Package ‘igvR’

May 6, 2024

Type Package

Title igvR: integrative genomics viewer

Version 1.25.0

Date 2023-10-09

Author Paul Shannon

Maintainer Arkadiusz Gladki <gladki.arkadiusz@gmail.com>

Depends R (>= 3.5.0), GenomicRanges, GenomicAlignments, BrowserViz (>= 2.17.1)

Imports methods, BiocGenerics, httpuv, utils, rtracklayer, VariantAnnotation, RColorBrewer, httr

Suggests RUnit, BiocStyle, knitr, rmarkdown, MotifDb, seqLogo

Description Access to igv.js, the Integrative Genomics Viewer running in a web browser.

URL https://gladkia.github.io/igvR/

License MIT + file LICENSE

LazyLoad yes

biocViews Visualization, ThirdPartyClient, GenomeBrowsers

Collate 'Track.R' 'igvAnnotationTrack.R' 'UCSCBedAnnotationTrack.R' 'DataFrameAnnotationTrack.R' 'VariantTrack.R' 'QuantitativeTrack.R' 'DataFrameQuantitativeTrack.R' 'UCSCBedGraphQuantitativeTrack.R' 'GRangesAnnotationTrack.R' 'GRangesQuantitativeTrack.R' 'GenomicAlignmentTrack.R' 'BedpeInteractionsTrack.R' 'RemoteAlignmentTrack.R' 'GWASTrack.R' 'GWASUrlTrack.R' 'GFF3Track.R' 'genomeSpec.R' 'igvR.R'

NeedsCompilation no

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.2.3

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

git_branch devel

git_last_commit dcc3575

git_last_commit_date 2024-04-30

Repository Bioconductor 3.20

Date/Publication 2024-05-06

Contents

`BedpeInteractionsTrack-class`

Constructor for BedpeInteractionsTrack

Description

`BedpeInteractionsTrack` creates an `IGV` track for two-location annotations

Usage

```
BedpeInteractionsTrack(
  trackName,
  table,
  color = "darkBlue",
  trackHeight = 50,
  displayMode = "EXPANDED",
  visibilityWindow = 1e+05
)
```

Arguments

`trackName`	A character string, used as track label by igv, we recommend unique names per track.
`table`	data.frame of 6 or more columns
`color`	A css color name (e.g., "red" or "#FF0000"
`trackHeight`	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
`displayMode`	"COLLAPSED", "SQUISHED" or "EXPANDED". Spelling and case must be precise.
`visibilityWindow`	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Value

A BedpeInteractionsTrack object

Examples

```
  #----------------------------
  #  first, from a local file
  #----------------------------

file <- system.file(package="igvR", "extdata", "sixColumn-demo1.bedpe")
tbl.bedpe <- read.table(file, sep="\t", as.is=TRUE, header=TRUE)
dim(tbl.bedpe)  #  32 6
track <- BedpeInteractionsTrack("bedpe-6", tbl.bedpe)

  #-----------------------------------------
  #  show the relevant portion of the genome
  #-----------------------------------------

shoulder <- 10000
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   setBrowserWindowTitle(igv, "Paired End Demo")
 roi <- with(tbl.bedpe, sprintf("%s:%d-%d", chrom1[1], min(start1)-shoulder, max(end2) + shoulder))
   showGenomicRegion(igv, roi)
   displayTrack(igv, track)
   }
```

---

currently.supported.stock.genomes

currently.supported.stock.genomes

Description

a helper function for mostly internal use, obtains the genome codes (e.g. 'hg38') supported by igv.js

Usage

```
currently.supported.stock.genomes(test = FALSE)
```

Arguments

test	logical

Value

an list of short genome codes, e.g., "hg38", "dm6", "tair10"

DataFrameAnnotationTrack-class

Constructor for DataFrameAnnotationTrack

Description

`DataFrameAnnotationTrack` creates an `IGV` track for bed objects imported using `rtracklayer`

Usage

```
DataFrameAnnotationTrack(
  trackName,
  annotation,
  color = "",
  displayMode = "SQUISHED",
  trackHeight = 50,
  expandedRowHeight = 30,
  squishedRowHeight = 15,
  maxRows = 500,
  searchable = FALSE,
  visibilityWindow = 1e+05
)
```

Arguments

`trackName`	A character string, used as track label by igv, we recommend unique names per track.
`annotation`	A base R `data.frame`
`color`	A CSS color name (e.g., "red" or "#FF0000"), leave as default empty string if supplying bed9 format with itemRgb.
`displayMode`	"COLLAPSED", "SQUISHED" or "EXPANDED". Spelling and case must be precise.
`trackHeight`	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
`expandedRowHeight`	Height of each row of features in "EXPANDED" mode.
`squishedRowHeight`	Height of each row of features in "SQUISHED" mode, for compact viewing.
`maxRows`	of features to display
`searchable`	If TRUE, labels on annotation elements may be used in search
`visibilityWindow`	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A DataFrameAnnotationTrack object

Examples

```
base.loc <- 88883100
tbl <- data.frame(chrom=rep("chr5", 3),
                  start=c(base.loc, base.loc+100, base.loc + 250),
                  end=c(base.loc + 50, base.loc+120, base.loc+290),
                  name=c("a", "b", "c"),
                  score=runif(3),
                  strand=rep("*", 3),
                  stringsAsFactors=FALSE)

track <- DataFrameAnnotationTrack("data.frame demo", tbl)

if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   setBrowserWindowTitle(igv, "DataFrameAnnotationTrack demo")
   displayTrack(igv, track)
   roi <- sprintf("%s:%d-%d", tbl$chrom[1], min(tbl$start)-100, max(tbl$start) + 100)
   showGenomicRegion(igv, roi)
   Sys.sleep(1)
   zoomOut(igv)
   }
```

DataFrameQuantitativeTrack-class

Constructor for DataFrameQuantitativeTrack

Description

`DataFrameQuantitativeTrack` creates and `IGV` track for bed objects imported using `rtracklayer`

Usage

```
DataFrameQuantitativeTrack(
  trackName,
  quantitativeData,
  color = "blue",
  trackHeight = 50,
  autoscale,
  min = NA_real_,
```

```
  max = NA_real_,
  visibilityWindow = 1e+05
)
```

