

Package ‘antarcticR’

April 28, 2017

Title What the Package Does (one line, title case)

Version 0.0.0.9000

Description What the package does (one paragraph).

Depends R (>= 3.3.3)

License What license is it under?

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1.9000

Author Luke Batten [aut, cre]

Maintainer Luke Batten <lukeb@physics.org>

RemoteType local

RemoteUrl /home/berg/Dropbox/LinuxSync/PhD/ANITA/2017Stuff/clusterDir/antarcticR

RemoteSha NA

RemoteBranch master

RemoteUsername NA

RemoteRepo NA

R topics documented:

clusterResult	2
csvToDF	2
csvToHaversineMat	3
drawAntarctica	3
genHaversineMat	4
plotAntarctica	4

Index	6
--------------	----------

clusterResult	<i>A function to use some clustering methods from the dbscan package</i>
---------------	--

Description

A function to use some clustering methods from the dbscan package

Usage

```
clusterResult(haversineMatrix, eps = 2e+05, minPts, eps_cl)
```

csvToDF	<i>Turn a longitude, latitude csv file into a dataframe</i>
---------	---

Description

Generate a dataframe from a longitude-latitude csv file

Usage

```
csvToDF(csvFile)
```

Arguments

csvFile	Your csv file
---------	---------------

Value

A dataframe

Examples

```
df <- csvToHaversineMat("myData.csv")
```

csvToHaversineMat	<i>A function to generate a Haversine matrix from a csv file</i>
-------------------	--

Description

Generate a distance matrix of great-circle distances from a csv file with longitude and latitude distances

Usage

```
csvToHaversineMat(csvFile)
```

Arguments

csvFile	Your csv file
---------	---------------

Value

A haversine distance matrix

Examples

```
mat <- csvToHaversineMat("myData.csv")
```

drawAntarctica	<i>Set up the drawing of a map of Antarctica</i>
----------------	--

Description

Set up the drawing of a map of Antarctica

Usage

```
drawAntarctica()
```

Examples

```
world3 <- drawAntarctica()  
world3
```

genHaversineMat	<i>A function to generate a Haversine matrix from a dataframe</i>
-----------------	---

Description

Generate a distance matrix of great-circle distances from a dataframe with longitude and latitude distances

Usage

```
genHaversineMat(df)
```

Arguments

df	Your data frame
----	-----------------

Value

A haversine distance matrix

Examples

```
points <- read.csv("dividedEvents1.csv",header=T, sep=",")
df.points <- as.matrix(points)
antFrame = data.frame(df.points)
print("Computing distance matrix...")
require(geosphere)
d <- genHaversineMat(antFrame)
```

plotAntarctica	<i>Plot points on the antarctic map</i>
----------------	---

Description

Plot points on the antarctic map

Usage

```
plotAntarctica(antMap, df, cluster = FALSE)
```

Arguments

antMap	your map made from drawAntarctica
df	Your data frame

Examples

```
world4 <- plotAntarctica(map, dataFrame)
world4
```

Index

- *Topic **Antarctica**,
 - clusterResult, [2](#)
 - drawAntarctica, [3](#)
 - plotAntarctica, [4](#)
- *Topic **Haversine**,
 - csvToHaversineMat, [3](#)
 - genHaversineMat, [4](#)
- *Topic **csv**,
 - csvToHaversineMat, [3](#)
- *Topic **dataframe**,
 - genHaversineMat, [4](#)
- *Topic **dataframe**
 - csvToDF, [2](#)
- *Topic **distance**
 - csvToHaversineMat, [3](#)
 - genHaversineMat, [4](#)
- *Topic **draw**,
 - clusterResult, [2](#)
 - drawAntarctica, [3](#)
 - plotAntarctica, [4](#)
- *Topic **matrix**,
 - csvToHaversineMat, [3](#)
 - genHaversineMat, [4](#)
- *Topic **plotx**
 - plotAntarctica, [4](#)
- *Topic **plot**
 - clusterResult, [2](#)
 - drawAntarctica, [3](#)

clusterResult, [2](#)
csvToDF, [2](#)
csvToHaversineMat, [3](#)

drawAntarctica, [3](#)

genHaversineMat, [4](#)

plotAntarctica, [4](#)