Package 'distRcpp'

November 6, 2016

Type Package

Title A Package To Compute Distance and Distance-Weighted Measures Using Rcpp

Version 0.5.1

Date 2016-11-06

Author Benjamin Skinner <b.skinner@vanderbilt.edu>

Maintainer Benjamin Skinner <b.skinner@vanderbilt.edu>

Description This package computes distance-weighted measures using Rcpp to speed up computations. It also computes simple distances.

License GPL (>= 2)

Imports Rcpp (>= 0.12.7)

LinkingTo Rcpp

RoxygenNote 5.0.1

NeedsCompilation yes

R topics documented:

`distRcpp` *A Package To Compute Distance and Distance-Weighted Measures Using Rcpp*

Description

This package computes distance-weighted measures using Rcpp to speed up computations. It also computes simple distances.

dist_1to1 *Compute one to one distance.*

Description

Compute distance between two points (one to one) and return single value.

Usage

```
dist_1to1(xlon, xlat, ylon, ylat, funname)
```

Arguments

xlon	Longitude for starting coordinate pair
xlat	Latitude for starting coordinate pair
ylon	Longitude for ending coordinate pair
ylat	Latitude for ending coordinate pair
funname	String name of distance function: Haversine, Vincenty

Value

Distance in meters

dist_1tom *Compute one to many distances.*

Description

Compute distances between single starting coordinate and vector of ending coordinates (one to many) and return vector.

Usage

```
dist_1tom(xlon, xlat, ylon, ylat, funname)
```

Arguments

xlon	Longitude for starting coordinate pair
xlat	Latitude for starting coordinate pair
ylon	Vector of longitudes for ending coordinate pairs
ylat	Vector of latitudes for ending coordinate pairs
funname	String name of distance function: Haversine, Vincenty

Value

Vector of distances in meters

dist_df — *Compute distance between corresponding coordinate pairs in data frame.*

Description

Compute distance between corresponding coordinate pairs and return vector. For use when creating a new data frame or tbl_df column.

Usage

```
dist_df(xlon, xlat, ylon, ylat, funname)
```

Arguments

xlon	Vector of longitudes for starting coordinate pairs
xlat	Vector of latitudes for starting coordinate pairs
ylon	Vector of longitudes for ending coordinate pairs
ylat	Vector of latitudes for ending coordinate pairs
funname	String name of distance function: Haversine, Vincenty

Value

Vector of distances between each coordinate pair in meters

dist_min — *Find minimum distance.*

Description

Find minimum distance between each starting point in **x** and possible end points, **y**.

Usage

```
dist_min(x_df, y_df, x_id = "id", x_lon_col = "lon", x_lat_col = "lat",
  y_lon_col = "lon", y_lat_col = "lat", dist_function = "Haversine")
```

Arguments

x_df	DataFrame with coordinates that need weighted measures
y_df	DataFrame with coordinates at which measures were taken
x_id	String name of unique identifer column in x_df
x_lon_col	String name of column in x_df with longitude values
x_lat_col	String name of column in x_df with latitude values
y_lon_col	String name of column in y_df with longitude values
y_lat_col	String name of column in y_df with latitude values
dist_function	String name of distance function: "Haversine" (default) or "Vincenty"

Value

DataFrame with minimum distance in meters

`dist_mtom` *Compute distance between each coordinate pair (many to many) and return matrix.*

Description

Compute distance between each coordinate pair (many to many) and return matrix.

Usage

```
dist_mtom(xlon, xlat, ylon, ylat, funname)
```

Arguments

`xlon`	Vector of longitudes for starting coordinate pairs
`xlat`	Vector of latitudes for starting coordinate pairs
`ylon`	Vector of longitudes for ending coordinate pairs
`ylat`	Vector of latitudes for ending coordinate pairs
`funname`	String name of distance function: Haversine, Vincenty

Value

Matrix of distances between each coordinate pair in meters

`dist_weighted_mean` *Interpolate inverse-distance-weighted measures.*

Description

Interpolate inverse-distance-weighted measures for each **x** coordinate using measures taken at surrounding **y** coordinates. Ending measures are weighted by inverse distance so that surrounding measures taken in nearby areas are given more weight in final average.

Usage

```
dist_weighted_mean(x_df, y_df, measure_col, x_id = "id", x_lon_col = "lon",
  x_lat_col = "lat", y_lon_col = "lon", y_lat_col = "lat",
  dist_function = "Haversine", dist_transform = "level", decay = 2)
```

Arguments

x_df	DataFrame with coordinates that need weighted measures
y_df	DataFrame with coordinates at which measures were taken
measure_col	String name of measure column in y_df
x_id	String name of unique identifer column in x_df
x_lon_col	String name of column in x_df with longitude values
x_lat_col	String name of column in x_df with latitude values
y_lon_col	String name of column in y_df with longitude values
y_lat_col	String name of column in y_df with latitude values
dist_function	String name of distance function: "Haversine" (default) or "Vincenty"
dist_transform	String value of distance weight transform: "level" (default) or "log"
decay	Numeric value of distance weight decay: 2 (default)

Value

Dataframe of distance-weighted values

---

popdist_weighted_mean *Interpolate population/inverse-distance-weighted measures.*

Description

Interpolate population/inverse-distance-weighted measures for each **x** coordinate using measures taken at surrounding **y** coordinates. Ending measures are double weighted by population and distance so that surrounding measures taken in nearby areas and those with greater populations are given more weight in final average.

Usage

```
popdist_weighted_mean(x_df, y_df, measure_col, x_id = "id",
  x_lon_col = "lon", x_lat_col = "lat", y_lon_col = "lon",
  y_lat_col = "lat", pop_col = "pop", dist_function = "Haversine",
  dist_transform = "level", decay = 2)
```

Arguments

x_df	DataFrame with coordinates that need weighted measures
y_df	DataFrame with coordinates at which measures were taken
measure_col	String name of measure column in y_df
x_id	String name of unique identifer column in x_df
x_lon_col	String name of column in x_df with longitude values
x_lat_col	String name of column in x_df with latitude values
y_lon_col	String name of column in y_df with longitude values
y_lat_col	String name of column in y_df with latitude values
pop_col	String name of column in x_df with population values
dist_function	String name of distance function: "Haversine" (default) or "Vincenty"
dist_transform	String value of distance weight transform: "level" (default) or "log"
decay	Numeric value of distance weight decay: 2 (default)

Value

Dataframe of population/distance-weighted values

Index