Package ‘tracktables’

April 16, 2024

Type Package

Title Build IGV tracks and HTML reports

Version 1.36.0

Date 2021-11-21

Author Tom Carroll, Sanjay Khadayate, Anne Pajon, Ziwei Liang

Maintainer Tom Carroll <`tc.infomatics@gmail.com`>

Description Methods to create complex IGV genome browser sessions and dynamic IGV reports in HTML pages.

biocViews Sequencing, ReportWriting

VignetteBuilder knitr

License GPL (>= 3)

LazyLoad yes

Depends R (>= 3.0.0)

Imports IRanges, GenomicRanges, XVector, Rsamtools, XML, tractor.base, stringr, RColorBrewer, methods

Suggests knitr, BiocStyle

Collate 'tracktablesFunctions.R'

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

git_branch RELEASE_3_18

git_last_commit 8291a0a

git_last_commit_date 2023-10-24

Repository Bioconductor 3.18

Date/Publication 2024-04-15

R topics documented:

`igvParam` *Parameters for displaying bigwigs, bams and intervals in IGV*

Description

Use igvParam(object)) to create a parameter object to control IGV display invoked from maketracktable() report or from makeIGVSession() XML files. Parameters for bigwig, bam and intervals files may be provided. See IGV manual for a full description of options.

Usage

```
igvParam(bigwig.altColor = "darkgrey", bigwig.color = "darkgrey",
  bigwig.autoScale = "true", bigwig.displayMode = "COLLAPSED",
  bigwig.featureVisibilityWindow = -1, bigwig.fontSize = 10,
  bigwig.normalize = "false", bigwig.renderer = "BAR_CHART",
  bigwig.sortable = "true", bigwig.visible = "true",
  bigwig.windowFunction = "mean", bigwig.baseline = 0,
  bigwig.drawBaseline = "true", bigwig.flipAxis = "false",
  bigwig.maximum = 50, bigwig.minimum = 0, bigwig.type = "LINEAR",
  interval.altColor = "darkgrey", interval.color = "darkgrey",
  interval.autoScale = "true", interval.displayMode = "character",
  interval.featureVisibilityWindow = -1, interval.fontSize = 10,
  interval.height = 40, interval.normalize = "false",
  interval.renderer = "BASIC_FEATURE", interval.sortable = "true",
  interval.visible = "true", interval.windowFunction = "count",
  bam.altColor = "darkgrey", bam.color = "darkgrey",
  bam.autoScale = "true", bam.displayMode = "EXPANDED",
  bam.featureVisibilityWindow = -1, bam.fontSize = 10,
  bam.showSpliceJunctions = "false", bam.colorByTag = "",
  bam.colorOption = "UNEXPECTED_PAIR", bam.flagUnmappedPairs = "false",
  bam.groupByTag = "", bam.maxInsertSize = 1000, bam.minInsertSize = 50,
  bam.shadeBasesOption = "QUALITY", bam.shadeCenters = "true",
  bam.showAllBases = "false", bam.sortByTag = "")

igvParam(bigwig.altColor = "darkgrey", bigwig.color = "darkgrey",
  bigwig.autoScale = "true", bigwig.displayMode = "COLLAPSED",
  bigwig.featureVisibilityWindow = -1, bigwig.fontSize = 10,
  bigwig.normalize = "false", bigwig.renderer = "BAR_CHART",
  bigwig.sortable = "true", bigwig.visible = "true",
  bigwig.windowFunction = "mean", bigwig.baseline = 0,
  bigwig.drawBaseline = "true", bigwig.flipAxis = "false",
```

```
  bigwig.maximum = 50, bigwig.minimum = 0, bigwig.type = "LINEAR",
  interval.altColor = "darkgrey", interval.color = "darkgrey",
  interval.autoScale = "true", interval.displayMode = "character",
  interval.featureVisibilityWindow = -1, interval.fontSize = 10,
  interval.height = 40, interval.normalize = "false",
  interval.renderer = "BASIC_FEATURE", interval.sortable = "true",
  interval.visible = "true", interval.windowFunction = "count",
  bam.altColor = "darkgrey", bam.color = "darkgrey",
  bam.autoScale = "true", bam.displayMode = "EXPANDED",
  bam.featureVisibilityWindow = -1, bam.fontSize = 10,
  bam.showSpliceJunctions = "false", bam.colorByTag = "",
  bam.colorOption = "UNEXPECTED_PAIR", bam.flagUnmappedPairs = "false",
  bam.groupByTag = "", bam.maxInsertSize = 1000, bam.minInsertSize = 50,
  bam.shadeBasesOption = "QUALITY", bam.shadeCenters = "true",
  bam.showAllBases = "false", bam.sortByTag = "")
```

