Package 'pointdensityP'

July 4, 2018

Title Point Density for Geospatial Data

Version 0.3.4

Author ``Paul Evangelista <paul.evangelista@usma.edu> and Dave Beskow <david.beskow@usma.edu>"

Maintainer Paul Evangelista <paul.evangelista@usma.edu>

Description The function pointdensity returns a density count and the temporal average for every point in the original list. The dataframe returned includes four columns: lat, lon, count, and date_avg. The ``lat" column is the original latitude data; the ``lon" column is the original longitude data; the ``count" is the density count of the number of points within a radius of radius*grid_size (the neighborhood); and the date_avg column includes the average date of each point in the neighborhood.

Depends R (>= 3.4.0)

Imports data.table, utils

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-07-04 16:20:03 UTC

R topics documented:

Arigon *Arigon dataset*

Description

A mock dataset containing meaningless events in a fictional state of Arigon (data overlays Oregon).

- latitude. Latitude of event.
- longitude. Longitude of event.
- date. Date of event.

Usage

```
data(Arigon)
```

Format

A data frame with 80000 rows and 3 variables

Author(s)

LTC Steve Henderson and The Department of Systems Engineering at West Point

clean_crime *Houston crime dataset*

Description

Lightly cleaned Houston crime; no NA events included and all dates recognized by `pointdensity`; data from January 2010 to August 2010 geocoded with Google Maps and courtesy of **ggmap**

Author(s)

Houston Police Department, City of Houston

References

http://www.houstontx.gov/police/cs/stats2.htm

pointdensity *Point density function for geospatial data*

Description

This function maps a dataset of geospatial points to a regular grid and calculates the density and temporal average of the points.

Usage

```
pointdensity(df, lat_col, lon_col, date_col = NULL, grid_size, radius)
```

Arguments

df	Data frame minimally containing latitude and longitude of spatial point data
lat_col	name of column in df that contains latitude or vertical dimension of data
lon_col	name of column in df that contains longitude or horizontal dimension of data
date_col	name of column in df that contains date associated with the event
grid_size	distance in kilometers between the grid lines that will support discretization of data and density reference
radius	distance in kilometers that represents the local neighborhood where an event adds density

Details

pointdensity returns a density count and the temporal average for every point in the original list. The dataframe returned includes four columns: lat, lon, count, and date_avg. The "lat" column is the original latitude data; the "lon" column is the original longitude data; the "count" is the density count of the number of points within a defined radius (the neighborhood); and the date_avg column includes the average date of each point in the neighborhood. Designed specifically for geospatial point processes and originally developed for military applications, this technique applies to any geospatial point process where there is a desire for an explainable measurement of density and maintaining fidelity of the original point locations. Typical spatial density plotting algorithms, such as kernel density estimation, implement some type of smoothing function that often results in a density value that is difficult to interpret. pointdensity was designed for ease of interpretation. Potential applications include analysis of military events, crime, and real estate transactions. An example follows with the Arigon data using **ggmap** (recommended) for visualization:

```
Arigon_density <- pointdensity(df = Arigon, lat_col = "latitude", lon_col = "longitude",
date_col = "date", grid_size = 1, radius = 2)
map_base <- qmap(location="44.12,-120.83", zoom = 7, darken=0.3)
map_base + geom_point(aes(x = lon, y = lat, colour = count), shape = 16, size = 2,
data = Arigon_density) + scale_colour_gradient(low = "green", high = "red")
```

Here is another example using the crime dataset from **ggmap**:

```
H_crime <- pointdensity(df = clean_crime, lat_col = "lat", lon_col = "lon",
grid_size = 1, radius = 4)
map_base <- qmap(location="29.76,-95.42", zoom = 11, darken=0.3)
map_base + geom_point(aes(x = lon, y = lat, colour = count), shape = 16, size = 2,
data = H_crime) + scale_colour_gradient(low = "green", high = "red")
```

Author(s)

Paul Evangelista <paul.evangelista@usma.edu>

David Beskow <david.beskow@usma.edu>

References

Wand, M. P. (1994). Fast Computation of Multivariate Kernel Estimators. *Journal of Computational and Graphical Statistics*, 3, 433-445.

Examples

```
Arigon_test <- Arigon[1:1000,]
Arigon_density <- pointdensity(df = Arigon_test, lat_col = "latitude",
lon_col = "longitude", date_col = "date", grid_size = 1, radius = 2)
```

Index