

# Package ‘epivizrChart’

December 30, 2024

**Title** R interface to epiviz web components

**Version** 1.28.0

**Description** This package provides an API for interactive visualization of genomic data using epiviz web components. Objects in R/BioConductor can be used to generate interactive R markdown/notebook documents or can be visualized in the R Studio's default viewer.

**Imports** epivizrData (>= 1.5.1), epivizrServer, htmltools, rjson, methods, BiocGenerics

**Suggests** testthat, roxygen2, knitr, Biobase, GenomicRanges, S4Vectors, IRanges, SummarizedExperiment, antiProfilesData, hgu133plus2.db, Mus.musculus, BiocStyle, Homo.sapiens, shiny, minfi, Rsamtools, rtracklayer, RColorBrewer, magrittr, AnnotationHub

**Collate** 'utils.R' 'EpivizChartDataMgr-class.R'  
'EpivizWebComponent-class.R' 'EpivizViewComponent-class.R'  
'EpivizDataSource-class.R' 'EpivizEnvironment-class.R'  
'EpivizChart-class.R' 'EpivizNavigation-class.R'  
'EpivizCharts-classes.R' 'vignette\_data.R' 'util-inits.R'  
'generics.R'

**VignetteBuilder** knitr

**Depends** R (>= 3.4.0)

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** true

**biocViews** Visualization, GUI

**RoxygenNote** 7.1.0

**git\_url** <https://git.bioconductor.org/packages/epivizrChart>

**git\_branch** RELEASE\_3\_20

**git\_last\_commit** 7dc2814

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.20

**Date/Publication** 2024-12-30

**Author** Brian Gottfried [aut],  
Jayaram Kancherla [aut],  
Hector Corrada Bravo [aut, cre]

**Maintainer** Hector Corrada Bravo <[hcorrada@gmail.com](mailto:hcorrada@gmail.com)>

## Contents

.constructURL . . . . .	3
.initialize_chart . . . . .	3
.register_all_the_epiviz_things . . . . .	4
.settings_as_df . . . . .	4
append_region . . . . .	5
bcode_eset . . . . .	5
BlocksTrack . . . . .	6
cgi_gr . . . . .	6
chart_default_settings_colors . . . . .	7
EpivizBlocksTrack-class . . . . .	7
epivizChart . . . . .	8
EpivizChart-class . . . . .	9
EpivizChartDataMgr-class . . . . .	10
EpivizDataSource-class . . . . .	10
epivizEnv . . . . .	11
EpivizEnvironment-class . . . . .	12
EpivizGenesTrack-class . . . . .	13
EpivizHeatmapPlot-class . . . . .	13
EpivizIGVTrack-class . . . . .	14
EpivizLinePlot-class . . . . .	14
EpivizLineTrack-class . . . . .	14
epivizNav . . . . .	15
EpivizNavigation-class . . . . .	15
EpivizScatterPlot-class . . . . .	16
EpivizStackedBlocksTrack-class . . . . .	17
EpivizStackedLinePlot-class . . . . .	17
EpivizStackedLineTrack-class . . . . .	17
EpivizViewComponent-class . . . . .	18
EpivizWebComponent-class . . . . .	18
get_available_chart_types . . . . .	19
get_registered_data_types . . . . .	19
HeatmapPlot . . . . .	20
json_parser . . . . .	20
json_writer . . . . .	21
LinePlot . . . . .	22
LineTrack . . . . .	22
plot,EpivizEnvironment,ANY-method . . . . .	23
rand_id . . . . .	23
ScatterPlot . . . . .	24
StackedBlocksTrack . . . . .	24
StackedLinePlot . . . . .	25
StackedLineTrack . . . . .	25
sumexp . . . . .	26
tcga_colon_blocks . . . . .	26
tcga_colon_curves . . . . .	27
tcga_colon_expression . . . . .	27

---

.constructURL      *Construct URL for Websocket connection between R and UI*

---

**Description**

Construct URL for Websocket connection between R and UI

**Usage**

```
.constructURL(host = "localhost", port = 7123L, path = "")
```

**Arguments**

host	host
port	port
path	path

**Value**

url

---

.initialize\_chart      *Initialize Epiviz Chart based on chart type*

---

**Description**

Initialize Epiviz Chart based on chart type

**Usage**

```
.initialize_chart(chart_type, ...)
```

**Arguments**

chart_type	Chart type.
...	Arguments for <a href="#">EpivizChart</a> objects.

