Package ‘quickmapr’

June 3, 2018

Type Package
Title Quickly Map and Explore Spatial Data
Version 0.3.0
URL https://www.github.com/jhollist/quickmapr
BugReports https://github.com/jhollist/quickmapr/issues
Maintainer Jeffrey W. Hollister <hollister.jeff@epa.gov>
License CC0

Description While analyzing geospatial data, easy visualization is often needed that allows for quick plotting, and simple, but easy interactivity. Additionally, visualizing geospatial data in projected coordinates is also desirable. The 'quickmapr' package provides a simple method to visualize 'sp', 'sf' (via coercion to 'sp'), and 'raster' objects, allows for basic zooming, panning, identifying, labeling, selecting, and measuring spatial objects. Importantly, it does not require that the data be in geographic coordinates.

Depends R (>= 3.0.0)
Imports raster, sp, rgeos, rgdal, httr, stats, grDevices, graphics, methods, sf
Suggests testthat, knitr
RoxygenNote 6.0.1
NeedsCompilation no
Author Jeffrey W. Hollister [aut, cre]
Repository CRAN
Date/Publication 2018-06-03 13:16:02 UTC

R topics documented:

buffer .................................................. 2
elev ..................................................... 2
examp_lakes ............................................ 3
Buffer of Lake Sunapee for quickmapr examples

Description
This example data is a SpatialPolygonsDataFrame representing a buffer of a single lake, Lake Sunapee.

Format
SpatialPolygonsDataFrame

elevation Data for quickmapr examples

Description
This example data is a RasterLayer of the National Elevation Dataset for a buffer of Lake Sunapee.

Format
RasterLayer of 455 x 246 and 30 meter resolution
Example data is a polygon dataset of many lakes in New Hampshire. Data for each lake are from the first version of the National Lakes Morphometry Dataset.

**Format**

SpatialPolygonsDataFrame

---

`f`  

*Returns qmap object to original extent*

---

## Description

Allows for restoring map to original extent. The initial map is replayed and the map extent of the object is reset to match the current display.

### Usage

```r
def(qmap_obj = NULL)
```

### Arguments

- `qmap_obj` a qmap_obj to restore

### Examples

```r
## Not run:  
data(lake)  
x<-qmap(list(lake,buffer,elev))  
z1(x)  
f(x)  
## End(Not run)
```


### Identify

**Description**

Interactively select an sp or raster object and print the data associated with it. `i()` only accepts a single input point.

**Usage**

```r
i(qmap_obj = NULL, i_idx = 1, loc = NULL)
```

**Arguments**

- `qmap_obj`: a qmap object from which to identify features.
- `i_idx`: a numeric value specifying which data layer to identify or a character string indicating the name of the layer. Defaults to 1.
- `loc`: A list with an x and y numeric indicating a location. Default is to interactively get loc value until escaped.

**Value**

Returns NULL. Identified values are printed to the screen.

**Examples**

```r
## Not run:
data(lake)
qm <- qmap(list(lake, elev, samples))
i(qm, "lake")
i(qm, "samples")
i(qm, 2)
## End(Not run)
```

---

### Label features

**Description**

It is useful to be able to provide labels to features on your map when examining the results of given analysis. This function adds labels for a given layer (currently just point or polygon layers).

**Usage**

```r
l(qmap_obj, field = NULL, layer = 1)
```
Arguments

- `qmap_obj`: a `qmap` object from which to pull the labels. Raster layers are ignored. Will also accept `sp` objects.
- `field`: a field in the `sp` object to use to label the features. Defaults to `row.names()`.
- `layer`: identify which `sp` layer to label. Defaults to first layer in `qmap_obj$map_data`.

Examples

```r
## Not run:
data(lake)
qm <- qmap(lake, width, buffer)
l(qm, 'COMID')
l(qm, layer = 2)
l(qm, layer = "buffer")

## End(Not run)
```

Lake Data for `quickmapr` examples

This example data is a `SpatialPolygonsDataFrame` of a single lake, Lake Sunapee.