Arguments

- `trackName` A character string, used as track label by igv, we recommend unique names per track.
- `quantitativeData` A base R `data.frame`
- `color` A CSS color name (e.g., "red" or "#FF0000")
- `trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
- `autoscale` Autoscale track to maximum value in view
- `min` Sets the minimum value for the data (y-axis) scale. Usually zero.
- `max` Sets the maximum value for the data (y-axis) scale. This value is ignored if autoscale is TRUE
- `visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A DataFrameQuantitativeTrack object

See Also

DataFrameAnnotationTrack

GRangesQuantitativeTrack

GRangesAnnotationTrack

DataFrameAnnotationTrack

DataFrameQuantitativeTrack

GRangesAnnotationTrack

GRangesQuantitativeTrack

GenomicAlignmentTrack

UCSCBedAnnotationTrack

UCSCBedGraphQuantitativeTrack

VariantTrack

igvAnnotationTrack

Examples

```
base.loc <- 88883100
tbl.blocks <- data.frame(chrom=rep("chr5", 3),
                  start=c(base.loc, base.loc+100, base.loc + 250),
                  end=c(base.loc + 50, base.loc+120, base.loc+290),
                  score=runif(3),
                  stringsAsFactors=FALSE)

track.blocks <- DataFrameQuantitativeTrack("blocks", tbl.blocks, autoscale=TRUE)

locs <- seq(from=base.loc, length.out=1000)
tbl.wig <- data.frame(chrom=rep("chr5", 1000), start=locs-1, end=locs,
                      score=runif(n=1000, min=-1, max=1))
track.wig <- DataFrameQuantitativeTrack("wig", tbl.wig, autoscale=FALSE,
                                        min=min(tbl.wig$score), max=max(tbl.wig$score),
                                        color="random")
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   setBrowserWindowTitle(igv, "DataFrameQuantitativeTrack demo")
   displayTrack(igv, track.blocks)
   roi <- sprintf("%s:%d-%d", tbl.blocks$chrom[1],
                  min(tbl.blocks$start)-1000, max(tbl.blocks$end) + 1000)
   showGenomicRegion(igv, roi)
   displayTrack(igv, track.wig)
   }
```

---

displayTrack,igvR-method

display the specified track in igv

Description

display the specified track in igv

Usage

```
## S4 method for signature 'igvR'
displayTrack(obj, track, deleteTracksOfSameName = TRUE)
```

Arguments

`obj`	An object of class igvR
`track`	An object of some terminal (leaf) subclass of Track
`deleteTracksOfSameName`	logical, default TRUE

Value

```
""
```

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   showGenomicRegion(igv, "MEF2C")
   base.loc <- 88883100
   tbl <- data.frame(chrom=rep("chr5", 3),
                     start=c(base.loc, base.loc+100, base.loc + 250),
                     end=c(base.loc + 50, base.loc+120, base.loc+290),
                     name=c("a", "b", "c"),
                     score=runif(3),
                     strand=rep("*", 3),
                     stringsAsFactors=FALSE)
   track <- DataFrameAnnotationTrack("dataframeTest", tbl, color="red",
                                     displayMode="EXPANDED")
   showGenomicRegion(igv, "chr5:88,881,962-88,885,045")
   displayTrack(igv, track)
   }
```

---

enableMotifLogoPopups,igvR-method

turn mottif log popups on or off

Description

Some tracks represent transcription factor binding sites, traditionally represented as a motif logo. use this method to enable that capability - which depends upon a properly constructed tbl.regions data.frame in a DataFrameAnnotationTrack: in addition to the usual (and mandatory) chrom, start, and end columns. To enable track-click popups over binding site, tbl.regions data.frame must also have a "name" column, which this format, by example: "MotifDb::Hsapiens-HOCOMOCOv10-MEF2C_HUMAN.H10MO.C" The first part of the name, "MotifDb::", tells igv you want to view the specified MotifDb pwm (motif logo, a matrix) when the binding site track element is clicked.

Limitations: This method only works after a call to setGenome(igv, "your genome of interest"). It only works with DataFrameAnnotationTrack objects (for now)

Usage

```
## S4 method for signature 'igvR'
enableMotifLogoPopups(obj, status)
```

Arguments

obj	An object of class igvR
status	TRUE or FALSE

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   new.region <- "chr5:88,882,214-88,884,364"
   showGenomicRegion(igv, new.region)
   base.loc <- 88883100
   element.names <- c("MotifDb::Hsapiens-HOCOMOCOv10-MEF2C_HUMAN.H10MO.C",
                      "fubar",
                      "MotifDb::Hsapiens-jaspar2018-MEF2C-MA0497.1")

   tbl.regions <- data.frame(chrom=rep("chr5", 3),
                             start=c(base.loc, base.loc+100, base.loc + 250),
                             end=c(base.loc + 50, base.loc+120, base.loc+290),
                             name=element.names,
                             score=round(runif(3), 2),
                             strand=rep("*", 3),
                             stringsAsFactors=FALSE)

  track <- DataFrameAnnotationTrack("dataframeTest", tbl.regions, color="darkGreen", displayMode="EXPANDED")
   displayTrack(igv, track)
   }
```

GenomicAlignmentTrack-class

Constructor for GenomicAlignmentTrack

Description

`GenomicAlignmentTrack` creates and `IGV` track for bed-like objects expressed as GRanges

Usage

```
GenomicAlignmentTrack(
  trackName,
  alignment,
  trackHeight = 50,
  visibilityWindow = 30000,
  color = "gray"
)
```

Arguments

`trackName`	A character string, used as track label by igv, we recommend unique names per track.
`alignment`	A GAlignments object

`trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)

`visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

`color` A character string, either a reconized color ("red") or a hex string ("#FF8532")

Details

Detailed description goes here

Value

A GenomicAlignmentTrack object

Examples

```
bamFile <- system.file(package="igvR", "extdata", "tumor.bam")
which <- GRanges(seqnames = "21", ranges = IRanges(10400126, 10400326))
param <- ScanBamParam(which=which, what = scanBamWhat())
x <- readGAlignments(bamFile, use.names=TRUE, param=param)
track <- GenomicAlignmentTrack("tumor", x)
```

---

getGenomicRegion,igvR-method

Obtain the chromosome and coordiates of the currently displayed genomic region.

Description

Some caution is needed with this function when called right after a lengthy browser operation - of which the main example is display a GenomicAlignmentTrack. igv.js does not at present allow us to delay the return from javascript pending completion of the track rendering. This does not pose much of a problem when you manipulate igv in the browser from R in normal interactive mode: simply wait for your last command to complete. But if you are running in programmatic mode, as we do when testing igvR, then caution is advised. See the test_displayAlignment function in unitTests/test_igvR.R.