`.register_all_the_epiviz_things`  
*(taken from epivizr) register epiviz actions*

---

**Description**

(taken from epivizr) register epiviz actions

**Usage**

```
.register_all_the_epiviz_things(srv, app)
```

**Arguments**

srv	epivizrServer object
app	EpivizApp object

---

`.settings_as_df`      *(taken from epivizr) print settings in a readable format*

---

**Description**

(taken from epivizr) print settings in a readable format

**Usage**

```
.settings_as_df(chart_settings)
```

**Arguments**

chart\_settings  chart settings

**Value**

chart settings as data frame

---

append_region	<i>Generic method to add navigation regions</i>
---------------	---

---

**Description**

Generic method to add navigation regions

**Usage**

```
append_region(x, ...)
```

```
## S4 method for signature 'EpivizEnvironment'
append_region(x, chr, start, end, return_parent = FALSE)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a>
...	other parameters
chr	chromosome id. ex. "chr11"
start	genomic region start
end	genomic region end
return_parent	To return the parent or the new navigation element. Defaults to FALSE

---

bcode_eset	<i>Example expression data from the Gene Expression barcode project</i>
------------	---

---

**Description**

See vignette data\_preprocessing for code to build this object.

**Usage**

```
data(bcode_eset)
```

**Format**

An `Biobase::ExpressionSet` object

---

BlocksTrack	<i>Method to add Blocks Track</i>
-------------	-----------------------------------

---

**Description**

Method to add Blocks Track

**Usage**

```
BlocksTrack(x, y, ...)
```

```
## S4 method for signature 'EpivizEnvironment'  
BlocksTrack(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

cgi_gr	<i>Locations of CpG Islands</i>
--------	---------------------------------

---

**Description**

Locations of CpG Islands

**Usage**

```
data(cgi_gr)
```

**Format**

A `GenomicRanges::GRanges` object with locations of CpG Islands in hg19

---

chart\_default\_settings\_colors  
*Get default chart settings and colors*

---

### **Description**

Get default chart settings and colors

### **Usage**

```
chart_default_settings_colors(chart_type)
```

### **Arguments**

chart\_type      chart type

### **Value**

list of settings and colors

---

EpivizBlocksTrack-class  
*Data container for an Epiviz Blocks Track.*

---

### **Description**

Data container for an Epiviz Blocks Track.

### **Methods**

get\_component\_type() Get component type for prefix of random id generator  
get\_default\_colors() Get default colors  
get\_default\_settings() Get default settings  
get\_name() Get name of Epiviz Web Component

---

epivizChart	<i>Initialize an <a href="#">EpivizChart</a> object to visualize in viewer or knit to HTML.</i>
-------------	---

---

### Description

Initialize an [EpivizChart](#) object to visualize in viewer or knit to HTML.

### Usage

```
epivizChart(
  data_obj = NULL,
  measurements = NULL,
  datasource_name = NULL,
  parent = NULL,
  chart = NULL,
  chr = NULL,
  start = NULL,
  end = NULL,
  settings = NULL,
  colors = NULL,
  ...
)
```

### Arguments

data_obj	A data object that will register to an <a href="#">EpivizData</a> object.
measurements	An <a href="#">EpivizMeasurement</a> object.
datasource_name	A name for the datasource. For example, "Mean by Sample Type".
parent	An object of class <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a> to append the chart within.
chart	The chart type to be visualized: "BlocksTrack", "HeatmapPlot", "LinePlot", "LineTrack", "ScatterPlot", "StackedLinePlot", "StackedLineTrack".
chr	The chromosome to filter on, e.g., chr="chr11".
start	The start location, e.g., start=110800000.
end	The end location, e.g., end=130383180.
settings	List of settings, e.g., list(title="Blocks Chart").
colors	List of colors. When chart is rendered to html this will be converted to a string encoded as JSON
...	Additional arguments passed to <a href="#">epivizrData::register</a> , e.g., type="bp", columns=c("normal, cancer").

### Value

An object of class [EpivizChart](#).



**Examples**

```

data(tcga_colon_blocks)
start <- 99800000
end <- 103383180
blocks_track <- epivizChart(tcga_colon_blocks, chr="chr11", start=start, end=end)
# See package vignette for more examples.