Arguments

`bigwig.altColor`	A character vector of alternate colour for bigwigs displayed in IGV.
`bigwig.color`	A character vector of main colour for bigwigs displayed in IGV.
`bigwig.displayMode`	A character vector of display mode for bigwigs displayed in IGV.
`bigwig.featureVisibilityWindow`	A numeric vector of feature visibility window for bigwigs displayed in IGV (Defaut -1).
`bigwig.fontSize`	A numeric vector of font size for bigwigs displayed in IGV.
`bigwig.autoScale`	A character vector ("true"/"false") to indicate whether data is autoscaled for bigwigs displayed in IGV (Default "true").
`bigwig.normalize`	A character vector ("true"/"false") to indicate whether data is normalised for bigwigs displayed in IGV (Default "false").
`bigwig.renderer`	A character vector of renderer for bigwigs displayed in IGV (Default "BAR_CHART").
`bigwig.sortable`	A character vector ("true"/"false") to indicate whether data is sortable for bigwigs displayed in IGV (Default "true").
`bigwig.visible`	A character vector ("true"/"false") to indicate whether data is visible for bigwigs displayed in IGV (Default "true").
`bigwig.windowFunction`	A character vector ("true"/"false") of window function for bigwigs displayed in IGV (Default "mean").
`bigwig.baseline`	A numeric vector of baseline bigwigs displayed in IGV.

`bigwig.drawBaseline`	A character vector ("true"/"false") of whether to draw baseline for bigwigs displayed in IGV (Default "true").
`bigwig.flipAxis`	A character vector ("true"/"false") to indicate whether to flip axis for bigwigs displayed in IGV (Default "false").
`bigwig.maximum`	A numeric vector of maximum value to display for bigwigs displayed in IGV (Default 50).
`bigwig.minimum`	A numeric vector of minimum value to display for bigwigs displayed in IGV (Default 0).
`bigwig.type`	A character vector of display type for bigwigs displayed in IGV (Default "LINEAR").
`interval.altColor`	A character vector of alternate colour for intervals displayed in IGV.
`interval.color`	A character vector of main colour for intervals displayed in IGV.
`interval.autoScale`	A character vector ("true"/"false") to indicate whether data is autoscaled for intervals displayed in IGV (Default "false").
`interval.displayMode`	A character vector of display mode for intervals displayed in IGV (Default "COLLAPSED").
`interval.featureVisibilityWindow`	A numeric vector of feature visibility window for intervals displayed in IGV (Defaut -1).
`interval.fontSize`	A numeric vector of font size for intervals displayed in IGV.
`interval.height`	A numeric vector of height for intervals displayed in IGV.
`interval.normalize`	A character vector ("true"/"false") to indicate whether data is normalised for intervals displayed in IGV (Default "false").
`interval.renderer`	A character vector of renderer for intervals displayed in IGV (Default "BASIC_FEATURE").
`interval.sortable`	A character vector ("true"/"false") to indicate whether data is sortable for intervals displayed in IGV (Default "true").
`interval.visible`	A character vector ("true"/"false") to indicate whether data is visible for intervals displayed in IGV (Default "true").
`interval.windowFunction`	A character vector ("true"/"false") of window function for intervals displayed in IGV (Default "count").
`bam.altColor`	A character vector of alternate colour for bam files displayed in IGV.
`bam.color`	A character vector of main colour for bam files displayed in IGV.