Usage

```
## S4 method for signature 'igvR'
getGenomicRegion(obj)
```

Arguments

`obj` An object of class igvR

Value

A list with four fields: chrom (character), start(numeric), end(numeric), string(character)

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   showGenomicRegion(igv, "MEF2C")
   getGenomicRegion(igv)
    # list(chrom="chr5", start=88717241, end=88884466, string="chr5:88,717,241-88,884,466")
   }
```

---

getSupportedGenomes,igvR-method

Get the shorthand codes (eg, "hg38") for the genomes currently supported by our use of igv.js

---

Description

Get the shorthand codes (eg, "hg38") for the genomes currently supported by our use of igv.js

Usage

```
## S4 method for signature 'igvR'
getSupportedGenomes(obj)
```

Arguments

obj	An object of class igvR

Value

A character vector, the short form names of the currently supported genomes

Examples

```
if(interactive()){
   igv <- igvR()
   getSupportedGenomes(igv)
   }
```

getTrackNames,igvR-method

Get the names of all the tracks currently displayed in igv

Description

Get the names of all the tracks currently displayed in igv

Usage

```
## S4 method for signature 'igvR'
getTrackNames(obj)
```

Arguments

obj	An object of class igvR

Value

A character vector

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg19")
   getTrackNames(igv)     # "Gencode v18"
   }
```

GFF3Track-class

Constructor for GFF3Track

Description

GFF3Track creates an IGV track for 9-column gene annotation tables

Usage

```
GFF3Track(
  trackName,
  tbl.track = data.frame(),
  url = NA_character_,
  indexURL = NA_character_,
  trackColor = "black",
  colorByAttribute = NA_character_,
  colorTable = list(),
```

```
  displayMode,
  trackHeight,
  visibilityWindow
)
```

Arguments

- `trackName` A character string, used as track label by igv, we recommend unique names per track.
- `tbl.track` data.frame with 9 columns as defined at http://uswest.ensembl.org/info/website/upload/gff3.html
- `url` character the web location of a 9-column table, gzipped or not
- `indexURL` character the matching tabix index file
- `trackColor` character a recognized color name or RGB triple
- `colorByAttribute` a name from a column 9 attribute
- `colorTable` list which maps the colorByAttribute values to different colors
- `displayMode` "COLLAPSED", "SQUISHED" or "EXPANDED". Spelling and case must be precise.
- `trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
- `visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A GFF3Track object

Examples

```
tbl.gff3 <- read.table(system.file(package="igvR", "extdata", "GRCh38.94.NDUFS2.gff3"),
                       sep="\t", as.is=TRUE)
colnames(tbl.gff3) <- c("seqid", "source", "type", "start", "end", "score", "strand",
                        "phase", "attributes")
colors <- list("antisense" = "blueviolet",
               "protein_coding" = "blue",
               "retained_intron" = "rgb(0, 150, 150)",
               "processed_transcript" = "purple",
               "processed_pseudogene" = "#7fff00",
               "unprocessed_pseudogene" = "#d2691e",
               "default" = "black")
track <- GFF3Track("dataframe gff3", tbl.gff3, colorByAttribute="biotype", colorTable=colors,
             url=NA_character_, indexURL=NA_character_, displayMode="EXPANDED", trackHeight=200,
```

```
                    visibilityWindow=100000)

   # gff3 table structure is not bed-like. find chrom, start, end as seen below

roi <- with(tbl.gff3, sprintf("%s:%d-%d",
                              seqid[1],
                              as.integer(min(start)) - 1000,
                              as.integer(max(end)) + 1000))
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   setBrowserWindowTitle(igv, "GWAS demo")
   showGenomicRegion(igv, roi)
   displayTrack(igv, track)
   }
```

GRangesAnnotationTrack-class

Constructor for GRangesAnnotationTrack

Description

GRangesAnnotationTrack creates and IGV track for bed-like objects expressed as GRanges

Usage

```
GRangesAnnotationTrack(
  trackName,
  annotationData,
  color = "darkGrey",
  displayMode = "SQUISHED",
  trackHeight = 50,
  expandedRowHeight = 30,
  squishedRowHeight = 15,
  maxRows = 500,
  searchable = FALSE,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
annotationData	A GRanges object with optional name metadata column
color	A CSS color name (e.g., "red" or "#FF0000")
displayMode	"COLLAPSED", "SQUISHED" or "EXPANDED". Spelling and case must be precise.

`trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)

`expandedRowHeight`
Height of each row of features in "EXPANDED" mode.

`squishedRowHeight`
Height of each row of features in "SQUISHED" mode, for compact viewing.

`maxRows` of features to display

`searchable` If TRUE, labels on annotation elements may be used in search

`visibilityWindow`
Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A GRangesAnnotationTrack object

Examples

```
base.loc <- 88883100
tbl <- data.frame(chrom=rep("chr5", 3),
                  start=c(base.loc, base.loc+100, base.loc + 250),
                  end=c(base.loc + 50, base.loc+120, base.loc+290),
                  name=c("a", "b", "c"),
                  strand=rep("*", 3),
                  stringsAsFactors=FALSE)

gr <- GRanges(tbl)
track <- GRangesAnnotationTrack("GRangesQTest", gr)
```

GRangesQuantitativeTrack-class

Constructor for GRangesQuantitativeTrack

Description

`GRangesQuantitativeTrack` creates and `IGV` track for bed objects imported using `rtracklayer`

Usage

```
GRangesQuantitativeTrack(
  trackName,
  quantitativeData,
  color = "blue",
  trackHeight = 50,
  autoscale = TRUE,
  min = NA_real_,
  max = NA_real_,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
quantitativeData	A GRanges object with (at least) a "score" metadata column
color	A CSS color name (e.g., "red" or "#FF0000")
trackHeight	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
autoscale	Autoscale track to maximum value in view
min	Sets the minimum value for the data (y-axis) scale. Usually zero.
max	Sets the maximum value for the data (y-axis) scale. This value is ignored if autoscale is TRUE
visibilityWindow	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A GRangesQuantitativeTrack object

Examples

```
base.loc <- 88883100
tbl <- data.frame(chrom=rep("chr5", 3),
                  start=c(base.loc, base.loc+100, base.loc + 250),
                  end=c(base.loc + 50, base.loc+120, base.loc+290),
                  name=c("a", "b", "c"),
                  score=runif(3),
                  strand=rep("*", 3),
                  stringsAsFactors=FALSE)
```

```
gr <- GRanges(tbl)
track <- GRangesQuantitativeTrack("GRangesQTest", gr)
```