```

---

EpivizChart-class      *Data container for an Epiviz chart component.*

---

**Description**

Data container for an Epiviz chart component.

**Fields**

**data** (list) Values of an epiviz chart's data attribute.  
**colors** (character) Epiviz chart's colors attribute.  
**settings** (list) Epiviz chart's settings attribute.  
**parent** An object of class [EpivizEnvironment](#) where chart is appended.

**Methods**

**get\_attributes()** Get attributes for rendering chart  
**get\_available\_settings()** Get available settings  
**get\_colors()** Get chart colors  
**get\_data()** Get chart data  
**get\_parent()** Get parent  
**get\_settings()** Get chart settings  
**navigate(chr, start, end)** Navigate chart to a genomic location  
    **chr** Chromosome  
    **start** Start location  
    **end** End location  
**render\_component(shiny = FALSE)** Render to html  
**revisualize(chart\_type)** Revisualize chart as the given chart type  
    **chart\_type** The type of chart to be visualized (BlocksTrack, HeatmapPlot, LinePlot, LineTrack, ScatterPlot, StackedLinePlot, StackedLineTrack)  
**set\_colors(colors)** Set chart colors  
**set\_data(data)** Set chart data  
**set\_settings(settings)** Modify current settings  
    **settings** List of new settings. Call **get\_available\_settings** for settings available to modify.

---

 EpivizChartDataMgr-class

*Class providing data manager for epiviz charts.*


---

### Description

Class providing data manager for epiviz charts.

### Fields

.ms\_list (environment) List of measurment records

.ms\_idCounter (integer) Counter for ID generator

### Methods

add\_genome(genome) Add genome to data manager (for seqInfo)

**chr** Chromosome

**start** Start location

**end** End location

add\_measurements( obj, datasource\_name = NULL, datasource\_obj\_name = deparse(substitute(obj)), ... )

Register measurements in data manager

get\_data(measurements, chr = NULL, start = NULL, end = NULL) Get data from data mgr based on measurements, chr, start, and end

**measurements** List of EpivizMeasurements

**chr** Chromosome

**start** Start location

**end** End location

register\_shiny\_handler(session) Handlers to enable interactions with Shiny session.

**session** Shiny session object

rm\_all\_measurements() Remove all registered measurements

rm\_measurements(ms\_obj\_or\_id) Remove registered measurements from a given data object

---

 EpivizDataSource-class

*Data container for an Epiviz Data Source component.*


---

### Description

Data container for an Epiviz Data Source component.

### Fields

provider\_type (character)

provider\_id (character)

provider\_url (character)

**Methods**

`get_attributes()` Get attributes for rendering web component  
`get_component_type()` Get component type for prefix of random id generator  
`get_name()` Get name of Epiviz Web Component  
`get_provider_id()` Get provider id  
`get_provider_type()` Get provider type  
`get_provider_url()` Get provider url  
`render_component(shiny = FALSE)` Render to html  
`set_provider_url(url)` Set provider url  
`set_provider_id(id)` Set provider id  
`set_provider_type(type)` Set provider type

---

epivizEnv	<i>Initialize an <a href="#">EpivizEnvironment</a> object.</i>
-----------	--

---

**Description**

Initialize an [EpivizEnvironment](#) object.

**Usage**

```
epivizEnv(chr = NULL, start = NULL, end = NULL, interactive = FALSE, ...)
```

**Arguments**

chr	The chromosome to filter on, e.g., chr="chr11"
start	The start location, e.g., start=99800000.
end	The end location, e.g., end=130383180.
interactive	(logical) enable if running a websocket/shiny server
...	Additional params to pass to <a href="#">EpivizWebComponent</a>

**Value**

An object of class [EpivizEnvironment](#)

**Examples**

```
epiviz <- epivizEnv(chr="chr11", start=99800000, end=103383180)
```

---

EpivizEnvironment-class

*Data container for an Epiviz environment component.*

---

## Description

Data container for an Epiviz environment component.

## Fields

`charts` List of class [EpivizViewComponent](#) used to track nested elements.

`interactive` Logical value of whether component is interactive with data source component..

`epiviz_ds` [EpivizDataSource](#) object for interactive documents.

## Methods

`add_data(...)` Add data to environment's data manager

... Arguments for `add_measurements` and `register`, e.g., `data`, `datasource_name`, `datasource_obj_name`, `type`, etc

`add_genome(genome, type = "gene_info", datasource_name = NULL)` Add a genome to the view, and a genes-track.

