Package ‘gwavr’

May 11, 2024

Title  Get Water Attributes Visually in R

Version  0.3.1

Description  Provides methods to Get Water Attributes Visually in R (‘gwavr’). This allows the user to point and click on areas within the United States and get back hydrological data, e.g. flowlines, catchments, basin boundaries, comids, etc.

URL  https://github.com/joshualerickson/gwavr/

BugReports  https://github.com/joshualerickson/gwavr/issues/

License  MIT + file LICENSE

Encoding  UTF-8

RoxygenNote  7.3.1

Imports  dplyr, httr, jsonlite, leaflet, leaflet.extras, nhdplusTools, purrr, scales, sf, shiny, promises, miniUI, shinyWidgets, tidyr, units, utils, elevatr, whitebox, terra, htmlwidgets

Suggests  spelling, knitr, rmarkdown, testthat (>= 3.0.0)

Language  en-US

Config/testthat/edition  3

LazyData  true

NeedsCompilation  no

Author  Joshua Erickson [aut, cre]

Maintainer  Joshua Erickson <joshualerickson@gmail.com>

Depends  R (>= 3.5.0)

Repository  CRAN

Date/Publication  2024-05-10 22:53:11 UTC

R topics documented:

base_map  ......................................................... 2
basinMod  ........................................................ 2
basinModUI  ....................................................... 3
convert_sf_geocollection ........................................ 4
df_site_new ......................................................... 4
get_basin_interactively .......................................... 5
get_nhdplus_interactively ....................................... 6
get_nldi_interactively ........................................... 7
get_stream_network_interactively ............................... 8
get_usgs_dv_interactively ....................................... 10
get_usgs_iv_interactively ....................................... 11
nhdplusMod ......................................................... 12
nhdplusModUI ....................................................... 13
rename_geometry .................................................... 13
streamnetworkMod .................................................. 14
streamnetworkModUI ............................................... 15
usgsdvMod .......................................................... 15
usgsdvModUI ......................................................... 16
usgsinstMod ........................................................ 16
usgsinstModUI ....................................................... 17

Index 18

base_map  Base Map

Description
A generic leaflet base map used in the shiny apps.

Usage
base_map()

Value
A leaflet map with provider layers: "Esri.WorldImagery", "CartoDB.Positron", "OpenStreetMap", "CartoDB.DarkMatter", "OpenTopoMap" "Hydrography"

basinMod  Shiny Module Server for basin generation

Description
Shiny Module Server for basin generation

Usage
basinMod(input, output, session, values, dem, threshold = 1000, map, ...)

Description  A generic leaflet base map used in the shiny apps.

Usage
base_map()

Value
A leaflet map with provider layers: "Esri.WorldImagery", "CartoDB.Positron", "OpenStreetMap", "CartoDB.DarkMatter", "OpenTopoMap" "Hydrography"
basinModUI

Arguments

  input  Shiny server function input
  output Shiny server function output
  session Shiny server function session
  values A reactive Values list to pass
  dem    A raster or terra object dem.
  threshold A threshold for stream initiation. 1000 (default).
  map    a background leaflet or mapview map to be used for editing. If NULL a blank mapview canvas will be provided.
  ...    arguments to pass to wbt_* functions.

Value

  server function for Shiny module

basinModUI  Shiny Module UI for basin generation

Description

  A shiny Module to.

Usage

  basinModUI(id, ...)

Arguments

  id character id for the Shiny namespace
  ... other arguments to leafletOutput()

Value

  UI function for Shiny module
convert_sf_geocollection

*Convert GEOMETRYCOLLECTION to POLYGONS*

**Description**

Convert GEOMETRYCOLLECTION to POLYGONS

**Usage**

`convert_sf_geocollection(x)`

**Arguments**

`x`  
A sf object

**Value**

A converted sf object from GEOMETRYCOLLECTION to POLYGON or MULTIPOLYGON.

---

`df_site_new`  
*Retired USGS Sites*

**Description**

A subset of data that has retired USGS sites and coordinates.

**Usage**

`df_site_new`

**Format**

`df_site_new`:

A data frame with 14,966 rows and 4 columns:

- **SiteNumber**  Site Number
- **SiteName**  Site Name
- **location.coordinates1**  Longitude coordinates
- **location.coordinates2**  Latitude coordinates ...
get_basin_interactively

Get Watershed Basin Interactively

Description

This function allows the user to delineate watershed basins interactively with a shiny app. It uses the elevatr package to acquire the Digital Elevation Model (DEM) or user inputted DEM and whitebox package to delineate the basin (see details).

