRanges object one segment, with optional fields cn, cn.type and cn.subtype
y	A GRanges object one segment, with optional fields cn, cn.type and cn.subtype

Value

A boolean. TRUE if the segments' chromosome/arm, start/end and CN tags match.

Examples

```
library(GenomicRanges)
library(IRanges)
s1 <- GRanges(seqnames = '1p', ranges = IRanges(start = 1, end = 10),
              cn = 5, cn.type = cntype.gain, cn.subtype = cntype.weakamp)
s2 <- GRanges(seqnames = '1p', ranges = IRanges(start = 1, end = 10),
              cn = 6, cn.type = cntype.gain, cn.subtype = cntype.weakamp)
same_segments(s1,s2)
```

same_segmentsets *Small function to test if two sets have the same CN segments*

Description

Small function to test if two sets have the same CN segments

Usage

```
same_segmentsets(grA, grB)
```

Arguments

grA	A GRanges object with fields cn and cn.type
grB	A GRanges object with fields cn and cn.type

Value

TRUE if the two sets contain exactly the same CN segments

score_loh *Compute the number HR deficiency-associated LOH regions.*

Description

Compute the number HR deficiency-associated LOH regions.

Usage

```
score_loh(segments, kit.coverage, armlevel.loh, armlevel.hetloss)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.
kit.coverage	A GRanges object containing the regions covered on each chromosome arm.
armlevel.loh	A list of arms with global/arm-level LOH alteration.
armlevel.hetloss	A list of arms with global/arm-level heterozygous losses.

Details

Procedure based on the paper from Abkevich et al., Br J Cancer 2012 (PMID: 23047548). All LOH segments larger than 15Mb but excluding chromosome with a global LOH alteration (to compute with the armlevel_alt function on LOH segments only). This score was linked to BRCA1/2-deficient tumors. Note that the function will merge overlapping or neighbor LOH segments (at a distance of 1bp).

Value

An integer representing the number of HRD-LOH regions.

Examples

```
armlevel.loh <- armlevel_alt(segs.chas_example[segs.chas_example$cn.type == cntype.loh],
                             kit.coverage = oncoscan_na33.cov)
armlevel.hetloss <- armlevel_alt(segs.chas_example[segs.chas_example$cn.subtype == cntype.hetloss],
                             kit.coverage = oncoscan_na33.cov)
score_loh(segs.chas_example, oncoscan_na33.cov, names(armlevel.loh), names(armlevel.hetloss))
```

---

score_lst *Compute the number of Large-scale State Transitions (LSTs).*

Description

Compute the number of Large-scale State Transitions (LSTs).

Usage

```
score_lst(segments, kit.coverage)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.
kit.coverage	A GRanges object containing the regions covered on each chromosome arm.

Details

Procedure based on the paper from Popova et al, Can. Res. 2012 (PMID: 22933060). First segments smaller than 3Mb are removed, then segments are smoothed with respect to copy number at a distance of 3Mb. The number of LSTs is the number of breakpoints (breakpoints closer than 3Mb are merged) that have a segment larger or equal to 10Mb on each side. This score was linked to BRCA1/2-deficient tumors.

Value

An integer representing the number of LSTs.

Examples

```
score_lst(segs.chas_example, oncoscan_na33.cov)
```

score_td *Compute the number of large tandem duplication (TDplus).*

Description

Compute the number of large tandem duplication (TDplus).

Usage

```
score_td(segments)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.

Details

Procedure based on the paper from Popova et al., Cancer Res 2016 (PMID: 26787835). The TDplus score is defined as the number of regions larger than 1Mb but smaller or equal to 10Mb with a gain of one or two copies (cntype.gain in the field cn.subtype). This score was linked to CDK12-deficient tumors. They also identified as second category of tandem duplication whose size is smaller or equal than 1Mb and around 300Kb but could not link it to a phenotype. Note that due to its resolution the Oncoscan assay will most likely miss this second category. Nonetheless it is reported by the function.

Value

A list of integer containing the TDplus score ("TDplus") and the small TD score ("TD").

Examples

```
score_td(segs.chas_example)
```

segs.chas_example *Expected segments from loading the ChAS file 'chas_example.txt'.*

Description

Expected segments from loading the ChAS file 'chas_example.txt'.

Usage

```
segs.chas_example
```

Format

A GRanges object containing the segments, their copy number (field cn) and their copy number types (field cn.type).

Source

segs.filename <- system.file("extdata", "chas_example.txt", package = "oncoscanR") mykit.cov <- get_oncoscan_coverage_from_probes() segs.chas_example <- load_chas(segs.filename, kit.coverage = mykit.cov)

---

trim_to_coverage *Trim segments with respect to the kit's coverage.*

Description

Trim segments with respect to the kit's coverage.

Usage

```
trim_to_coverage(segments, kit.coverage)
```

Arguments

segments	A GRanges object containing the segments, their copy number and copy number types.
kit.coverage	A GRanges object containing the regions covered on each chromosome arm.

Details

All segments that are not entirely contained within the kit coverage will be trimmed to the coverage's limits.

Value

A GRanges object containing the cleaned segments, their copy number and copy number types.

Examples

```
segs.trimmed <- trim_to_coverage(segs.chas_example, oncoscan_na33.cov)
```

workflow_oncoscan.run *Run the standard workflow for Oncoscan ChAS files.*

Description

Run the standard workflow for Oncoscan ChAS files.

Usage

```
workflow_oncoscan.run(chas.fn, gender)
```

Arguments

chas.fn	Path to the text-export ChAS file
gender	Gender of the sample (M or F)

Details

Identifies the globally altered arms (\>=80% of arm altered), computes the LST, HR-LOH, TD+ and TD scores. The amplification is defined as a CN subtype cntype.weakamp or cntype.strongamp. An arm is gained if of CN type cntype.gain unless the arm is amplified.

Value

A list of lists with the following elements: armlevel = list(AMP= list of arms, GAIN= list of arms, LOSS= list of

Examples

```
segs.filename <- system.file("extdata", "chas_example.txt", package = "oncoscanR")
workflow_oncoscan.run(segs.filename, "M")
```

Index