**genome** annotation object. eg. `Homo.sapiens`

`append_chart(chart)` Append chart or navigation to environment

`append_region(chr = NULL, start = NULL, end = NULL, ...)` Add a child Navigation element to the given genomic region

`get_charts()` Get charts within environment

`get_component_type()` Get component type for prefix of random id generator

`get_measurements()` Get measurements

`get_name()` Get name of Epiviz Web Component

`get_rows(...)` Get row data from environment's data manager

`get_values(...)` Get value data from environment's data manager

`init_region(chr = NULL, start = NULL, end = NULL)` Initialize navigation based on a genomic region

**chr** Chromosome

**start** Start location

**end** End location

`init_regions(regions)` Initialize navigations based on genomic regions

**regions** List of named lists of genomic locations, e.g., `list(list(chr='chr11', start=99800000, end=103383180))`

`is_interactive()` Return whether the environment is interactive with a data source

`navigate(chr = NULL, start = NULL, end = NULL)` Navigate environment to genomic location

**chr** Chromosome

**start** Start location

**end** End location

`order_charts(ordered_charts)` Order the charts within an environment  
     **charts** An ordered list of EpivizViewComponent objects  
`plot(...)` Plot an EpivizChart within the environment  
     ... Arguments for epivizChart  
`register_shiny_handler(session)` Enable components to interact with Shiny session.  
     **session** Shiny session object  
`remove_all_charts()` Remove all charts from environment  
`remove_chart(chart)` Remove chart from environment  
`render_component(shiny = FALSE)` Render to html  
     **shiny** if rendering component in a shiny environment  
`set_charts(charts)` Set charts of environment

EpivizGenesTrack-class

*Data container for an Epiviz Genes Track.*

### Description

Data container for an Epiviz Genes Track.

### Methods

`get_component_type()` Get component type for prefix of random id generator  
`get_default_colors()` Get default colors  
`get_default_settings()` Get default settings  
`get_name()` Get name of Epiviz Web Component

EpivizHeatmapPlot-class

*Data container for an Epiviz Heatmap Plot.*

### Description

Data container for an Epiviz Heatmap Plot.

### Methods

`get_component_type()` Get component type for prefix of random id generator  
`get_default_colors()` Get default colors  
`get_default_settings()` Get default settings  
`get_name()` Get name of Epiviz Web Component

---

EpivizIGVTrack-class *Data container for an Epiviz Genes Track.*

---

**Description**

Data container for an Epiviz Genes Track.

**Methods**

get\_attributes() Get attributes for rendering component  
get\_component\_type() Get component type for prefix of random id generator  
get\_default\_colors() Get default colors  
get\_default\_settings() Get default settings  
get\_name() Get name of Epiviz Web Component

---

EpivizLinePlot-class *Data container for an Epiviz Line Plot.*

---

**Description**

Data container for an Epiviz Line Plot.

**Methods**

get\_component\_type() Get component type for prefix of random id generator  
get\_default\_colors() Get default colors  
get\_default\_settings() Get default settings  
get\_name() Get name of Epiviz Web Component

---

EpivizLineTrack-class *Data container for an Epiviz Line Track.*

---

**Description**

Data container for an Epiviz Line Track.

**Methods**

get\_component\_type() Get component type for prefix of random id generator  
get\_default\_colors() Get default colors  
get\_default\_settings() Get default settings  
get\_name() Get name of Epiviz Web Component

---

epivizNav	<i>Initialize an <a href="#">EpivizNavigation</a> object to visualize in viewer or knit to HTML.</i>
-----------	--

---

**Description**

Initialize an [EpivizNavigation](#) object to visualize in viewer or knit to HTML.

**Usage**

```
epivizNav(
  chr = NULL,
  start = NULL,
  end = NULL,
  parent = NULL,
  interactive = FALSE,
  ...
)
```

**Arguments**

chr	The chromosome to filter on, e.g., chr="chr11".
start	The start location, e.g., start=99800000.
end	The end location, e.g., end=130383180.
parent	An object of class [ <a href="#">EpivizEnvironment</a> ] or <a href="#">EpivizNavigation</a> to append the chart within.
interactive	(logical) enable if running a websocket/shiny server
...	Additional arguments for initializing navigation, e.g., gene and geneInRange.

**Value**

An object of class [EpivizNavigation](#).