GWASTrack-class *Constructor for GWASTrack*

Description

GWASTrack creates an IGV manhattan track GWAS data

Usage

```
GWASTrack(
  trackName,
  table,
  chrom.col,
  pos.col,
  pval.col,
  colorTable = list(),
  autoscale = TRUE,
  min = 0,
  max = 10,
  trackHeight = 50,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
table	data.frame of 6 or more columns
chrom.col	numeric, the column number of the chromosome column
pos.col	numeric, the column number of the position column
pval.col	numeric, the column number of the GWAS pvalue colum
colorTable	a named list of CSS colors, by chromosome name - exact matches to the names in the GWAS table.
autoscale	logical, controls how min and max of the y-axis are determined
min	numeric when autoscale is FALSE, use this minimum y
max	numeric when autoscale is FALSE, use this maximum y
trackHeight	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
visibilityWindow	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Value

A GWASTrack object

Examples

```
file <- system.file(package="igvR", "extdata", "gwas-5k.tsv")
tbl.gwas <- read.table(file, sep="\t", header=TRUE, quote="")
dim(tbl.gwas)
track <- GWASTrack("gwas 5k", tbl.gwas, chrom.col=12, pos.col=13, pval.col=28)

if(interactive()){
  igv <- igvR()
  setGenome(igv, "hg38")
  setBrowserWindowTitle(igv, "GWAS demo")
  displayTrack(igv, track)
  Sys.sleep(1)  # pause before zooming in
  showGenomicRegion(igv, "chr6:32,240,829-32,929,353")
  }
```

GWASUrlTrack *Constructor for GWASUrlTrack*

Description

GWASUrlTrack creates an IGV manhattan track GWAS data

Usage

```
GWASUrlTrack(
  trackName,
  url,
  chrom.col,
  pos.col,
  pval.col,
  colorTable = list(),
  autoscale = TRUE,
  min = 0,
  max = 10,
  trackHeight = 50,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
url	character

chrom.col	numeric, the column number of the chromosome column
pos.col	numeric, the column number of the position column
pval.col	numeric, the column number of the GWAS pvalue colum
colorTable	a named list of CSS colors, by chromosome name - exact matches to the names in the GWAS table.
autoscale	logical, controls how min and max of the y-axis are determined
min	numeric when autoscale is FALSE, use this minimum y
max	numeric when autoscale is FALSE, use this maximum y
trackHeight	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
visibilityWindow	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Value

A GWASUrlTrack object

Examples

```
  track <- GWASUrlTrack("GWAS from url",
                        "https://s3.amazonaws.com/igv.org.demo/gwas_sample.tsv.gz",
                         chrom.col=12, pos.col=13, pval.col=28)

    # note: this track is autoscaled.  apparently some infinite values in the file,
    # leading to a flat, low track.  reproduce this in static html, report issue to igv.js
    # temporary workaround: use the interactive track gear to set display range.

if(interactive()){
    igv <- igvR()
    setGenome(igv, "hg38")
    setBrowserWindowTitle(igv, "GWAS URL demo")
    displayTrack(igv, track)
    }
```

---

igvAnnotationTrack-class

Constructor for igvAnnotationTrack

Description

Constructor for igvAnnotationTrack

Usage

```
igvAnnotationTrack(
  trackName,
  annotation,
  fileFormat = c("bed"),
  color = "gray",
  displayMode = c("SQUISHED", "COLLAPSED", "EXPANDED"),
  sourceType = "file",
  trackHeight = 30,
  expandedRowHeight = 30,
  squishedRowHeight = 15,
  maxRows = 500,
  searchable = FALSE,
  visibilityWindow = 1e+05
)
```

Arguments

`trackName` A character string, used as track label by igv, we recommend unique names per track.

`annotation` An opague type, currently either a data.frame, GRanges, or UCSCBed object from rtracklayer.

`fileFormat` Only "bed" is currently supported.

`color` A CSS color name (e.g., "red" or "#FF0000")

`displayMode` "COLLAPSED", "EXPANDED", or "SQUISHED"

`sourceType` Only "file" sources are currently supported.

`trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)

`expandedRowHeight` Height of each row of features in "EXPANDED" mode.

`squishedRowHeight` Height of each row of features in "SQUISHED" mode, for compact viewing.

`maxRows` of features to display

`searchable` If TRUE, labels on annotation elements may be used in search

`visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Value

An igvAnnotationTrack object

igvR-class *Create an igvR object*

Description

The igvR class provides an R interface to igv.js, a rich, interactive, full-featured, javascript browser-based genome browser. One constructs an igvR instance on a specified port (default 9000), the browser code is loaded, and a websocket connection openend. After specifying the reference genome, any number of genome tracks may be created, displayed, and navigated.

Usage

```
igvR(
  portRange = 15000:15100,
  host = "localhost",
  title = "igvR",
  browserFile = igvBrowserFile,
  quiet = TRUE
)
```

Arguments

portRange	The constructor looks for a free websocket port in this range. 15000:15100 by default
host	character, often "localhost" but (as with RStudio Server deployment) can be a remote host
title	Used for the web browser window, "igvR" by default
browserFile	The full path to the bundled html, js and libraries, and css which constitute the browser app
quiet	A logical variable controlling verbosity during execution

Value

An object of the igvR class

Examples

```
if(interactive()){
   igv <- igvR(title="igv demo")
   setGenome(igv, "hg38")
   showGenomicRegion(igv, "MEF2C")
     #-----------------------------------------------------------------
     # an easy transparent way to create a bed track
     #-----------------------------------------------------------------
   base.loc <- 88883100
   tbl <- data.frame(chrom=rep("chr5", 3),
                     start=c(base.loc, base.loc+100, base.loc + 250),
```

```
                    end=c(base.loc + 50, base.loc+120, base.loc+290),
                    name=c("a", "b", "c"),
                    score=runif(3),
                    strand=rep("*", 3),
                    stringsAsFactors=FALSE)

  track <- DataFrameAnnotationTrack("dataframeTest", tbl, color="red", displayMode="EXPANDED")
   displayTrack(igv, track)
   showGenomicRegion(igv, sprintf("chr5:%d-%d", base.loc-100, base.loc+350))
   } # if interactive
```

---

parseAndValidateGenomeSpec

parseAndValidateGenomeSpec

Description

a helper function for internal use by the igvShiny constructor, but possible also of use to those building an igvShiny app, to test their genome specification for validity