`bam.autoScale`	A character vector ("true"/"false") to indicate whether data is autoscaled for bam files displayed in IGV (Default "false").
`bam.displayMode`	A character vector of display mode for bam files displayed in IGV (Default "EXPANDED").
`bam.featureVisibilityWindow`	A numeric vector of feature visibility window for bam files displayed in IGV (Defaut -1).
`bam.fontSize`	A numeric vector of font size for intervals displayed in IGV.
`bam.showSpliceJunctions`	A character vector ("true"/"false") to indicate whether to show splice juntions for bam files displayed in IGV (Default "false").
`bam.colorByTag`	A character vector to indicate whether to colour reads by Tags for Bam files (Defualt = "").
`bam.colorOption`	A character vector of option to highlight Tags for Bam files (Defualt = "UNEXPECTED_PAIR").
`bam.flagUnmappedPairs`	A character vector ("true"/"false") to indicate whether to flaf unmapped pairs for bam files displayed in IGV (Default "false").
`bam.groupByTag`	A character vector ("true"/"false") to indicate how to groups reads by Tag for bam files displayed in IGV (Default "").
`bam.maxInsertSize`	A numeric vector of maximum insert size to display for Bam files.
`bam.minInsertSize`	A numeric vector of minimum insert size to display for Bam files.
`bam.shadeBasesOption`	A character vector of option to shade bases for Bam files (Default "QUALITY").
`bam.shadeCenters`	A character vector ("true"/"false") to indicate whether to shade centres for bam files displayed in IGV (Default "false").
`bam.showAllBases`	A character vector ("true"/"false") to indicate to show all bases for bam files displayed in IGV (Default "false").
`bam.sortByTag`	A character vector ("true"/"false") to indicate how to sort reads by Tag for bam files displayed in IGV (Default "").

Value

An igvParam class object to use with maketracktable,MakeIGVSession and makeIGVSessionMXL

Author(s)

Thomas Carroll

Examples

```
## Simple initialisation of an IGVParam object
igvDisplayParams <- igvParam()


## More custom initialisation of an IGVParam object .
igvDisplayParams <- igvParam(bigwig.color="red",bigwig.autoScale = "false",
bigwig.minimum = 10,bigwig.maximum = 100)


# See full parameters and IGV online manual for more details on customistions


## Use igvParams with maketracktables function to customise bigwig display colour and data range.
fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)

intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)

bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)

intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)

fileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)

fileSheet <- as.matrix(cbind(fileSheet,NA))

colnames(fileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(fileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
                     c("ProB","ProB","ProB","ProB"))

colnames(SampleSheet) <- c("SampleName","Antibody","Species")
MakeIGVSession(SampleSheet,fileSheet,
igvdirectory=getwd(),"Example","mm9",
igvParams=igvDisplayParams)
```

`Intervals` *Example genomic intervals*

Description

This dataset contains peaks from an in-house EBF1 ChIP-seq

Usage

```
data(Intervals)
```

Details

- Intervals GRanges object containing EBF1 peaks

Value

A GRanges object with two rows

`makebedtable` *Make HTML pages for interval files or GRanges.*

Description

Creates HTML pages for interval files or GRanges (Tracktables Interval Report).

Usage

```
makebedtable(grangesObject, name, basedirectory)
```

Arguments

`grangesObject`	A GRanges object.
`name`	Character of the name for Interval HTML report.
`basedirectory`	Character of the directory to which HTML report is writen.

Value

A character of file location for the Tracktables HTML Report

Author(s)

Thomas Carroll

Examples

```
data(Intervals)
htmlpage <- makebedtable(Intervals,"EBF_PeaksTable.html",getwd())
```

`MakeIGVSampleMetadata` *Make sample metadata file for use with IGV.*

Description

Creates sample metadata file for IGV from a Samplesheet of metadata and FileSheet of file locations.

Usage

```
MakeIGVSampleMetadata(SampleSheet, fileSheet, igvdirectory)
```

Arguments

`SampleSheet`	A data.frame object with metadata information for samples. First column must contain unique sample ids.
`fileSheet`	A data.frame of file locations. First column must contain the unique sample ids.
`igvdirectory`	A character of the directory to which sample metadata file is written.

Value

A character of file location for the IGV sample information file.

Author(s)

Thomas Carroll

Examples

```
fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)

intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)

bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)

intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)

fileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)

fileSheet <- as.matrix(cbind(fileSheet,NA))

colnames(fileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(fileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
```

```
                      c("ProB","ProB","ProB","ProB"))

colnames(SampleSheet) <- c("SampleName","Antibody","Species")
MakeIGVSampleMetadata(SampleSheet,fileSheet,igvdirectory=getwd())
```

`MakeIGVSession` *Make IGV session XML and sample information file*

Description

#' Creates IGV session XML and sample information file from a Samplesheet of metadata and FileSheet of file locations.