**Examples**

```
epiviz <- epivizNav(chr="chr11", start=99800000, end=103383180)
```

---

`EpivizNavigation-class`

*Data container for an Epiviz navigation component.*

---

**Description**

Data container for an Epiviz navigation component.

**Fields**

gene (character) Gene  
 geneInRange (character) Nearest Gene in range.  
 parent An object of class [EpivizEnvironment](#) where navigation is appended.

**Methods**

add\_genome(genome, type = "gene\_info", datasource\_name = NULL) Add a genome to the view, and a genes-track.

**genome** annotation object. eg. Homo.sapiens

clone\_charts(charts) Clone EpivizCharts and append to navigation. Each chart must already exist in the navigation's data manager, otherwise an error will occur when attempting to initialize using their measurements

**charts** list of EpivizCharts whose data exists in the navigation's data manager

get\_attributes() Get attributes for rendering chart

get\_component\_type() Get component type for prefix of random id generator

get\_gene() Get gene

get\_geneInRange() Get gene in range

get\_name() Get name of Epiviz Web Component

render\_component(shiny = FALSE) Render to html

**shiny** if rendering component in a shiny environment

set\_gene(gene) Set gene

set\_geneInRange(gene) Set step ratio

---

EpivizScatterPlot-class

*Data container for an Epiviz Scatter Plot.*

---

**Description**

Data container for an Epiviz Scatter Plot.

**Methods**

get\_component\_type() Get component type for prefix of random id generator

get\_default\_colors() Get default colors

get\_default\_settings() Get default settings

get\_name() Get name of Epiviz Web Component



---

EpivizStackedBlocksTrack-class

*Data container for an Epiviz Blocks Track.*

---

**Description**

Data container for an Epiviz Blocks Track.

**Methods**

get\_component\_type() Get component type for prefix of random id generator

get\_default\_colors() Get default colors

get\_default\_settings() Get default settings

get\_name() Get name of Epiviz Web Component

---

EpivizStackedLinePlot-class

*Data container for an Epiviz Stacked Line Plot.*

---

**Description**

Data container for an Epiviz Stacked Line Plot.

**Methods**

get\_component\_type() Get component type for prefix of random id generator

get\_default\_colors() Get default colors

get\_default\_settings() Get default settings

get\_name() Get name of Epiviz Web Component

---

EpivizStackedLineTrack-class

*Data container for an Epiviz Stacked Line Track.*

---

**Description**

Data container for an Epiviz Stacked Line Track.

**Methods**

get\_component\_type() Get component type for prefix of random id generator

get\_default\_colors() Get default colors

get\_default\_settings() Get default settings

get\_name() Get name of Epiviz Web Component

EpivizViewComponent-class

*Data container for an Epiviz web component.*

---

### **Description**

Data container for an Epiviz web component.

### **Fields**

chr (CharacterOrNULL) Chromosome location.

start (NumericOrNULL) Start location.

end (NumericOrNULL) End location.

measurements (ListOrNULL) list of measurements of class [EpivizMeasurement](#).

### **Methods**

get\_attributes() Get attributes for rendering web component

get\_chr() Get chromosome

get\_end() Get end

get\_measurements() Get measurements

get\_start() Get start

set\_chr(chr) Set the chromosome

set\_end(end) Set end

set\_measurements(ms) Set measurements

set\_start(start) Set start

---

EpivizWebComponent-class

*Data container for an Epiviz Web component.*

---

### **Description**

Data container for an Epiviz Web component.

### **Fields**

data\_mgr [EpivizChartDataMgr](#)

name (character) Epiviz chart type (tag name).

class (CharacterOrNULL) Epiviz chart's class attribute.

id (character) Epiviz chart's id attribute.

