

Package ‘geostRuct’

April 18, 2013

Title: Geographical genetic structure maps

Version: 0

Authors: Kimberly J. Gilbert & Peter D. Fields

Description: An R package to visualize genetic structure data on geographic maps. Genetic clustering data from commonly used programs such as Structure, easily visualized in geographic space as pie charts or bar plots. Other types of frequency data are also equally applicable. Builds off of existing functions in ‘mapplots’.

Notes

Purpose:

- Visualize

 - Pie charts on a map

 - Bar plots on a map

 - Could try to interface with R googlemaps too, in order to plot on real maps

Target audience:

- Population geneticists

- Users of structure

Users of anyone with frequency data in space

To figure out:

Format of input data

.txt and .csv

Number of columns limited

Titles of columns best to be set, at least for lat lon, maybe type1 type2 etc for pie slice columns We definitely want to constrain this for simplicity. We might do something like k.1, or some other replacement for k thats less associated with structure.

Packages to call

-see pseudocode

Format of output image

.pdf .jpeg .tiff

Size of output image will either have to set this constant or have point size change with it, can make standard point size a fraction of final image size? Probably easiest that way

Colors

Text labels - user option

Legend - user option

Transparency - user option for percent transparent

Size scaling - user option, based on total pop size or all points same size

Prevent point overlapping - user option

Map scale bar - user option

1 Pseudocode

Function will have a set of default settings that users can edit when desired.

I. Load packages

- a. Maps
- b. Mapdata
- c. Maptools
- d. Mapplots add.pie
- e. Scales - transparency

II. Read data

- a. Data file needs columns:
 - i. Latitude
 - ii. Longitude
 - iii. Total (optional, could be pop size or sample size at a point)
 - iv. Slice 1
 - v. Slice 2
 - vi. Slice 3, 4, (past 2 optional)
 - vii. ID (optional, for text if desired)
- b. Read in data function should have a setting to specify whether its taking a .txt file or a .csv file — or might be equivalent; if its not, can make a conversion function once data is in
 - i. `read.txt()` and `read.csv()`
 - ii. Once one of those reads in the data, convert to the same format, each conversion will have its own function
- c. Additionally, have an option to read data from CLUMPP format
 - i. .pop files should be sufficient for pie charts
 - ii. .ind files could work for bar plots
 - iii. use could have a second data file where gps points correspond to the numbered pops in the clump outputs
 1. `match()` function can be used here for pairing up data by numerical IDs

III. Make map

- a. Extent to cover
 - i. Based on GPS data read in see the `basic.map()` function in `mapplots`
 - ii. Color
 - iii. Probably want to base final output file size on this extent

IV. Add points

- a. Solid or transparent, border or no border
- b. Size scaled or constant
 - i. Need to set base size, probably determined proportionally from output image size, or let user decide
- c. Colors
- d.

V. Add barplots — another function, get to this later after working out the pies; `mapplots` package also has useful code for making barplots on top of existing plots!

- a. Position
- b. Colors
- c. Size
- VI. Add extras
 - a. Text
 - i. Size
 - ii. Position
 - b. Legend
 - i. Location
 - ii. Size
 - c. Scale bar
 - i. Location
 - ii. Size
 - d. box()
- VII. Output file — maybe see somehow if this can be kept separate? Or should it be innate in a way similar to RgoogleMaps has pros and cons either way
 - a. Format
 - i. .pdf
 - ii. .tiff
 - iii. .jpeg
 - b. File path to send to

To Do

Come up with a range of phony sample files

Must cover the range of the globe, and a range of zoomed in vs zoomed out, and a range of number of colored slices

Peter is working on non-overlapping points

Kim is starting on coding the function

Remaining:

get data to test

first function

first function

second function

second function