Package 'epivizrStandalone'

November 13, 2026

Title Run Epiviz Interactive Genomic Data Visualization App within R

Version 1.41.0

Maintainer Hector Corrada Bravo <hcorrada@gmail.com>

Author Hector Corrada Bravo, Jayaram Kancherla

Description This package imports the epiviz visualization JavaScript app for genomic data interactive visualization. The 'epivizrServer' package is used to provide a web server running completely within R. This standalone version allows to browse arbitrary genomes through genome annotations provided by Bioconductor packages.

Depends R (>= 3.2.3), epivizr (>= 2.3.6), methods

License MIT + file LICENSE

LazyData true

Imports git2r, epivizrServer, Seqinfo, BiocGenerics, GenomicFeatures, S4Vectors

Suggests testthat, knitr, rmarkdown, OrganismDbi (>= 1.13.9), Mus.musculus, Biobase, BiocStyle

RoxygenNote 7.1.1

VignetteBuilder knitr

biocViews Visualization, Infrastructure, GUI

NeedsCompilation no

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

git_branch devel

git_last_commit f5fb41f

git_last_commit_date 2026-04-28

Repository Bioconductor 3.24

Date/Publication 2026-11-12

Contents

setStandalone 2
startStandalone 2
startStandaloneApp 4

Index **6**

setStandalone *Set settings for epiviz standalone repository.*

Description

The epiviz app run by function `startStandalone` in this package is cloned as a git repository. This function intializes the settings specifying which git repository is used. It can be either a github repository (the usual case), or local repository containing the epiviz JS app (used for testing and development).

Usage

```
setStandalone(
  url = "https://github.com/epiviz/epiviz.git",
  branch = "min",
  local_path = NULL,
  non_interactive = FALSE
)
```

Arguments

`url`	(character) github url to use. defaults to (`"https://github.com/epiviz/epiviz.git"`).
`branch`	(character) branch on the github repository. defaults to (master).
`local_path`	(character) if you already have a local instance of epiviz and would like to run standalone use this.
`non_interactive`	(logical) don't download repo, used for testing purposes.

Value

path to the epiviz app git repository

Examples

```
# argument non_interactive here to avoid downloading when testing
# package. Remove non_interactive argument when you try it out.
setStandalone(url="https://github.com/epiviz/epiviz.git", branch="master", non_interactive=TRUE)
```

startStandalone *Start a standalone* `epivizr` *session.*

Description

Uses the local repository of epiviz JS app to start a standalone epivizr session through the `startEpiviz` function. The epiviz app requires a list of sequence names and lengths (e.g., chromsome names and lengths) to setup genome browsing. These can be passed in the `seqinfo` argument or derived from the `gene_track` argument. The `gene_track` argument can be used to pass a genome annotation and add a gene track to the epiviz browser. See package vignette for further detail.

Usage

```
startStandalone(
  gene_track = NULL,
  seqinfo = NULL,
  keep_seqlevels = NULL,
  chr = NULL,
  start = NULL,
  end = NULL,
  non_interactive = FALSE,
  register_function = epivizr:::.register_all_the_epiviz_things,
  use_viewer_option = FALSE,
  host = "127.0.0.1",
  ...
)
```

Arguments

gene_track	(OrganismDb) an object of type OrganismDb or TxDb
seqinfo	(Seqinfo) an object of type Seqinfo from which sequence names and lengths are obtained
keep_seqlevels	(character) vector of sequence names to include in the standalone app
chr	(character) chromosome to browse to on app startup.
start	(integer) start location to browse to on app startup.
end	(integer) end location to browse to on app startup.
non_interactive	(logical) run server in non-interactive mode. Used for testing and development.
register_function	(function) function used to initialize actions in epiviz app. Used for testing and development.
use_viewer_option	(logical) run application in viewer defined by `getOption("viewer")`. This allows standalone app to run in Rstudio's viewer (FALSE by default)
host	(character) host address for application (127.0.0.1 by default)
...	additional arguments passed to startEpiviz.

Value

An object of class EpivizApp

Examples

```
# see package vignete for example usage
seqinfo <- Seqinfo::Seqinfo(c("chr1","chr2"), c(10,20))
app <- startStandalone(seqinfo=seqinfo, non_interactive=TRUE)
app$stop_app()
```

`startStandaloneApp` *Start a standalone* `epivizr` *session.*

Description

Uses the local installation of the epiviz desktop app to start a standalone epivizr session through the `startEpiviz` function. The epiviz app requires a list of sequence names and lengths (e.g., chromsome names and lengths) to setup genome browsing. These can be passed in the `seqinfo` argument or derived from the `gene_track` argument. The `gene_track` argument can be used to pass a genome annotation and add a gene track to the epiviz browser. See package vignette for further detail.

Usage

```
startStandaloneApp(
  gene_track = NULL,
  seqinfo = NULL,
  keep_seqlevels = NULL,
  chr = NULL,
  start = NULL,
  end = NULL,
  non_interactive = FALSE,
  register_function = epivizr:::.register_all_the_epiviz_things,
  host = "127.0.0.1",
  ...
)
```

Arguments

`gene_track`	(OrganismDb) an object of type `OrganismDb` or `TxDb`
`seqinfo`	(Seqinfo) an object of type `Seqinfo` from which sequence names and lengths are obtained
`keep_seqlevels`	(character) vector of sequence names to include in the standalone app
`chr`	(character) chromosome to browse to on app startup.
`start`	(integer) start location to browse to on app startup.
`end`	(integer) end location to browse to on app startup.
`non_interactive`	(logical) run server in non-interactive mode. Used for testing and development.
`register_function`	(function) function used to initialize actions in epiviz app. Used for testing and development.
`host`	(character) host address for application (127.0.0.1 by default)
`...`	additional arguments passed to `startEpiviz`.

Value

An object of class `EpivizApp`

Examples

```
# see package vignete for example usage
seqinfo <- Seqinfo::Seqinfo(c("chr1","chr2"), c(10,20))
app <- startStandalone(seqinfo=seqinfo, non_interactive=TRUE)
app$stop_app()
```

Index

EpivizApp, *3, 4*

OrganismDb, *3, 4*

Seqinfo, *3, 4*
setStandalone, 2
startEpiviz, *2–4*
startStandalone, *2*, 2
startStandaloneApp, 4

TxDb, *3, 4*