**Methods**

- `get_attributes()` Get attributes for rendering web component
- `get_class()` Get class
- `get_data_mgr()` Get data manager
- `get_id()` Get id
- `get_name()` Get name of Epiviz Web Component
- `set_class(class)` Set chart class
- `set_id(id)` Set chart id
- `set_name(name)` Set name

---

`get_available_chart_types`  
*Construct URL for Websocket connection between R and UI*

---

**Description**

Construct URL for Websocket connection between R and UI

**Usage**

`get_available_chart_types()`

**Value**

url

---

`get_registered_data_types`  
*Construct URL for Websocket connection between R and UI*

---

**Description**

Construct URL for Websocket connection between R and UI

**Usage**

`get_registered_data_types()`

**Value**

url

---

HeatmapPlot	<i>Method to add Heatmap Plot</i>
-------------	-----------------------------------

---

**Description**

Method to add Heatmap Plot

**Usage**

```
HeatmapPlot(x, y, ...)

## S4 method for signature 'EpivizEnvironment'
HeatmapPlot(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

json_parser	<i>JSON parser used by this package</i>
-------------	---

---

**Description**

Currently this just renames [fromJSON](#) in the `rjson` package.

**Usage**

```
json_parser(
  json_str,
  file,
  method = "C",
  unexpected.escape = "error",
  simplify = TRUE
)
```

**Arguments**

json_str	json string to parse
file	file to read json_Str from
method	method used to parse json
unexpected.escape	handling escape characters, one of error, skip, keep
simplify	if TRUE, convert json-encoded lists to vectors

**Value**

a JSON object

**See Also**[fromJSON](#)**Examples**

```
json_parser('{ "a":true, "b":false, "c":null}')
```

---

`json_writer`*JSON writer used by this package*

---

**Description**

Currently this just renames [toJSON](#) in the rjson package.

**Usage**

```
json_writer(x, indent = 0, method = "C")
```

**Arguments**

<code>x</code>	object to write to json
<code>indent</code>	integer specifying how much indentation to use when formatting the JSON object; if 0, no pretty-formatting is used
<code>method</code>	method used to write json

**Value**

a string with JSON encoding of object

**See Also**[toJSON](#)**Examples**

```
json_writer(1:10)
```

LinePlot

*Method to add Line Plot*

---

**Description**

Method to add Line Plot

**Usage**

```
LinePlot(x, y, ...)
```

```
## S4 method for signature 'EpivizEnvironment'  
LinePlot(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

LineTrack*Method to add Line Track*

---

**Description**

Method to add Line Track

**Usage**

```
LineTrack(x, y, ...)
```

```
## S4 method for signature 'EpivizEnvironment'  
LineTrack(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

plot,EpivizEnvironment,ANY-method

*Generic methods to plot charts and add navigation regions*

---

### Description

Generic methods to plot charts and add navigation regions

### Usage

```
## S4 method for signature 'EpivizEnvironment,ANY'  
plot(x, y, ...)
```

### Arguments

x                    an object of type [EpivizEnvironment](#) or [EpivizNavigation](#)  
y                    a genomic data object  
...                  other parameters for the plot method

### Examples

```
## Not run:  
library(epivizrChart)  
library(Homo.sapiens)  
require(magrittr)  
  
# example data set  
data(sumexp)  
# create an environment element  
epivizEnv <- epivizEnv()  
  
# chain and add navigation regions and plots.  
epivizEnv %>%  
  plot(sumexp, datasource_name="sumExp", columns=c("cancer", "normal")) %>%  
  append_region(chr="chr11", start=118000000, end=121000000) %>%  
  plot(sumexp, datasource_name="sumExp", columns=c("normal", "cancer"))  
epivizEnv  
  
## End(Not run)
```

---

rand\_id

*Random ID generator for epiviz charts*

---

### Description

Random ID generator for epiviz charts

### Usage

```
rand_id(prefix = "")
```

**Arguments**

prefix            prefix for random ID

**Value**

random ID

---

ScatterPlot            *Method to add Scatter Plot*

---

**Description**

Method to add Scatter Plot

**Usage**

```
ScatterPlot(x, y, ...)
```

```
## S4 method for signature 'EpivizEnvironment'
ScatterPlot(x, y, ...)
```

**Arguments**

x                    an object of type [EpivizEnvironment](#) or [EpivizNavigation](#)  
y                    a genomic data object  
...                   other parameters for the plot method

---

StackedBlocksTrack    *Method to add Stacked Blocks Track*

---

**Description**

Method to add Stacked Blocks Track

**Usage**

```
StackedBlocksTrack(x, y, ...)
```

```
## S4 method for signature 'EpivizEnvironment'
StackedBlocksTrack(x, y, ...)
```

**Arguments**

x                    an object of type [EpivizEnvironment](#) or [EpivizNavigation](#)  
y                    a genomic data object  
...                   other parameters for the plot method



---

StackedLinePlot	<i>Method to add Stacked Line Plot</i>
-----------------	--

---

**Description**

Method to add Stacked Line Plot

**Usage**

```
StackedLinePlot(x, y, ...)  
  
