

# Package: gdverse (via r-universe)

October 18, 2024

**Title** Analysis of Spatial Stratified Heterogeneity

**Version** 1.1-1

**Description** Analyzing spatial factors and exploring spatial associations based on the concept of spatial stratified heterogeneity, and also takes into account local spatial dependencies, spatial interpretability, potential spatial interactions, and robust spatial stratification. Additionally, it supports geographical detector models established in academic literature.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**URL** <https://stsc1.github.io/gdverse/>, <https://github.com/stsc1/gdverse>

**BugReports** <https://github.com/stsc1/gdverse/issues>

**Depends** R (>= 4.1.0)

**Imports** dplyr, forcats, geosphere, ggplot2, magrittr, parallel, patchwork, purrr, reticulate, rpart, scatterpie, sdsfun (>= 0.4.0), sf, stats, tibble, tidyr, utils

**Suggests** cowplot, knitr, Rcpp, readr, rmarkdown, terra, testthat (>= 3.0.0)

**LinkingTo** Rcpp

**VignetteBuilder** knitr

**LazyData** true

**Config/testthat/edition** 3

**NeedsCompilation** yes

**Author** Wenbo Lv [aut, cre, cph]

(<<https://orcid.org/0009-0002-6003-3800>>), Yangyang Lei [aut]

(<<https://orcid.org/0009-0002-6518-3613>>), Fangmei Liu [aut]

(<<https://orcid.org/0009-0008-8634-7588>>), Wufan Zhao [aut]

(<<https://orcid.org/0000-0002-0265-3465>>), Yongze Song [aut]

(<<https://orcid.org/0000-0003-3420-9622>>), Jianwu Yan [aut]

(<<https://orcid.org/0000-0001-9411-8453>>)

**Maintainer** Wenbo Lv <lyu.geosocial@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-10-17 06:00:03 UTC

## Contents

all2int . . . . .	4
cpsd_disc . . . . .	4
cpsd_spade . . . . .	6
ecological_detector . . . . .	7
factor_detector . . . . .	8
F_informationloss . . . . .	9
gd . . . . .	10
gd_bestunidisc . . . . .	11
generate_subsets . . . . .	12
geodetector . . . . .	13
gozh . . . . .	14
gozh_detector . . . . .	15
idsa . . . . .	16
interaction_detector . . . . .	18
inverse_distance_weight . . . . .	19
lesh . . . . .	20
loess_optscale . . . . .	21
ndvi . . . . .	22
NTDs . . . . .	22
opgd . . . . .	23
pid_idsa . . . . .	24
plot.ecological_detector . . . . .	25
plot.factor_detector . . . . .	26
plot.gd_result . . . . .	26
plot.gozh_result . . . . .	27
plot.idsa_result . . . . .	27
plot.interaction_detector . . . . .	28
plot.lesh_result . . . . .	29
plot.opgd_result . . . . .	30
plot.rgd_result . . . . .	30
plot.risk_detector . . . . .	31
plot.sesu_gozh . . . . .	32
plot.sesu_opgd . . . . .	32
plot.spade_result . . . . .	33
plot.srsgd_result . . . . .	34
plot.srs_ecological_detector . . . . .	34
plot.srs_factor_detector . . . . .	35
plot.srs_interaction_detector . . . . .	35
print.ecological_detector . . . . .	36
print.factor_detector . . . . .	37
print.gd_result . . . . .	37

print.gozh_result . . . . .	38
print.idsa_result . . . . .	38
print.interaction_detector . . . . .	39
print.lesh_result . . . . .	39
print.opgd_result . . . . .	40
print.rgd_result . . . . .	41
print.rid_result . . . . .	41
print.risk_detector . . . . .	42
print.sesu_gozh . . . . .	42
print.sesu_opgd . . . . .	43
print.spade_result . . . . .	43
print.srsgd_result . . . . .	44
print.srs_ecological_detector . . . . .	45
print.srs_factor_detector . . . . .	45
print.srs_interaction_detector . . . . .	46
psd_iev . . . . .	46
psd_pseudop . . . . .	47
psd_spade . . . . .	48
psmd_pseudop . . . . .	49
psmd_spade . . . . .	51
rgd . . . . .	52
rid . . . . .	53
risk_detector . . . . .	55
robust_disc . . . . .	55
rpart_disc . . . . .	56
sesu_gozh . . . . .	57
sesu_opgd . . . . .	59
shuffle_vector . . . . .	61
sim . . . . .	62
spade . . . . .	62
spd_lesh . . . . .	64
srsgd . . . . .	65
srs_ecological_detector . . . . .	66
srs_factor_detector . . . . .	67
srs_geodetector . . . . .	68
srs_interaction_detector . . . . .	69
srs_table . . . . .	70
srs_wt . . . . .	70
weight_assign . . . . .	70

---

all2int	<i>convert all discretized vectors to integer</i>
---------	---

---

**Description**

convert all discretized vectors to integer

**Usage**

```
all2int(x)
```

**Arguments**

x                    A discretized vector.

**Value**

An integer vector.

