Package: GWpcor (via r-universe)

May 24, 2026

Type Package

Title Geographically Weighted Partial Correlation Coefficient

Version 0.1.7

Description Implements a geographically weighted partial correlation which is an extension from gwss() function in the 'GWmodel' package (Percival and Tsutsumida (2017) <doi:10.1553/giscience2017_01_s36>).

License GPL-3 + file LICENSE

Encoding UTF-8

LazyData true

Language en-US

Depends R (>= 3.5.0)

Imports methods, dplyr, sp, sf, geodist, pracma, corpcor, foreach, parallel, doParallel

SystemRequirements C++11, GDAL (>= 2.0.1), GEOS (>= 3.4.0), PROJ (>= 4.8.0)

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

Repository https://gwpcor.r-universe.dev

Date/Publication 2021-11-19 01:59:59 UTC

RemoteUrl https://github.com/gwpcor/gwpcor

RemoteRef HEAD

RemoteSha bcb057ef77a56ff9d0040ab85b5fd4ac0f30d9cf

Contents

gwpcor *Geographically Weighted Correlation and Partial Correlation*

Description

This function calculates the geographically weighted correlation and partial correlation between two variables given others. The function is designed by the gwss function from the GWmodel package, and the cor2pcor function from the corpcor package.

Usage

```
gwpcor(sdata, res_dp, vars, method = c("pearson", "spearman"),
kernel = "bisquare", adaptive = FALSE, bw, dMat,
geodisic_measure = "cheap", foreach = FALSE)
```

Arguments

sdata	a Spatial*DataFrame (i.e. SpatialPointsDataFrame or SpatialPolygonsDataFrame as defined in package sp), or a sf object.
res_dp	A Spatial*DataFrame object for providing summary locations, i.e. SpatialPointsDataFrame or SpatialPolygonsDataFrame as defined in package sp, or a sf object.
vars	A vector of variable names to be used for the analysis.
method	A character string indicating which correlation and partial correlation coefficients to compute. "pearson" or "spearman" are accepted.
kernel	function chosen as follows: gaussian: wgt = exp(-0.5 * (vdist / bw)^2); exponential: wgt = exp(-vdist / bw); bisquare: wgt = (1 -(vdist / bw)^2)^2 if vdist < bw, wgt = 0 otherwise; tricube: wgt = (1 -(vdist / bw)^3)^3 if vdist < bw, wgt = 0 otherwise; boxcar: wgt = 1 if dist < bw, wgt = 0 otherwise
adaptive	if TRUE, an adaptive kernel where the bandwidth (bw) corresponds to the proportion of the number of nearest neighbours (i.e. adaptive distance) is employed. The default is FALSE, where a fixed kernel is employed (bandwidth is a fixed distance).
bw	Bandwidth size. If adaptive kernel, bw should be the proportion of the number of nearest neighbours (0 < bw <=1). For fixed kernel, the Euclid distance.
dMat	A pre-specified distance matrix, it can be calculated by the function st_distance().
geodisic_measure	geodisic_measure is used when latlon coordinate. The distance is cauclated by geodist::geodist(). One of "haversine" "vincenty", "geodesic", or "cheap" specifying desired method of geodesic distance calculation. "Cheap" is the fastest way but may have errors if the ROI is large.
foreach	Whether parallel computation is implemented or not.

Value

SDF	A SpatialPointsDataFrame (may be gridded) or SpatialPolygonsDataFrame object (see package "sp") when the input is Spatial*DataFrame or a sf class object when input is sf, with local covariances, local correlations (Pearson's), local correlations (Spearman's), p-values of local correlations (Pearson's), p-values of local correlations (Spearman's), local partial correlations (Pearson's), local partial correlations (Spearman's), p-values of local partial correlations (Pearson's), and p-values of local partial correlations (Spearman's).
vars	Names of variables used for the calculation.
kernel	The name of kernel used for the calculation.
adaptive	Whether aadaptive kernel is employed or not (TRUE/FALSE),
bw	The bandwidth size used for the calculation.

Author(s)

Tsutsumida N. and Percival J.

References

Percival J. and Tsutsumida N. (2017) Geographically weighted partial correlation for spatial analysis, GI_forum, Issue 1, 36-43, URL http://dx.doi.org/10.1553/giscience2017_01_s36

Isabella Gollini, Binbin Lu, Martin Charlton, Christopher Brunsdon, Paul Harris (2015). GWmodel: An R Package for Exploring Spatial Heterogeneity Using Geographically Weighted Models. Journal of Statistical Software, 63(17), 1-50. URL http://www.jstatsoft.org/v63/i17/.

Binbin Lu, Paul Harris, Martin Charlton, Christopher Brunsdon (2014). The GWmodel R package: further topics for exploring spatial heterogeneity using geographically weighted models. Geospatial Information Science, 17(2), 85-101. URL http://dx.doi.org/10.1080/10095020.2014.917453

Examples

```
#NOTE: This example only shows how to implement gwpcor using sample data (meuse) in sp package.
#Results do not suggest any meanings.

#import data from sp package
library(sp)
library(sf)
data(meuse, package = "sp")
meuse_sf <- st_as_sf(meuse, coords = c("x", "y"), crs = 28992)

#implement gwpcor as an example
#the bandwidth is arbitrary.
res <- gwpcor(sdata = meuse_sf, vars = c("cadmium","copper", "zinc"),
method = "pearson", kernel = "bisquare",adaptive = TRUE,
bw = 0.25, geodisic_measure = "cheap", foreach = FALSE)
```

Index