## S4 method for signature 'EpivizEnvironment'  
StackedLinePlot(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

StackedLineTrack	<i>Method to add Stacked Line Track</i>
------------------	---

---

**Description**

Method to add Stacked Line Track

**Usage**

```
StackedLineTrack(x, y, ...)  
  
## S4 method for signature 'EpivizEnvironment'  
StackedLineTrack(x, y, ...)
```

**Arguments**

x	an object of type <a href="#">EpivizEnvironment</a> or <a href="#">EpivizNavigation</a>
y	a genomic data object
...	other parameters for the plot method

---

 sumexp

*Example SummarizedExperiment for epivizr vignette*


---

**Description**

Example SummarizedExperiment for epivizr vignette

**Usage**

```
data(sumexp)
```

**Format**

A SummarizedExperiment::RangedSummarizedExperiment object.

---

tcga\_colon\_blocks

*Example methylation data (blocks) for epivizr vignette.*


---

**Description**

Example results from methylation analysis of human chromosome 11 using the minfi package of TCGA 450k beadarray samples. This object contains large regions of methylation difference between tumor and normal samples obtained from `minfi::blockFinder()`.

**Usage**

```
data(tcga_colon_blocks)
```

**Format**

A GenomicRanges::GRanges object with 129 and mcols:

value average smooth methylation difference within block

area block area estimate ( $\text{abs}(\text{value}) * \text{length}$ )

cluster id of cluster blockgroup within which block occurs

indexStart index of first cluster in block

indexEnd index of last cluster in block

L number of clusters in block

clusterL number of probes in block

p.value permutation p.value based on difference conditioned on length

fwer family-wise error rate estimate based on difference conditioned on length

p.valueArea permutation p.value based on area

fwerArea family-wise error rate estimate based on area

**Source**

TCGA project: <https://tcga-data.nci.nih.gov/tcga/>

---

tcga_colon_curves	<i>Example methylation data (smoothed methylation levels) for epivizr vignette</i>
-------------------	--

---

### Description

Example results from methylation analysis of human chromosome 11 using the `minfi` package of TCGA 450k beadarray samples. This object contains probe cluster level methylation estimates from `minfi::blockFinder()`.

### Usage

```
data(tcga_colon_curves)
```

### Format

A `GenomicRanges::GRanges` object with 7135 ranges and `mcols`:

`id` probe cluster id

`type` probe cluster type

`blockgroup` probe cluster block group

`diff` raw methylation percentage difference between normal and tumor

`smooth` smooth methylation percentage difference between normal and tumor

`normalMean` mean methylation estimate for normal samples

`cancerMean` mean methylation estimate for cancer samples

### Source

TCGA project: <https://tcga-data.nci.nih.gov/tcga/>

---

tcga_colon_expression	<i>Example exon-level RNAseq data from TCGA project for epivizr vignette.</i>
-----------------------	---

---

### Description

A `SummarizedExperiment::RangedSummarizedExperiment` object containing exon-level counts from RNAseq data for colon tumor and normal tissue from the TCGA project. Only exons in human chromosome 11 are included.

### Usage

```
data(tcga_colon_expression)
```

**Format**

A `SummarizedExperiment::RangedSummarizedExperiment` object with 12,800 rows (exons) and 40 samples.

`assay(tcga_colon_expression)` exon-level count matrix

`colData(tcga_colon_expression)` a `DataFrame` containing sample information. Normal/Tumor status is given in column `sample_type`

**Source**

TCGA project: <https://tcga-data.nci.nih.gov/tcga/>

# Index

.constructURL, [3](#)  
.initialize\_chart, [3](#)  
.register\_all\_the\_epiviz\_things, [4](#)  
.settings\_as\_df, [4](#)

append\_region, [5](#)  
append\_region, EpivizEnvironment-method  
(append\_region), [5](#)

bcode\_eset, [5](#)  
BlocksTrack, [6](#)  
BlocksTrack, EpivizEnvironment-method  
(BlocksTrack), [6](#)

cgi\_gr, [6](#)  
chart\_default\_settings\_colors, [7](#)