Usage

```
parseAndValidateGenomeSpec(
  genomeName,
  initialLocus = "all",
  stockGenome = TRUE,
  dataMode = NA,
  fasta = NA,
  fastaIndex = NA,
  genomeAnnotation = NA
)
```

Arguments

genomeName	character usually one short code of a supported ("stock") genome (e.g., "hg38") or for a user-supplied custom genome, the name you wish to use
initialLocus	character default "all", otherwise "chrN:start-end" or a recognized gene symbol
stockGenome	logical default TRUE
dataMode	character either "stock", "localFile" or "http"
fasta	character when supplying a custom (non-stock) genome, either a file path or a URL
fastaIndex	character when supplying a custom (non-stock) genome, either a file path or a URL, essential for all but the very small custom genomes.
genomeAnnotation	character when supplying a custom (non-stock) genome, a file path or URL pointing to a genome annotation file in a gff3 format

Value

an options list directly usable by igvApp.js, and thus igv.js

See Also

[currently.supported.stock.genomes()] for stock genomes we support.

Examples

```
genomeSpec <- parseAndValidateGenomeSpec("hg38", "APOE")  # the simplest case
base.url <- "https://gladki.pl/igvr/testFiles/sarsGenome"
fasta.file <- sprintf("%s/%s", base.url,"Sars_cov_2.ASM985889v3.dna.toplevel.fa")
fastaIndex.file <-  sprintf("%s/%s", base.url, "Sars_cov_2.ASM985889v3.dna.toplevel.fa.fai")
annotation.file <-  sprintf("%s/%s", base.url, "Sars_cov_2.ASM985889v3.101.gff3")
custom.genome.title <- "SARS-CoV-2"
genomeOptions <- parseAndValidateGenomeSpec(genomeName=custom.genome.title,
                                            initialLocus="all",
                                            stockGenome=FALSE,
                                            dataMode="http",
                                            fasta=fasta.file,
                                            fastaIndex=fastaIndex.file,
                                            genomeAnnotation=annotation.file)
```

---

ping,igvR-method *Test the connection between your R session and the webapp*

Description

Test the connection between your R session and the webapp

Usage

```
## S4 method for signature 'igvR'
ping(obj, msecDelay = 0)
```

Arguments

obj	An object of class igvR
msecDelay	don't return until these many milliseconds have passed, default 0

Value

"pong"

Examples

```
if(interactive()){
   igv <- igvR()
   ping(igv)
   }
```

QuantitativeTrack-class

Constructor for QuantitativeTrack

Description

QuantitativeTrack creates an IGV track for genomic tracks in which a numerical value is associated with each reported location.

Usage

```
QuantitativeTrack(
  trackName,
  quantitativeData,
  fileFormat = c("wig", "bigWig", "bedGraph", "gwas"),
  color = "gray",
  sourceType = c("file", "url"),
  trackHeight = 50,
  autoscale = TRUE,
  min = NA_real_,
  max = NA_real_,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
quantitativeData	A polyvalent object, either a data.frame, GRanges, or UCSCBedGraphQuantitative object
fileFormat	only "bedGraph" supported at present; wig and bigWig support soon.
color	A CSS color name (e.g., "red" or "#FF0000")
sourceType	only "file" supported at present ("gcs" for Google Cloud Storage, and "ga4gh" for the Global Alliance API may come)
trackHeight	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
autoscale	Autoscale track to maximum value in view
min	Sets the minimum value for the data (y-axis) scale. Usually zero.

`max` Sets the maximum value for the data (y-axis) scale. This value is ignored if autoscale is TRUE

`visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description will go here

Value

A QuantitativeTrack object

`RemoteAlignmentTrack-class`

Constructor for RemoteAlignmentTrack

Description

`RemoteAlignmentTrack` creates an `IGV` track for remote bam files

Usage

```
RemoteAlignmentTrack(
  trackName,
  bamUrl,
  bamIndex = NULL,
  trackHeight = 50,
  visibilityWindow = 30000,
  color = "gray"
)
```

Arguments

`trackName` A character string, used as track label by igv, we recommend unique names per track.

`bamUrl` The URL of a bam file

`bamIndex` The URL of a bam index file. Defaults to <bamUrl>.bai

`trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)

`visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

`color` A character string, either a reconized color ("red") or a hex string ("#FF8532")

Details

Detailed description goes here

Value

A RemoteAlignmentTrack object

---

removeTracksByName,igvR-method *Remove named tracks*

Description

Remove named tracks

Usage

```
## S4 method for signature 'igvR'
removeTracksByName(obj, trackNames)
```

Arguments

obj	An object of class igvR
trackNames	a character vector

Value

A character vector

See Also

getTrackNames

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg19")
   showGenomicRegion(igv, "MEF2C")
     # create three arbitrary tracks
   base.loc <- 88883100
   tbl <- data.frame(chrom=rep("chr5", 3),
                     start=c(base.loc, base.loc+100, base.loc + 250),
                     end=c(base.loc + 50, base.loc+120, base.loc+290),
                     name=c("a", "b", "c"),
                     score=runif(3),
                     strand=rep("*", 3),
                     stringsAsFactors=FALSE)
```

```
track.1 <- DataFrameAnnotationTrack("track.1", tbl, color="red", displayMode="SQUISHED")
track.2 <- DataFrameAnnotationTrack("track.2", tbl, color="blue", displayMode="SQUISHED")
track.3 <- DataFrameAnnotationTrack("track.3", tbl, color="green", displayMode="SQUISHED")
 displayTrack(igv, track.1)
 displayTrack(igv, track.2)
 displayTrack(igv, track.3)
 removeTracksByName(igv, "track.2")
   #----------------------------------------
   # bulk removal of the remaining tracks,
   # but leave the h19 reference track
   #----------------------------------------
 removeTracksByName(igv, getTrackNames(igv)[-1])
 }
```

---

saveToSVG,igvR-method *Get entire igv browser image in svg*

Description

Get entire igv browser image in svg

Usage

```
## S4 method for signature 'igvR'
saveToSVG(obj, filename)
```

Arguments

obj	An object of class igvR
filename	character string, the name of the file to which the svg text will be written

Value

A character vector

---

setCustomGenome,igvR-method *Specify the reference genome you wish to use, via full specification of all urls*

Description

Specify the reference genome you wish to use, via full specification of all urls

Usage

```
## S4 method for signature 'igvR'
setCustomGenome(
  obj,
  id,
  genomeName,
  fastaURL,
  fastaIndexURL,
  chromosomeAliasURL = NA,
  cytobandURL = NA,
  geneAnnotationName = NA,
  geneAnnotationURL = NA,
  geneAnnotationTrackHeight = 200,
  geneAnnotationTrackColor = "darkblue",
  initialLocus = "all",
  visibilityWindow = 1e+06
)
```

Arguments

`obj` An object of class igvR

`id` character string, a short name, displayed in the browser, e.g., "hg38", "tair10".