Usage

```
MakeIGVSession(SampleSheet, fileSheet, igvdirectory, XMLname, genomeName,
  locusName = "All", colourBy = NULL, igvParams = igvParam())
```

Arguments

`SampleSheet`	A data.frame object with metadata information for samples. First column must contain unique sample ids.
`fileSheet`	A data.frame of file locations. First column must contain the unique sample ids.
`igvdirectory`	A character of the directory to which sample metadata file is written.
`XMLname`	A character of the name for IGV session xml
`genomeName`	A character of genome for IGV (See IGV user guide for details)
`locusName`	A character of locus to display in igv on loading (See IGV user guide for details)
`colourBy`	Character defining which sample metadata to be used for colouring bigwig files
`igvParams`	An object of class igvParam containing display parameters for IGV. When providing a list, this list must be same length as number of samples and each element have two numeric values corresponding to minimum and maximum value to be used in setting data range. Currently only "autoscale" or a list of minimum and maximum values are accepted.

Value

A character of file location for the IGV session XML

Author(s)

Thomas Carroll

Examples

```
fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)

intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)

bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)

intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)

fileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)

fileSheet <- as.matrix(cbind(fileSheet,NA))

colnames(fileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(fileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
                     c("ProB","ProB","ProB","ProB"))

colnames(SampleSheet) <- c("SampleName","Antibody","Species")
MakeIGVSession(SampleSheet,fileSheet,igvdirectory=getwd(),"Example","mm9")
```

`MakeIGVSessionXML` *Make IGV session XML*

Description

Creates session XML for IGV from a FileSheet of file locations.

Usage

```
MakeIGVSessionXML(fileSheet, igvdirectory, XMLname, genomeName,
  locusName = "All", colourBy = NULL, igvParams = igvParam())
```

Arguments

`fileSheet`	A data.frame of file locations. First column must contain the unique sample ids.
`igvdirectory`	A character of the directory to which IGV XML session is written.
`XMLname`	A character of the name for IGV session xml
`genomeName`	A character of genome for IGV (See IGV user guide for details)
`locusName`	A character of locus to display in igv on loading (See IGV user guide for details)
`colourBy`	Character vector of RGB colours to use for colouring displayed BigWigs

`igvParams`	An object of class igvParam containing display parameters for IGV. When providing a list, this list must be same length as number of samples and each element have two numeric values corresponding to minimum and maximum value to be used in setting data range. Currently only "autoscale" or a list of minimum and maximum values are accepted.

Value

A character of file location for the IGV session XML

Author(s)

Thomas Carroll

Examples

```
fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)

intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)

bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)

intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)

fileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)

fileSheet <- as.matrix(cbind(fileSheet,NA))

colnames(fileSheet) <- c("SampleName","bigwig","interval","bam")

MakeIGVSessionXML(fileSheet,igvdirectory=getwd(),"Example","mm9")
```

`maketracktable` *Make HTML pages for IGV sessions (Tracktables Experiment Report)*

Description

Creates HTML table of sample metadata and all required files for interacting with IGV.

Usage

```
maketracktable(fileSheet, SampleSheet, filename, basedirectory, genome,
  colourBy = NULL, igvParams = igvParam())
```

Arguments

`fileSheet`	A data frame containing sample file locations (e.g. BigWig locations).
`SampleSheet`	A data frame containing sample metadata
`filename`	Character of name for tracktables HTML report. (.html prefix is added automatically)
`basedirectory`	Character of directory for tracktables HTML report, IGV sessions and any interval files
`genome`	Character of genome for IGV (See IGV user guide for details)
`colourBy`	Character defining which sample metadata to be used for colouring bigwig files
`igvParams`	An object of class igvParam containing display parameters for IGV. When providing a list, this list must be same length as number of samples and each element have two numeric values corresponding to minimum and maximum value to be used in setting data range. Currently only "autoscale" or a list of minimum and maximum values are accepted.

Value

An object containing XML document (HTMLInternalDocument,XMLInternalDocument,XMLAbstractDocument)

Author(s)

Thomas Carroll

Examples

```
fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)

intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)

bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)

intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)

fileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)

fileSheet <- as.matrix(cbind(fileSheet,NA))

colnames(fileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(fileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
                     c("ProB","ProB","ProB","ProB"))

colnames(SampleSheet) <- c("SampleName","Antibody","Species")
  HTMLreport <- maketracktable(fileSheet,SampleSheet,
```

```
"IGV_Example.html",
basedirectory=getwd(),
"mm9")
```

Index