Usage

get_basin_interactively(
  map = NULL,
  ns = "basin-ui",
  viewer = shiny::paneViewer(),
  title = "Delineate Basin",
  dem = NULL,
  threshold = 1000,
  ...
)

Arguments

map         a background leaflet or mapview map to be used for editing. If NULL a blank mapview canvas will be provided.
ns          string name for the Shiny namespace to use. The ns is unlikely to require a change.
viewer      function for the viewer. See Shiny viewer. NOTE: when using browserViewer(browser = getOption("browser")) to open the app in the default browser, the browser window will automatically close when closing the app (by pressing "done" or "cancel") in most browsers. Firefox is an exception. See Details for instructions on how to enable this behaviour in Firefox.
title       string to customize the title of the UI window. The default is "Delineate Basin".
dem         A 'SpatRaster' object dem. (optional)
threshold   A threshold for stream initiation. 1000 (default).
...          other arguments to leafletOutput() in module and/or wbt_* functions.

Details

This function will throw an error if you don’t draw the bounding box (rectangle) first and you didn’t include your own DEM. Once the user has drawn the bounding box (or added own DEM) then you can use the marker as a pour point location.

Steps
get_nhdplus_interactively

1. Input a well-suited DEM zoom level and threshold. (skip if own DEM is inputted)
2. Draw bounding box (rectangle or polygon) (skip if own DEM is inputted).
3. Use marker to place pour point(s).
4. If necessary, change 'Cell Threshold' to change drainage density.
5. Repeat steps 1-4 if needed.
6. When finished, press 'done' and basins will be saved as a list in local environment.

In addition, this function uses both whitebox::wbt_feature_preserving_smoothing() and whitebox::wbt_breach_depressions() prior to running the flow direction and flow accumulation (both d8) algorithms.

Value
A sf object that contains watershed polygons the user collected during shiny session.

Note
The marker will only work for the most current stream raster. You can have multiple areas but you need to make sure that you are on the most current raster when selecting basins or the app will crash. If you add your own DEM then you don’t need to draw a bounding box.

Examples

if(interactive()){
  basin_data <- get_basin_interactively()
}

get_nhdplus_interactively

Get NHDPlus Interactively

Description
This function allows the user go get NHDPlus realizations interactively with a shiny app.

Usage

get_nhdplus_interactively(
  ns = "hydro-ui",
  viewer = shiny::paneViewer(),
  title = "NHDPlus",
  ...
)
get_nldi_interactively

Arguments

ns string name for the Shiny namespace to use. The ns is unlikely to require a change.

viewer function for the viewer. See Shiny viewer. NOTE: when using browserViewer(browser = getOption("browser")) to open the app in the default browser, the browser window will automatically close when closing the app (by pressing "done" or "cancel") in most browsers. Firefox is an exception. See Details for instructions on how to enable this behaviour in Firefox.

title string to customize the title of the UI window. The default is "NHDPlus".

... other arguments to leafletOutput() in module.

Value

A list of sf objects that the user collected during shiny session.

Note

The picker list has several options right now: NHDPlus Catchments, NHDPlus Flowlines, NHDPlus Waterbodies, NHDPlus Outlet, HUC 2-12, NWIS Site.

Examples

if(interactive()){
  nhdplus_data <- get_nhdplus_interactively()
}

get_nldi_interactively

Get Hydro Network-Linked Data Index (NLDI) Interactively

Description

This function uses the NLDI API to allow the user to visually select a location (point) to get numerous hydrologic realizations.

Usage

get_nldi_interactively()

Value

A list with sf objects.
**Note**

The picker list has three options right now: Total Basin, All Local Catchments and Only Local Catchment. Descriptions below:

- **Total Basin**: This will return the upstream tributaries (UT), upstream main (UM), basin boundary and site data above the user point.
- **All Local Catchments**: This will return the upstream tributaries (UT) and all the local NHD-PLusV2 catchments above the user point. In addition, each catchment will contain the zonal stats associated with 'CAT' in NLDI.
- **Only Local Catchment**: This will only return the catchment at the point and tributary. In addition, it will also include the zonal stat for that catchment.

**Examples**

```r
if(interactive()){
  nldi_data <- get_nldi_interactively()
}
```

---

**get_stream_network_interactively**

*Get Stream Network Interactively*

**Description**

This function allows the user to get stream networks and watersheds interactively with a shiny app. It uses the elevatr package to acquire the Digital Elevation Model (DEM) or user inputted DEM and whitebox package to delineate the stream network and watersheds (see details).

