Package 'CNViz'

October 25, 2024

Type Package

Title Copy Number Visualization

Version 1.13.0

Description CNViz takes probe, gene, and segment-level log2 copy number ratios and launches a Shiny app to visualize your sample's copy number profile. You can also integrate loss of heterozygosity (LOH) and single nucleotide variant (SNV) data.

Depends R (>= 4.0), shiny (>= 1.5.0)

Imports dplyr, stats, utils, grDevices, plotly, karyoploteR, CopyNumberPlots, GenomicRanges, magrittr, DT, scales, graphics

License Artistic-2.0

Encoding UTF-8

biocViews Visualization, CopyNumberVariation, Sequencing, DNASeq

RoxygenNote 7.1.1

Suggests rmarkdown, knitr

VignetteBuilder knitr

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

git_branch devel

git_last_commit c757067

git_last_commit_date 2024-04-30

Repository Bioconductor 3.20

Date/Publication 2024-10-25

Author Rebecca Greenblatt [aut, cre]

Maintainer Rebecca Greenblatt <rebecca.greenblatt@gmail.com>

Contents

`all_tcga2018_data` *Data from 2018 TCGA studies from cBioPortal*

Description

A dataset containing the study name and aggregated gene level copy number data

Usage

```
all_tcga2018_data
```

Format

A data frame with 14944 rows and 6 variables:

hugoGeneSymbol hugo gene symbol

Gain proportion of cohort with gain in this gene

Amplification proportion of cohort with amplification in this gene

ShallowDeletion proportion of cohort with shallow deletion in this gene

DeepDeletion proportion of cohort with deep deletion in this gene

study_name cancer type and sample size

Source

https://github.com/waldronlab/cBioPortalData See data-raw folder.

`cbio_studies` *Names of 2018 TCGA studies from cBioPortal*

Description

A dataset containing the names and studyIds of the 2018 TCGA studies from cBioPortal.

Usage

```
cbio_studies
```

Format

A data frame with 32 rows and 2 variables:

Cancer Name of diagnosis and sample size

studyId studyId that can be used in the cBioPortalData R package

Source

https://github.com/waldronlab/cBioPortalData See data-raw folder.

cytoband_data *Genomic locations of cytoband labels*

Description

A dataset containing the chr, start and end position for cytobands according to hg38.

Usage

```
cytoband_data
```

Format

A data frame with 863 rows and 6 variables:

chrom chromosome

chromStart start position

chromEnd end position

name cytoband name

gieStain color

color HEX color

Source

https://genome.ucsc.edu/cgi-bin/hgTables

gene_data *Gene data for vignette example*

Description

A dataset containing simulated gene data as sample input for launchCNViz

Usage

```
data(gene_data)
```

Format

A dataframe with 112 rows and 6 variables

chr chromosome

start start location

end end location

gene gene name

log2 log2 copy number ratio

weight weight given to log2 value

loh loss of heterozygosity

Source

Center for Personalized Diagnositics at the University of Pennsylvania

---

`launchCNViz` *Launches CNViz, a shiny app to visualize your sample's copy number data.*

Description

CNViz launches a shiny application to visualize your sample's copy number data. At least one of probe_data, gene_data, or segment_data must be supplied; sample_name, variant_data and meta_data are all optional. The more inputs supplied, the more informative the application will be. See the CNViz vignette for more information. Use the hg38 reference genome. CNViz only displays a single sample's data.

Usage

```
launchCNViz(
  sample_name = "sample",
  probe_data = data.frame(),
  gene_data = data.frame(),
  segment_data = data.frame(),
  variant_data = data.frame(),
  meta_data = data.frame()
)
```