EpivizBlocksTrack  
(EpivizBlocksTrack-class), [7](#)  
EpivizBlocksTrack-class, [7](#)  
EpivizChart, [3, 8](#)  
EpivizChart (EpivizChart-class), [9](#)  
epivizChart, [8](#)  
EpivizChart-class, [9](#)  
EpivizChartDataMgr, [18](#)  
EpivizChartDataMgr  
(EpivizChartDataMgr-class), [10](#)  
EpivizChartDataMgr-class, [10](#)  
EpivizData, [8](#)  
EpivizDataSource, [12](#)  
EpivizDataSource  
(EpivizDataSource-class), [10](#)  
EpivizDataSource-class, [10](#)  
epivizEnv, [11](#)  
EpivizEnvironment, [5, 6, 8, 9, 11, 16, 20, 22–25](#)  
EpivizEnvironment  
(EpivizEnvironment-class), [12](#)  
EpivizEnvironment-class, [12](#)  
EpivizGenesTrack  
(EpivizGenesTrack-class), [13](#)  
EpivizGenesTrack-class, [13](#)  
EpivizHeatmapPlot  
(EpivizHeatmapPlot-class), [13](#)  
EpivizHeatmapPlot-class, [13](#)  
EpivizIGVTrack (EpivizIGVTrack-class),  
[14](#)  
EpivizIGVTrack-class, [14](#)  
EpivizLinePlot (EpivizLinePlot-class),  
[14](#)  
EpivizLinePlot-class, [14](#)  
EpivizLineTrack  
(EpivizLineTrack-class), [14](#)  
EpivizLineTrack-class, [14](#)  
EpivizMeasurement, [8, 18](#)  
epivizNav, [15](#)  
EpivizNavigation, [6, 8, 15, 20, 22–25](#)  
EpivizNavigation  
(EpivizNavigation-class), [15](#)  
EpivizNavigation-class, [15](#)  
epivizrData::register, [8](#)  
EpivizScatterPlot  
(EpivizScatterPlot-class), [16](#)  
EpivizScatterPlot-class, [16](#)  
EpivizStackedBlocksTrack  
(EpivizStackedBlocksTrack-class),  
[17](#)  
EpivizStackedBlocksTrack-class, [17](#)  
EpivizStackedLinePlot  
(EpivizStackedLinePlot-class),  
[17](#)  
EpivizStackedLinePlot-class, [17](#)  
EpivizStackedLineTrack  
(EpivizStackedLineTrack-class),  
[17](#)  
EpivizStackedLineTrack-class, [17](#)  
EpivizViewComponent, [12](#)  
EpivizViewComponent  
(EpivizViewComponent-class), [18](#)  
EpivizViewComponent-class, [18](#)  
EpivizWebComponent, [11](#)  
EpivizWebComponent  
(EpivizWebComponent-class), [18](#)  
EpivizWebComponent-class, [18](#)

fromJSON, [20, 21](#)  
get\_available\_chart\_types, [19](#)

`get_registered_data_types`, [19](#)

`HeatmapPlot`, [20](#)  
`HeatmapPlot`, `EpivizEnvironment`-method  
    (`HeatmapPlot`), [20](#)

`json_parser`, [20](#)  
`json_writer`, [21](#)

`LinePlot`, [22](#)  
`LinePlot`, `EpivizEnvironment`-method  
    (`LinePlot`), [22](#)  
`LineTrack`, [22](#)  
`LineTrack`, `EpivizEnvironment`-method  
    (`LineTrack`), [22](#)

`minfi::blockFinder()`, [26](#), [27](#)

`plot`, `EpivizEnvironment`, ANY-method, [23](#)

`rand_id`, [23](#)

`ScatterPlot`, [24](#)  
`ScatterPlot`, `EpivizEnvironment`-method  
    (`ScatterPlot`), [24](#)  
`StackedBlocksTrack`, [24](#)  
`StackedBlocksTrack`, `EpivizEnvironment`-method  
    (`StackedBlocksTrack`), [24](#)  
`StackedLinePlot`, [25](#)  
`StackedLinePlot`, `EpivizEnvironment`-method  
    (`StackedLinePlot`), [25](#)  
`StackedLineTrack`, [25](#)  
`StackedLineTrack`, `EpivizEnvironment`-method  
    (`StackedLineTrack`), [25](#)

`sumexp`, [26](#)

`tcga_colon_blocks`, [26](#)  
`tcga_colon_curves`, [27](#)  
`tcga_colon_expression`, [27](#)  
`toJSON`, [21](#)