**Usage**

```r
get_stream_network_interactively(
  map = NULL,
  ns = "streamnetwork-ui",
  viewer = shiny::paneViewer(),
  title = "Streamnetwork",
  dem = NULL,
  threshold = 1000,
  ...
)
```

**Arguments**

- **map**: a background leaflet or mapview map to be used for editing. If NULL a blank mapview canvas will be provided.
- **ns**: string name for the Shiny namespace to use. The ns is unlikely to require a change.
**get_stream_network_interactively**

viewer function for the viewer. See Shiny viewer. NOTE: when using browserViewer(browser = getOption("browser")) to open the app in the default browser, the browser window will automatically close when closing the app (by pressing "done" or "cancel") in most browsers. Firefox is an exception. See Details for instructions on how to enable this behaviour in Firefox.

title string to customize the title of the UI window. The default is "Delineate Basin".

dem A raster or terra object dem. (optional)
threshold A threshold for stream initiation. 1000 (default).
... other arguments to leafletOutput() in module and/or wbt_* functions.

**Details**

This function uses the package **elevatr** to download the DEM (unless you provide your own). Once the user has drawn the bounding box or inputted DEM and selected appropriate zoom (resolution) and threshold then the app will create basins and streams.

**Steps**

1. Input a well-suited DEM zoom level and threshold. (skip if own DEM is inputted)
2. Draw bounding box (rectangle or polygon) (skip if own DEM is inputted).
3. Wait for layers to respond.
4. Repeat steps 1-4 if needed.
5. when finished, press 'done' and stream network and watersheds will be saved as a list in local environment with the associated flow accumulation and flow direction.

In addition, this function uses both whitebox::wbt_feature_preserving_smoothing() and whitebox::wbt_breach_depressions() prior to running the flow direction and flow accumulation (both d8) algorithms.

**Value**

A list of sf objects that the user collected during shiny session as well as flow accumulation and direction paths to tif. Each list will contain two sf objects: watersheds and streams. The streams object will also return these attributes: tribid, strahler, slope, length, mainstem, FID, STRM_VAL.

**Note**

If you add your own DEM then you don’t need to draw a bounding box.

**Examples**

```r
if(interactive()){
  streamnetwork <- get_stream_network_interactively()
}
```
get_usgs_dv_interactively

Get United States Geologic Survey (USGS) Daily Flow Values Interactively

---

Description

This function allows the user to select United States Geologic Survey (USGS) stations and get back daily flow values based on station selected. It uses the USGS Water Services to get the values as well as the USGS Sites.

Usage

get_usgs_dv_interactively(
  ns = "usgsdv-ui",
  viewer = shiny::paneViewer(),
  title = "Get USGS Daily Flow Values",
  ...
)

Arguments

ns string name for the Shiny namespace to use. The ns is unlikely to require a change.

viewer function for the viewer. See Shiny viewer. NOTE: when using browserViewer(browser = getOption("browser")) to open the app in the default browser, the browser window will automatically close when closing the app (by pressing "done" or "cancel") in most browsers. Firefox is an exception. See Details for instructions on how to enable this behaviour in Firefox.

title string to customize the title of the UI window. The default is "Get USGS Instantaneous Flow Values".

... other arguments to leafletOutput() in module.

Details

Steps

1. Select the sites you want to retrieve.
2. When finished, press 'done' and sites daily flow values will be saved to a data.frame in local environment.

Value

A data.frame that contains flow values based on the station(s) selected during shiny session.
get_usgs_iv_interactively

Note

You can select multiple stations. The information from the hover details is not included in the data.frame that is returned, e.g. rate of change, percentile description.

Examples

```r
if(interactive()){
  dv_usgs <- get_usgs_dv_interactively()
}
```

get_usgs_iv_interactively

Get United States Geologic Survey (USGS) Instantaneous Flow Values Interactively

Description

This function allows the user to select United States Geologic Survey (USGS) stations and get back instantaneous flow values based on control number of days from now. It uses the USGS Water Services to get the values as well as the USGS Dashboard to get current conditions (circle markers on map).

Usage

```r
get_usgs_iv_interactively(
  ns = "usgsiv-ui",
  viewer = shiny::paneViewer(),
  title = "Get USGS Instantaneous Flow Values",
  ...
)
```

Arguments

- **ns** string name for the Shiny namespace to use. The ns is unlikely to require a change.
- **viewer** function for the viewer. See Shiny viewer. NOTE: when using browserViewer(browser = getOption("browser")) to open the app in the default browser, the browser window will automatically close when closing the app (by pressing "done" or "cancel") in most browsers. Firefox is an exception. See Details for instructions on how to enable this behaviour in Firefox.
- **title** string to customize the title of the UI window. The default is "Get USGS Instantaneous Flow Values".
- **...** other arguments to leafletOutput() in module.
Details

Steps
1. Select the sites you want to retrieve.
2. Make sure you have the right days.
3. When finished, press 'done' and sites instantaneous flow values will be saved to a data.frame in local environment.

Value
A data.frame that contains flow values based on the station(s) selected during shiny session.

Note
You can select multiple stations but the number of days from now control will take the final number when you select the done button. The information from the hover details is not included in the data.frame that is returned, e.g. rate of change, percentile description.

