

geo in R

Jacob Kasper

April 4, 2016

1 Why geo?

Geo was developed for spatial plotting in and around Iceland by Hóskuldur Björnsson at the Marine Research Institute in Iceland. It is a useful package which is easy to learn and has a whole host of functions which are quite intuitive. Also is the `geoextras` package which expands on the functionality of `geo`. `Geo` is currently maintained and hosted on github <https://github.com/Hafro/geo>

```
> #<<fig=TRUE,echo=FALSE>>=
> # install.packages("devtools")
> # I had to install Rtools
> # install rtools 3.3 http://cran.r-project.org/bin/windows/Rtools
> # install.packages("maps")
> # devtools::install_github("hafro/geo")
> #install.packages("geoextras", repos="http://r-forge.r-project.org", type="source")
> library(mapdata)
> library(geo)
> library(geoextras)
> options(bg="white")
> geoplots()
> geoplots(xlim = list(lat = c(40, 43), lon = c(-68, -72)),
+         country = "none", cex=.5, projection = "Lambert", grid=FALSE)
> mygrd <- list(lat=seq(40,43, by=1), lon=seq(-68,-72, by=-1))
> geogrid(mygrd, lwd=.5, col="lightgrey")
> geoworld(database = "worldHires", allowed.size = 1e6)
> ##x <- geocode() ##or use any dataframe....
> ##you need to click on the plot to use this function,
> ##it makes a dataframe then press esc to save the dataframe
> x<-data.frame(lat=c( 40.1, 41 , 40.1 ), lon=c(-71, -71, -69, -69))
> geopolygon(x, col="orange")
> labloc<-list(lat=c(41,40),lon=c(-70,-69))
> geolegend(labloc, c("my area"), col=c("deeppink"), pt.cex=2, pch=20, bty="n")
> geopoints(41,-70, col="red", cex=2)
> geolines(c(41, 42),c(-70, -69), col="blue")
> arcdist(41, -70, 42, -69) ## defaults to nmi

[1] 74.96458

> Arrow(list(lat=c(41, 41.5), lon=c(-70.5, -71)), angle = 15, col = 2)
>
```

2 What else can geo do?

Data manipulations

`geo.Split.poly()`: Split a polygon into a list

`geoconvert()`: Convert between different representations of latitude and longitude, namely degrees-minutes-decimal minutes and decimal degrees.

geoconvert.1() Vector of decimal-minute-decimal minutes and returns converted value in decimal degrees
 geoconvert.2() Convert from decimal degrees to degrees, minutes and fractional minutes representation (DDMMmm) of lat or lon.

These are just a few of the useful functions, there are many many more build into this package, some of which are mentioned in this table.

geoplot	plot	For setting up the map
geopoints	points	adding points on a map
geolines	lines	adding lines on a map
geosymbols	symbols	adding symbols on a map
geotext	text	adding text on a map
geopolygon	polygon	adding a polygon on a map
geolocator	locator	getting positions on the map by clicking the mouse
geoidentify	identify	identifying a point in a dataset be the mouse
geoarea		calculating size of an area
geoaxis	axis	plot axis if geoplot is called with axlabels=F
geocontour	contour	plotting contour lines
geocontour.fill		plotting filled countours
geodefine		defining a closed area
geointersect		calculating the intersection or union of two polygons
gbplot		plotting depth contours
geoworld		plotting coastlines
geoexpand	expand.grid	expanding a grid
geoimage	image	plotting an image (rarely used).
geoinside		find the points inside a polygon
pointkriging		surface fitting using kriging
variogram		calculations of variograms
variofit		fitting of variogram
SMB.std.background		setting up plot that fits the SMB area
arc.dist		geographic distance
lodist		geograpich distance using the mouse
litir		plot a chessboard that shows the colors
The use of these programs is best described by an example		