Arguments

`sample_name`	A string with the ID/name of your sample.
`probe_data`	A dataframe or GRanges object containing probe-level data. If a dataframe, column names must include chr, gene, start, end, log2. chr/seqnames column should be formatted as 'chr1' through 'chrX', 'chrY'. start, end and log2 should be numeric. If a GRanges object, gene and log2 are metadata columns. Optional column/metadata: weight, where weight is numeric.
`gene_data`	A dataframe or GRanges object containing gene-level data - one row per gene. If a dataframe, column names must include chr, gene, start, end, log2. chr/seqnames column should be formatted as 'chr1' through 'chrX', 'chrY'. start, end and log2 should be numeric. If a GRanges object, gene and log2 are metadata columns. Optional columns/metadata: weight, loh; where weight is numeric and loh values are TRUE or FALSE.
`segment_data`	A dataframe or GRanges object containing segment-level data. If a dataframe, column names must include chr, start, end, log2. chr column should be formatted as 'chr1' through 'chrX', 'chrY'. start, end and log2 should be numeric. If a GRanges object, log2 is a metadata column. Optional column/metadata: loh; where loh values are TRUE or FALSE.
`variant_data`	A dataframe or VRanges object containg SNVs and short indels and columns of your choosing. If a dataframe, the only required columns are gene and mutation_id. Optional column: start; where start indicates the starting position of the mutation. If a VRanges object, make sure gene is one of the metadata columns,

so it can be tied to the gene or probe data; a mutation_id column can also be included, otherwise it will be constructed. Additional columns might include depth, allelic_fraction, ref, alt.

`meta_data` A dataframe containing your sample's metadata - columns of your choosing. Optional column: ploidy; ploidy will be rounded to the nearest whole number. Additional columns might include purity. This dataframe should only have one row.

Value

a Shiny application

Examples

```
probes <- data.frame(chr = c("chr1", "chr1", "chr4", "chr4", "chrX"),
gene = c("NOTCH2", "NOTCH2", "KIT", "TET2", "BTK"),
start = c(119922221, 119967406,54732072,105243553,101360541),
end = c(119922461,119967646,54732192,105243793,101360781),
log2 = c(-0.0832403,-0.0578757,0.2131540,-0.3189430,-0.7876670),
weight = c(0.684114, 0.681546,0.606129,0.682368,0.405772))
segments <- data.frame(chr = c("chr1","chr1", "chr4", "chr4", "chrX"),
start = c(1050069, 124932724,   1942322,  51743951,   1198732),
end = c(122026459, 246947668,  49712061, 188110779,  37098762),
log2 = c(1, 1, 1, 1, 0.5849625), loh = c(FALSE, FALSE, FALSE, TRUE, TRUE))
meta <- data.frame(purity = c(.5),
ploidy = c(2), sex = c("Female"))

launchCNViz(sample_name = "sample123", probe_data = probes,
segment_data = segments, meta_data = meta)
```

`meta_data` *Metadata for vignette example*

Description

A dataset containing simulated metadata as sample input for launchCNViz

Usage

```
data(meta_data)
```

Format

A dataframe with 1 rows and 2 variables

purity sample purity

ploidy tumor ploidy

Source

Center for Personalized Diagnositics at the University of Pennsylvania

probe_data *Probe data for vignette example*

Description

A dataset containing simulated probe data as sample input for launchCNViz

Usage

```
data(probe_data)
```

Format

A data frame with 2006 rows and 6 variables:

chr chromosome
start start location
end end location
gene gene name
log2 log2 copy number ratio
weight weight given to log2 value

Source

Center for Personalized Diagnositics at the University of Pennsylvania

segment_data *Segment data for vignette example*

Description

A dataset containing simulated segment data as sample input for launchCNViz

Usage

```
data(segment_data)
```

Format

A dataframe with 101 rows and 5 variables

chr chromosome
start start location
end end location
log2 log2 copy number ratio
loh loss of heterozygosity

Source

Center for Personalized Diagnositics at the University of Pennsylvania

variant_data *Variant data for vignette example*

Description

A dataset containing simulated SNV and indel data as sample input for launchCNViz

Usage

```
data(variant_data)
```

Format

A dataframe with 119 rows and 4 variables

gene gene name

mutation_id string with information about snv

depth read depth

start starting location

Source

Center for Personalized Diagnositics at the University of Pennsylvania

Index