Examples

```r
if(interactive()){
  iv_usgs <- get_usgs_iv_interactively()
}
```

---

**nhdplusMod**

*Shiny Module Server for nhdplus*

**Description**
Shiny Module Server for nhdplus

**Usage**

`nhdplusMod(input, output, session, values)`

**Arguments**

- **input**: Shiny server function input
- **output**: Shiny server function output
- **session**: Shiny server function session
- **values**: A reactive Values list to pass

**Value**

server function for Shiny module
Shiny Module UI for nhdplus

Description
A shiny Module to.

Usage
nhdplusModUI(id, ...)

Arguments
id character id for the the Shiny namespace
... other arguments to leafletOutput()

Value
UI function for Shiny module

rename_geometry Rename Geometry Column

Description
Rename Geometry Column

Usage
rename_geometry(g, name)

Arguments
g A sf object
name character.

Value
A sf object with a renamed geometry column.

Note
This function was grabbed from stack overflow from the legend spacedman.
streamnetworkMod  

**Shiny Module Server for stream networks**

**Description**

Shiny Module Server for stream networks

**Usage**

```r
streamnetworkMod(
  input,
  output,
  session,
  values,
  dem,
  threshold = 1000,
  map,
  ...
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Shiny server function input</td>
</tr>
<tr>
<td>output</td>
<td>Shiny server function output</td>
</tr>
<tr>
<td>session</td>
<td>Shiny server function session</td>
</tr>
<tr>
<td>values</td>
<td>A reactive Values list to pass</td>
</tr>
<tr>
<td>dem</td>
<td>A raster or terra object dem. (optional)</td>
</tr>
<tr>
<td>threshold</td>
<td>A threshold for stream initiation. 1000 (default).</td>
</tr>
<tr>
<td>map</td>
<td>a background leaflet or mapview map to be used for editing. If NULL a blank mapview canvas will be provided.</td>
</tr>
<tr>
<td>...</td>
<td>arguments to pass to wbt_* functions.</td>
</tr>
</tbody>
</table>

**Value**

server function for Shiny module
**streamnetworkModUI**

*Shiny Module UI for stream network generation*

**Description**

A shiny Module to.

**Usage**

```
streamnetworkModUI(id, ...)  
```

**Arguments**

- `id` character id for the the Shiny namespace
- `...` other arguments to `leafletOutput()`

**Value**

UI function for Shiny module

---

**usgsdvMod**

*Shiny Module Server for United States Geologic Survey (USGS) daily values*

**Description**

Shiny Module Server for United States Geologic Survey (USGS) daily values

**Usage**

```
usgsdvMod(input, output, session, values)  
```

**Arguments**

- `input` Shiny server function input
- `output` Shiny server function output
- `session` Shiny server function session
- `values` A reactive Values list to pass

**Value**

server function for Shiny module
usgsdvModUI  \hspace{1cm} Shiny Module UI for United States Geologic Survey (USGS) daily values

**Description**

A shiny Module to.

**Usage**

```r
usgsdvModUI(id, ...)
```

**Arguments**

- **id**: character id for the Shiny namespace
- **...**: other arguments to `leafletOutput()`

**Value**

UI function for Shiny module

usgsinstMod  \hspace{1cm} Shiny Module Server for United States Geologic Survey (USGS) instantaneous values

**Description**

Shiny Module Server for United States Geologic Survey (USGS) instantaneous values

**Usage**

```r
usgsinstMod(input, output, session, values)
```

**Arguments**

- **input**: Shiny server function input
- **output**: Shiny server function output
- **session**: Shiny server function session
- **values**: A reactive Values list to pass

**Value**

server function for Shiny module
usgsinstModUI

| usgsinstModUI | Shiny Module UI for United States Geologic Survey (USGS) instantaneous values |

**Description**

A shiny Module to.

**Usage**

`usgsinstModUI(id, ...)`

**Arguments**

- `id` character id for the the Shiny namespace
- `...` other arguments to `leafletOutput()`

**Value**

UI function for Shiny module
## Index

* **datasets**
  - df_site_new, 4

base_map, 2
basinMod, 2
basinModUI, 3

convert_sf_geocollection, 4
df_site_new, 4
elevatr, 9

get_basin_interactively, 5
get_nhdplus_interactively, 6
get_nldi_interactively, 7
get_stream_network_interactively, 8
get_usgs_dv_interactively, 10
get_usgs_iv_interactively, 11

nhdplusMod, 12
nhdplusModUI, 13

rename_geometry, 13

streamnetworkMod, 14
streamnetworkModUI, 15

usgsdvMod, 15
usgsdvModUI, 16
usgsinstMod, 16
usgsinstModUI, 17

viewer, 5, 7, 9–11