`genomeName` character string, possibly longer, more descirptive then the id, e.g., "Human (GRCh38/hg38)"

`fastaURL` character string, e.g."https://s3.amazonaws.com/igv.broadinstitute.org/genomes/seq/hg38/hg38.fa"

`fastaIndexURL` character string, e.g. "https://s3.amazonaws.com/igv.broadinstitute.org/genomes/seq/hg38/hg38.fa.fai"

`chromosomeAliasURL` character string, default NA, a tab-delimited file supporting multiple equivalent chromosome names. see details

`cytobandURL` character string, default NA, a cytoband ideogram file in UCSC format, e.g. "https://s3.amazonaws.com/igv.broadinstitute.org/annotations/hg38/cytoBandIdeo.txt"

`geneAnnotationName` character string, e.g. "Refseq Genes", default NA

`geneAnnotationURL` character string, e.g. "https://s3.amazonaws.com/igv.org.genomes/hg38/refGene.txt.gz", default NA

`geneAnnotationTrackHeight` numeric, pixels, e.g. 500. default 200

`geneAnnotationTrackColor` character string, any legal CSS color, default "darkblue"

`initialLocus` character string, e.g. "chr5:88,621,308-89,001,037" or "MEF2C"

`visibilityWindow` numeric, number of bases over which to display features, default 1000000

Value

An empty string, an error message if any of the urls could not be reached

Examples

```
if(interactive()){
   igv <- igvR()
   setCustomGenome(igv,
                   id="hg38",
                   genomeName="Human (GRCh38/hg38)",
             fastaURL="https://s3.amazonaws.com/igv.broadinstitute.org/genomes/seq/hg38/hg38.fa",
             fastaIndexURL="https://s3.amazonaws.com/igv.broadinstitute.org/genomes/seq/hg38/hg38.fa.fai",
                   chromosomeAliasURL=NA,
             cytobandURL="https://s3.amazonaws.com/igv.broadinstitute.org/annotations/hg38/cytoBandIdeo.txt",
                   geneAnnotationName="Refseq Genes",
             geneAnnotationURL="https://s3.amazonaws.com/igv.org.genomes/hg38/refGene.txt.gz",
                   geneAnnotationTrackHeight=300,
                   geneAnnotationTrackColor="darkgreen",
                   initialLocus="chr5:88,621,308-89,001,037",
                   visibilityWindow=5000000)
   }
```

`setGenome,igvR-method` *Specify the reference genome, currently limited to hg38, hg19, mm10, tair10.*

Description

Specify the reference genome, currently limited to hg38, hg19, mm10, tair10.

Usage

```
## S4 method for signature 'igvR'
setGenome(obj, genomeName)
```

Arguments

`obj`	An object of class igvR
`genomeName`	A character string, one of "hg38", "hg19", "mm10", "tair10"

Value

An empty string, an error message if the requested genome is not yet supported

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "mm10")
   }
```

setTrackClickFunction,igvR-method

Specify (supply) the javascript function run on track click event

Description

Specify (supply) the javascript function run on track click event

Usage

```
## S4 method for signature 'igvR'
setTrackClickFunction(obj, javascriptFunction)
```

Arguments

`obj`	An object of class igvR
`javascriptFunction`	expressed as a 2-element named list: body + args

Value

`""`

setTrackHeight,igvR-method

Remove named tracks

Description

Remove named tracks

Usage

```
## S4 method for signature 'igvR'
setTrackHeight(obj, trackName, newHeight)
```

Arguments

`obj`	An object of class igvR
`trackName`	a character string
`newHeight`	integer, in ixels

Value

nothing

See Also

getTrackNames

`showGenomicRegion,igvR-method`

Set the visible region, by explicit chromLoc string, or by named features in any curently loaded annotation tracks

---

Description

Set the visible region, by explicit chromLoc string, or by named features in any curently loaded annotation tracks

Usage

```
## S4 method for signature 'igvR'
showGenomicRegion(obj, region)
```

Arguments

`obj`	An object of class igvR
`region`	A genomic location (rendered "chr5:9,234,343-9,236,000" or as a list: list(chrom="chr9", start=9234343, end=9236000)) or a labeled annotation in a searchable track, often a gene symbol, eg "MEF2C"

Value

`""`

Examples

```
if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   showGenomicRegion(igv, "MEF2C")
   x <- getGenomicRegion(igv)
      #--------------------
      # zoom out 2kb
      #--------------------
   showGenomicRegion(igv, with(x, sprintf("%s:%d-%d", chrom, start-1000, end+1000)))
   }
```

showTrackLabels,igvR-method *Hide or show igv track labels*

Description

Hide or show igv track labels

Usage

```
## S4 method for signature 'igvR'
showTrackLabels(obj, newState)
```

Arguments

obj	An object of class igvR
newState	logigal, either TRUE or FALSE

Value

""

Track-class *Constructor for Track*

Description

Constructor for Track

Usage

```
Track(
  trackType = c("annotation", "quantitative", "alignment", "variant", "gwas"),
  sourceType = c("file", "gcs", "ga4gh"),
 fileFormat = c("bed", "gff", "gff3", "gtf", "wig", "bigWig", "bedGraph", "bam", "vcf",
    "seg"),
  trackName,
  onScreenOrder,
  color,
  height,
  autoTrackHeight,
  minTrackHeight,
  maxTrackHeight,
  visibilityWindow
)
```

Arguments

- `trackType` One of "annotation", "quantitative", "variant".
- `sourceType` Only "file" is currently supported.
- `fileFormat` One of "bed", "bedGraph", "vcf"
- `trackName` A character string, used as track label by igv, we recommend unique names per track.
- `onScreenOrder` Numeric, for explicit placement of track within the current set.
- `color` A CSS color name (e.g., "red" or "#FF0000")
- `height` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
- `autoTrackHeight` If true, then track height is adjusted dynamically, within the bounds set by minHeight and maxHeight, to accomdodate features in view
- `minTrackHeight` In pixels, minimum allowed
- `maxTrackHeight` In pixels, maximum allowed
- `visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Value

An object of class Track

References

https://github.com/igvteam/igv.js/wiki/Tracks

https://www.w3schools.com/cssref/css_colors.asp

`trackInfo,Track-method`

Get basic info about a track: its type, file format, source and S4 class name

---

Description

Get basic info about a track: its type, file format, source and S4 class name

Usage