**Examples**

```
all2int(factor(letters[1:3],levels = c('b','a','c')))
all2int(letters[1:3])
```

---

cpsd_disc	<i>optimal spatial data discretization based on SPADE q-statistics</i>
-----------	--

---

**Description**

Function for determining the optimal spatial data discretization based on SPADE q-statistics.

**Usage**

```
cpsd_disc(
  formula,
  data,
  wt,
  discnum = 3:22,
  discmethod = "quantile",
  strategy = 2L,
  increase_rate = 0.05,
  cores = 1,
  return_disc = TRUE,
  seed = 123456789,
  ...
)
```

**Arguments**

formula	A formula of optimal spatial data discretization.
data	A data.frame or tibble of observation data.
wt	The spatial weight matrix.
discnum	(optional) A vector of number of classes for discretization. Default is 3:22.
discmethod	(optional) The discretization methods. Default all use quantile. Noted that robust will use robust_disc(); rpart will use rpart_disc(); Others use sdsfun::discretize_vector().
strategy	(optional) Discretization strategy. When strategy is 1L, choose the highest SPADE model q-statistics to determinate optimal spatial data discretization parameters. When strategy is 2L, The optimal discrete parameters of spatial data are selected by combining LOESS model.
increase_rate	(optional) The critical increase rate of the number of discretization. Default is 5%.
cores	(optional) A positive integer(default is 1). If cores > 1, a 'parallel' package cluster with that many cores is created and used. You can also supply a cluster object.
return_disc	(optional) Whether or not return discretized result used the optimal parameter. Default is TRUE.
seed	(optional) Random seed number, default is 123456789.Setting random seed is useful when the sample size is greater than 3000(the default value for largeN) and the data is discretized by sampling 10%(the default value for samp_prop in st_unidisc()).
...	(optional) Other arguments passed to st_unidisc(),robust_disc() or rpart_disc().

**Value**

A list with the optimal parameter in the provided parameter combination with k, method and disc(when return\_disc is TRUE).

x discretization variable name  
k optimal number of spatial data discreteization  
method optimal spatial data discretization method  
disc the result of optimal spatial data discretization

**Note**

When the discmethod is configured to robust, it will operate at a significantly reduced speed. Consequently, the use of robust discretization is not advised.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Yongze Song & Peng Wu (2021) An interactive detector for spatial associations, *International Journal of Geographical Information Science*, 35:8, 1676-1701, DOI:10.1080/13658816.2021.1882680

## Examples

```
data('sim')
wt = inverse_distance_weight(sim$lo, sim$la)
cpsd_disc(y ~ xa + xb + xc,
          data = sim,
          wt = wt)
```

---

cpsd\_spade

*compensated power of spatial determinant(CPSD)*

---

## Description

Function for calculate compensated power of spatial determinant  $Q_s$ .

## Usage

```
cpsd_spade(yobs, xobs, xdisc, wt)
```

## Arguments

yobs	Variable Y
xobs	The original undiscretized covariable X.
xdisc	The discretized covariable X.
wt	The spatial weight matrix.

## Details

The power of compensated spatial determinant formula is

$$Q_s = \frac{q_s}{q_{s_{in\ for\ kep}}} = \frac{1 - \frac{\sum_{h=1}^L N_h \Gamma_{kdep}}{N \Gamma_{totaldep}}}{1 - \frac{\sum_{h=1}^L N_h \Gamma_{hind}}{N \Gamma_{totalind}}}$$

## Value

A value of compensated power of spatial determinant  $Q_s$ .

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE), International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

## Examples

```
data('sim')
wt = inverse_distance_weight(sim$lo,sim$la)
xa = sim$xa
xa_disc = sdsfun::discretize_vector(xa,5)
cpsd_spade(sim$y,xa,xa_disc,wt)
```

---

ecological\_detector     *ecological\_detector*

---

## Description

Compare the effects of two factors  $X_1$  and  $X_2$  on the spatial distribution of the attribute  $Y$ .

## Usage

```
ecological_detector(y, x1, x2, alpha = 0.95)
```

## Arguments

y	Dependent variable, continuous numeric vector.
x1	Covariate $X_1$ , factor, character or discrete numeric.
x2	Covariate $X_2$ , factor, character or discrete numeric.
alpha	(optional) Confidence level of the interval, default is 0.95.

## Value

A list.

F-statistic the result of F statistic for ecological detector

P-value the result of P value for ecological detector

Ecological is there a significant difference between the two factors  $X_1$  and  $X_2$  on the spatial distribution of the attribute  $Y$

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
ecological_detector(y = 1:7,  
  x1 = c('x', rep('y', 3), rep('z', 3)),  
  x2 = c(rep('a', 2), rep('b', 2), rep('c', 3)))
```

---

factor\_detector      *factor\_detector*

---

**Description**

The factor detector q-statistic measures the spatial stratified heterogeneity of a variable Y, or the determinant power of a covariate X of Y.

**Usage**

```
factor_detector(y, x)
```

**Arguments**

y	Variable Y, continuous numeric vector.
x	Covariate X, factor, character or discrete numeric.

**Value**

A list.

Q-statistic the q statistic for factor detector

P-value the p value for factor detector

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
factor_detector(y = 1:7, x = c('x', rep('y', 3), rep('z', 3)))
```



---

F\_informationloss      *measure information loss by information entropy*

---

**Description**

Function for measure information loss by shannon information entropy.