Format

`SpatialPolygonsDataFrame` with 1 lakes, each with 13 variables

Maximum Lake Length line for `quickmapr` examples

This example data is a line dataset representing the maximum lake length.

Format

`SpatialLines`
\section*{m \hspace{1cm} Measure}

\textbf{Description}

Allows interactive selection of points and returns length of selecte line in units of the current qmap object.

\textbf{Usage}

\texttt{m(qmap_obj = NULL)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{qmap_obj} \hspace{1cm} a qmap object from which to measure features.
\end{itemize}

\section*{nlcd \hspace{1cm} Land Cover Data for \texttt{quickmapr} examples}

\textbf{Description}

This example data is a RasterLayer of the National Land Cover Dataset for a buffer of Lake Sunapee.

\textbf{Format}

RasterLayer of 467 x 262 and 30 meter resolution

\section*{p \hspace{1cm} Pan the current plot}

\textbf{Description}

Interactively reposition the current plot. Works on an existing qmap object. Simply pass that object to \texttt{p()}. A single repositioning results and the extent of the qmap object is changed.

\textbf{Usage}

\texttt{p(qmap_obj = NULL, loc = NULL)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{qmap_obj} \hspace{1cm} A qmap object. Optional, but performs better with larger data sets.
  \item \texttt{loc} \hspace{1cm} A list with an x and y numeric indicating a location. Default is to interactively get loc value until escaped.
\end{itemize}
Examples

```r
## Not run:
data(lake)
x <- qmap(list(lake, buffer, elev))
p()  
## Or
p(x)

## End(Not run)
```

---

**plot.qmap**

*Default plotting of a qmap object*

**Description**

Plots the qmap class and uses the order, colors, extent, and fill option from `qmap`.

**Usage**

```r
## S3 method for class 'qmap'
plot(x, ...)
```

**Arguments**

- `x` : input qmap class to plot
- `...` : options passed to image or plot

---

**print.qmap**

*Default plotting of a qmap object*

**Description**

Plots a qmap object

**Usage**

```r
## S3 method for class 'qmap'
print(x, ...)```

**Arguments**

- `x` : input qmap class to print
- `...` : options passed to plot
Build qmap object

Description
This function builds the qmap object that forms the basis for the rest of the quickmapr package.

Usage
qmap(..., extent = NULL, order = 1:length(mapdata), colors = 1:length(mapdata), fill = FALSE, prj = TRUE, basemap = c("none", "1m_aerial", "topo"), resolution = 300)

Arguments
- ... Spatial objects to map. Maybe passed as objects, a list of spatial objects, or a qmap object
- extent A sp or raster object to use as the initial extent of the map. Defaults to the maximum extent of all input object
- order draw order of the spatial object. Defaults to order in mapdata
- colors line colors. Defaults to 1:length(mapdata)
- fill Logical to determine if polygons should be filled (using colors) or just the border colored.
- prj Logical to check projections of input spatial objects. Transformation, if needed, should be done prior to mapping with sp::spTransform().
- basemap a basemap generated from get_basemap
- resolution Specifies the width in pixels of the retrieved basemap. Larger values result in higher resolution images but since the images are downloaded for each zoom level can result in delays. Default is 300, while ~600 is a decent compromise for performance and image quality.

Value
Function displays a map from the input mapdata parameter and returns a recorded plot.

Examples
```r
## Not run:
data(lake)
mymap<-list(elev,lake,buffer,length,samples)
qm<-qmap(mymap)
#change draw order and which data is displayed
qmap(qm,order=c(2,3,5))
#add a basemap
qm<-qmap(qm,basemap="1m_aerial", resolution = 800)
## End(Not run)
```
Description

This package provides tools to quickly visualize and interact with \texttt{sp} and \texttt{raster} objects. Simple interactivity includes panning, zooming, identifying, and labelling. The goal of \texttt{quickmapr} is to make working with spatial data easy and quick, it is not intended to provide publication quality maps.

Select

Description

Interactively select a single \texttt{sp} or \texttt{raster} object and return the object. \texttt{s()} only accepts a single input point.

Usage

\begin{verbatim}
s(qmap_obj = NULL, s_idx = 1L, loc = NULL)
\end{verbatim}

Arguments

- \texttt{qmap_obj} a \texttt{qmap} object from which to select features.
- \texttt{s_idx} a numeric value specifying which data layer to select or a character string indicating the name of the layer. Defaults to 1.
- \texttt{loc} A list with an x and y numeric indicating a location. Default is to interactively get loc value.

Value

Returns a selected \texttt{sp} object

Examples

\begin{verbatim}
## Not run:
data(life)
qm<-.qmap(list(life,elev,samples))
s(qm,"life")
s(qm,3)
## End(Not run)
\end{verbatim}
samples  

*Point data for quickmapr examples*

**Description**

This is a fabricated point dataset for what might be a common type of data collected for a lake. This example data is a fabricated point dataset with data for each location.

**Format**

SpatialPointsDataFrame

width  

*Maximum lake width for quickmapr examples*

**Description**

This example data is a line dataset representing the maximum lake width. Maximum lake width is defined as the longest line perpendicular to the maximum lake length.

**Format**

SpatialLines

**ze**  

*Zooms in on extent*

**Description**

Select a bounding box interactively and zoom to that extent. Works on an existing qmap object. Simply pass that object to `ze()`. A single zoom in to the selected extent results and the extent of the qmap object is changed.

**Usage**

`ze(qmap_obj = NULL, extent = NULL)`

**Arguments**

- `qmap_obj`  
  A qmap object. Optional, but performs better with larger data sets.

- `extent`  
  A Spatial* object to specify extent to zoom into.
Examples