```
## S4 method for signature 'Track'
trackInfo(obj)
```

Arguments

obj	An object of base class Track

Value

A list with four fields: trackType, fileFormat, source, class name

---

trackSize,BedpeInteractionsTrack-method

Retrieve the size of the BedpeInteractionsTrack

Description

Retrieve the size of the BedpeInteractionsTrack

Usage

```
## S4 method for signature 'BedpeInteractionsTrack'
trackSize(obj)
```

Arguments

obj	An object of class BedpeInteractionsTrack

Value

The number of elements

---

trackSize,DataFrameAnnotationTrack-method

Retrieve the size of the DataFrameAnnotationTrack

Description

Retrieve the size of the DataFrameAnnotationTrack

Usage

```
## S4 method for signature 'DataFrameAnnotationTrack'
trackSize(obj)
```

Arguments

obj	An object of class UCSCBedAnnotationTrack

Value

The number of elements

Examples

```
base.loc <- 88883100
tbl <- data.frame(chrom=rep("chr5", 3),
                  start=c(base.loc, base.loc+100, base.loc + 250),
                  end=c(base.loc + 50, base.loc+120, base.loc+290),
                  name=c("a", "b", "c"),
                  score=runif(3),
                  strand=rep("*", 3),
                  stringsAsFactors=FALSE)

track <- DataFrameAnnotationTrack("dataframeTest", tbl)
trackSize(track)
```

---

trackSize,DataFrameQuantitativeTrack-method

Retrieve the size of the DataFrameQuantitativeTrack

Description

Retrieve the size of the DataFrameQuantitativeTrack

Usage

```
## S4 method for signature 'DataFrameQuantitativeTrack'
trackSize(obj)
```

Arguments

obj	An object of class DataFrameQuantitativeTrack

Value

The number of elements

trackSize,GenomicAlignmentTrack-method

Retrieve the size of the GenomicAlignmentTrack

Description

Retrieve the size of the GenomicAlignmentTrack

Usage

```
## S4 method for signature 'GenomicAlignmentTrack'
trackSize(obj)
```

Arguments

obj	An object of class GenomicAlignmentTrack

Value

The number of elements

trackSize,GFF3Track-method

Retrieve the size of the GFF3Track

Description

Retrieve the size of the GFF3Track

Usage

```
## S4 method for signature 'GFF3Track'
trackSize(obj)
```

Arguments

obj	An object of class UCSCBedAnnotationTrack

Value

The number of elements

trackSize,GRangesAnnotationTrack-method

Retrieve the size of the GRangesAnnotationTrack

Description

Retrieve the size of the GRangesAnnotationTrack

Usage

```
## S4 method for signature 'GRangesAnnotationTrack'
trackSize(obj)
```

Arguments

obj	An object of class GRangesAnnotationTrack

Value

The number of elements

trackSize,GRangesQuantitativeTrack-method

Retrieve the size of the GRangesQuantitativeTrack

Description

Retrieve the size of the GRangesQuantitativeTrack

Usage

```
## S4 method for signature 'GRangesQuantitativeTrack'
trackSize(obj)
```

Arguments

obj	An object of class GRangesQuantitativeTrack

Value

The number of elements

trackSize,GWASTrack-method

Retrieve the size of the GWASTrack

Description

Retrieve the size of the GWASTrack

Usage

```
## S4 method for signature 'GWASTrack'
trackSize(obj)
```

Arguments

obj	An object of class GWASTrack

Value

The number of elements

trackSize,GWASUrlTrack-method

Retrieve the size of the GWASUrlTrack

Description

Retrieve the size of the GWASUrlTrack

Usage

```
## S4 method for signature 'GWASUrlTrack'
trackSize(obj)
```

Arguments

obj	An object of class GWASUrlTrack

Value

The number of elements

trackSize,QuantitativeTrack-method

Retrieve the size of the QuantitativeTrack

Description

Retrieve the size of the QuantitativeTrack

Usage

```
## S4 method for signature 'QuantitativeTrack'
trackSize(obj)
```

Arguments

obj	An object of class UCSCBedAnnotationTrack

Value

The number of elements

trackSize,UCSCBedAnnotationTrack-method

Retrieve the size of theUCSCBedAnnotationTrack

Description

Retrieve the size of theUCSCBedAnnotationTrack

Usage

```
## S4 method for signature 'UCSCBedAnnotationTrack'
trackSize(obj)
```

Arguments

obj	An object of class UCSCBedAnnotationTrack

Value

The number of elements

Examples

```
bed.filepath <- system.file(package = "rtracklayer", "tests", "test.bed")
gr.bed <- rtracklayer::import(bed.filepath)
track.1 <- UCSCBedAnnotationTrack("UCSC bed", gr.bed, color="blue", displayMode="SQUISHED")
trackSize(track.1)
```

---

trackSize,UCSCBedGraphQuantitativeTrack-method

Retrieve the size of the UCSCBedGraphQuantitativeTrack

Description

Retrieve the size of the UCSCBedGraphQuantitativeTrack

Usage

```
## S4 method for signature 'UCSCBedGraphQuantitativeTrack'
trackSize(obj)
```

Arguments

obj	An object of class UCSCBedGraphQuantitativeTrack

Value

The number of elements

---

trackSize,VariantTrack-method

Retrieve the size of the VariantTrack

Description

Retrieve the size of the VariantTrack

Usage

```
## S4 method for signature 'VariantTrack'
trackSize(obj)
```

Arguments

obj	An object of class VariantTrack

Value

The number of elements

UCSCBedAnnotationTrack-class

Constructor for UCSCBedAnnotationTrack

Description

UCSCBedAnnotationTrack creates and IGV track for bed objects imported using rtracklayer

Usage