```r
## Not run:
data(lake)
qmap(list(lake, buffer, elev))
ze()

## End(Not run)
```

---

**zi**

*Zooms in on current plot*

---

**Description**

Interactively zoom in on the current plot. Works on an existing `qmap` object. Simply pass that object to `zi()`. A single zoom in results and the extent of the `qmap` object is changed.

**Usage**

```r
zi(qmap_obj = NULL, zoom_perc = 0.5, loc = NULL)
```

**Arguments**

- `qmap_obj`: A `qmap` object. Optional, but performs better with larger data sets.
- `zoom_perc`: A proportion to determine the zoom level. The x and y axes are reduced by this amount. Default is 0.5.
- `loc`: A list with an x and y numeric indicating a location. Default is to interactively get loc value until escaped.

**Examples**

```r
## Not run:
data(lake)
x<-qmap(list(lake, buffer, elev))
zi(x)

## End(Not run)
```
Description

Interactively zoom out on the current plot. Works on an existing qmap object. Simply pass that object to zo(). A single zoom out results and the extent of the qmap object is changed.

Usage

zo(qmap_obj = NULL, zoom_perc = 0.5, loc = NULL)

Arguments

qmap_obj A qmap object. Optional, but performs better with larger data sets.
zoom_perc A proportion to determine the zoom level. The x and y axes are increased by this amount. Default is 0.5.
loc A list with an x and y numeric indicating a location. Default is to interactively get loc value until escaped.

Examples

## Not run:
data(lake)
qm<-qmap(list(lake$buffer,elev))
zo(qm)

## End(Not run)
Index

*Topic **datasets**
  buffer, 2
  elev, 2
  examp_lakes, 3
  lake, 5
  length, 5
  nlcd, 6
  samples, 10
  width, 10

buffer, 2

elev, 2
examp_lakes, 3

f, 3

get_basemap, 8

i, 4

l, 4
lake, 5
length, 5

m, 6

nlcd, 6

p, 6
plot.qmap, 7
print.qmap, 7

qmap, 8, 8
quickmapr, 8, 9, 9

raster, 8, 9

s, 9
samples, 10
sp, 8, 9

width, 10
ze, 10
zi, 11
zo, 12