```
UCSCBedAnnotationTrack(
  trackName,
  annotation,
  color = "darkGrey",
  displayMode = "SQUISHED",
  trackHeight = 50,
  expandedRowHeight = 30,
  squishedRowHeight = 15,
  maxRows = 500,
  searchable = FALSE,
  visibilityWindow = 1e+05
)
```

Arguments

trackName	A character string, used as track label by igv, we recommend unique names per track.
annotation	A UCSCData object imported by rtracklayer
color	A CSS color name (e.g., "red" or "#FF0000")
displayMode	"COLLAPSED", "SQUISHED" or "EXPANDED". Spelling and case must be precise.
trackHeight	track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
expandedRowHeight	Height of each row of features in "EXPANDED" mode.
squishedRowHeight	Height of each row of features in "SQUISHED" mode, for compact viewing.
maxRows	of features to display
searchable	If TRUE, labels on annotation elements may be used in search
visibilityWindow	Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A UCSCBedAnnotationTrack object

Examples

```
bed.filepath <- system.file(package = "rtracklayer", "tests", "test.bed")
gr.bed <- rtracklayer::import(bed.filepath)
track <- UCSCBedAnnotationTrack("UCSC bed", gr.bed, color="blue", displayMode="SQUISHED")

if(interactive()){
    igv <- igvR()
    setGenome(igv, "hg38")
    setBrowserWindowTitle(igv, "UCSC bed10 demo")
    showGenomicRegion(igv, "chr7:127,469,879-127,476,276")
    displayTrack(igv, track)
    }
```

UCSCBedGraphQuantitativeTrack-class

Constructor for UCSCBedGraphQuantitativeTrack

Description

`UCSCBedGraphQuantitativeTrack` creates an `IGV` track for bedGraph objects imported with `rtracklayer`

Usage

```
UCSCBedGraphQuantitativeTrack(
  trackName,
  quantitativeData,
  color = "blue",
  trackHeight = 50,
  autoscale = TRUE,
  min = NA_real_,
  max = NA_real_,
  visibilityWindow = 1e+05
)
```

Arguments

`trackName`	A character string, used as track label by igv, we recommend unique names per track.

`quantitativeData`
: A GRanges object with (at least) a "score" metadata column

`color`
: A CSS color name (e.g., "red" or "#FF0000")

`trackHeight`
: track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)

`autoscale`
: Autoscale track to maximum value in view

`min`
: Sets the minimum value for the data (y-axis) scale. Usually zero.

`max`
: Sets the maximum value for the data (y-axis) scale. This value is ignored if autoscale is TRUE

`visibilityWindow`
: Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A UCSCBedGraphQuantitativeTrack object

Examples

```
bedGraph.filepath <- system.file(package = "rtracklayer", "tests", "test.bedGraph")
gr.bedGraph <- rtracklayer::import(bedGraph.filepath)
track <- UCSCBedGraphQuantitativeTrack("UCSCBedGraphTest", gr.bedGraph)

if(interactive()){
   igv <- igvR()
   setGenome(igv, "hg38")
   setBrowserWindowTitle(igv, "UCSC BedGraph demo")
   displayTrack(igv, track)
   Sys.sleep(1)  # pause before zoomin
   showGenomicRegion(igv, "chr18:59,103,373-59,105,673")
   }
```

`url.exists` *url.exists*

Description

a helper function for mostly internal use, tests for availability of a url, modeled after file.exists

a helper function for mostly internal use, tests for availability of a url, modeled after file.exists

Usage

```
url.exists(url)

url.exists(url)
```

Arguments

`url`	character the http address to test

Value

logical TRUE or FALSE

logical TRUE or FALSE

Examples

```
if(interactive()){
   igv <- igvR()
   ping(igv)
   }
```

VariantTrack-class *Constructor for VariantTrack*

Description

`VariantTrack` creates an `IGV` track for VCF (variant call format) objects, either local or at a remote url

Usage

```
VariantTrack(
  trackName,
  vcf,
  trackHeight = 50,
  anchorColor = "pink",
  homvarColor = "rgb(17,248,254)",
  hetvarColor = "rgb(34,12,253)",
  homrefColor = "rgb(200,200,200)",
  displayMode = "EXPANDED",
  visibilityWindow = 1e+05
)
```

Arguments

- `trackName` A character string, used as track label by igv, we recommend unique names per track.
- `vcf` A VCF object from the VariantAnnotation package, or a list(url=x, index=y) pointing to a vcf file
- `trackHeight` track height, typically in range 20 (for annotations) and up to 1000 (for large sample vcf files)
- `anchorColor` CSS color name (e.g., "red" or "#FF0000") for the "anchoring" graphical segment in the track
- `homvarColor` CSS color name for homozygous variant samples, rgb(17,248,254) by default (~turquoise)
- `hetvarColor` CSS color name for heterzygous variant samples, rgb(34,12,253) by default (~royalBlue)
- `homrefColor` CSS color names for homozygous reference samples, rgb(200,200,200) by default (~lightGray)
- `displayMode` "COLLAPSED", "EXPANDED", or "SQUISHED"
- `visibilityWindow` Maximum window size in base pairs for which indexed annotations or variants are displayed. Defaults: 1 MB for variants, whole chromosome for other track types.

Details

Detailed description goes here

Value

A VariantTrack object

Examples

```
    #----------------------------
    #  first, from a local file
    #----------------------------

f <- system.file("extdata", "chr22.vcf.gz", package="VariantAnnotation")
roi <- GRanges(seqnames="22", ranges=IRanges(start=c(50301422, 50989541),
                                             end=c(50312106, 51001328),
                                             names=c("gene_79087", "gene_644186")))
vcf.sub <- VariantAnnotation::readVcf(f, "hg19", param=roi)
track.local <- VariantTrack("chr22-tiny", vcf.sub)

    #----------------------------
    # now try a url track
    #----------------------------

data.url <- sprintf("%s/%s", "https://s3.amazonaws.com/1000genomes/release/20130502",
                     "ALL.wgs.phase3_shapeit2_mvncall_integrated_v5b.20130502.sites.vcf.gz")
```

```
index.url <- sprintf("%s.tbi", data.url)
url <- list(data=data.url, index=index.url)

track.url <- VariantTrack("1kg", url)
```

---

zoomIn,igvR-method *zoom the genome view in by a factor of 2*

Description

zoom the genome view in by a factor of 2

Usage

```
## S4 method for signature 'igvR'
zoomIn(obj)
```

Arguments

obj	An object of class igvR

Value

""

---

zoomOut,igvR-method *zoom the genome view out by a factor of 2*

Description

zoom the genome view out by a factor of 2

Usage

```
## S4 method for signature 'igvR'
zoomOut(obj)
```

Arguments

obj	An object of class igvR

Value

""

Index