**Usage**

```
F_informationloss(xvar, xdisc)
```

**Arguments**

xvar                    The original undiscretized vector.  
xdisc                   The discretized vector.

**Details**

The information loss measured by information entropy formula is  $F = - \sum_{i=1}^N p_{(i)} \log_2 p_{(i)} - \left( - \sum_{h=1}^L p_{(h)} \log_2 p_{(h)} \right)$

**Value**

A numeric value of information loss measured by information entropy.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
F_informationloss(1:7,c('x',rep('y',3),rep('z',3)))
```

---

gd *native geographical detector(GD) model*

---

### Description

Function for native geographical detector model.

### Usage

```
gd(formula, data, type = "factor", alpha = 0.95)
```

### Arguments

formula	A formula of geographical detector model.
data	A data.frame, tibble or sf object of observation data.
type	(optional) The type of geographical detector, which must be one of factor(default), interaction, risk, ecological. You can run one or more types at one time.
alpha	(optional) Specifies the size of the alpha (confidence level). Default is 0.95.

### Value

A list.

factor the result of factor detector  
interaction the result of interaction detector  
risk the result of risk detector  
ecological the result of ecological detector

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

Jin-Feng Wang, Xin-Hu Li, George Christakos, Yi-Lan Liao, Tin Zhang, XueGu & Xiao-Ying Zheng (2010) Geographical Detectors-Based Health Risk Assessment and its Application in the Neural Tube Defects Study of the Heshun Region, China, International Journal of Geographical Information Science, 24:1, 107-127, DOI: 10.1080/13658810802443457

### Examples

```
data("NTDs")  
g = gd(incidence ~ watershed + elevation + soiltype,  
      data = NTDs,type = c('factor','interaction'))  
g
```

---

gd_bestunidisc	<i>best univariate discretization based on geodetector q-statistic</i>
----------------	--

---

**Description**

Function for determining the best univariate discretization based on geodetector q-statistic.

**Usage**

```
gd_bestunidisc(
  formula,
  data,
  discnum = 3:22,
  discmethod = c("sd", "equal", "geometric", "quantile", "natural"),
  cores = 1,
  return_disc = TRUE,
  seed = 123456789,
  ...
)
```

**Arguments**

formula	A formula of best univariate discretization.
data	A data.frame or tibble of observation data.
discnum	(optional) A vector of number of classes for discretization. Default is 3:22.
discmethod	(optional) A vector of methods for discretization, default is using c("sd", "equal", "geometric", "quantile", "natural") by invoking sdsfun.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
return_disc	(optional) Whether or not return discretized result used the optimal parameter. Default is TRUE.
seed	(optional) Random seed number, default is 123456789.
...	(optional) Other arguments passed to sdsfun::discretize_vector().

**Value**

A list with the optimal parameter in the provided parameter combination with k, method and disc(when return\_disc is TRUE).

x the name of the variable that needs to be discretized  
k optimal discretization number  
method optimal discretization method  
disc optimal discretization results

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
data('sim')
gd_bestunidisc(y ~ xa + xb + xc,
              data = sim,
              discnum = 3:6)
```

---

generate_subsets	<i>generate subsets of a set</i>
------------------	----------------------------------

---

**Description**

generate subsets of a set

**Usage**

```
generate_subsets(set, empty = TRUE, self = TRUE)
```

**Arguments**

set	A vector.
empty	(optional) When empty is TRUE, the generated subset includes the empty set, otherwise the empty set is removed. Default is TRUE.
self	(optional) When self is TRUE, the resulting subset includes the set itself, otherwise the set itself is removed. Default is TRUE.

**Value**

A list.

**Examples**

```
generate_subsets(letters[1:3])
generate_subsets(letters[1:3],empty = FALSE)
generate_subsets(letters[1:3],self = FALSE)
generate_subsets(letters[1:3],empty = FALSE,self = FALSE)
```

---

geodetector                      *geographical detector*

---

## Description

geographical detector

## Usage

```
geodetector(formula, data, type = "factor", alpha = 0.95)
```

## Arguments

formula	A formula of geographical detector model.
data	A data.frame or tibble of observation data.
type	(optional) The type of geographical detector, which must be one of factor(default), interaction, risk, ecological.
alpha	(optional) Specifies the size of the alpha (confidence level). Default is 0.95.

## Value

A list of tibble with the corresponding result under different detector types.

factor the result of factor detector  
 interaction the result of interaction detector  
 risk the result of risk detector  
 ecological the result of ecological detector

## Note

Note that only one type of geodetector is supported at a time in geodetector().

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## Examples

```
geodetector(y ~ x1 + x2,
  tibble::tibble(y = 1:7,
    x1 = c('x', rep('y', 3), rep('z', 3)),
    x2 = c(rep('a', 2), rep('b', 2), rep('c', 3))))

geodetector(y ~ x1 + x2,
  tibble::tibble(y = 1:7,
    x1 = c('x', rep('y', 3), rep('z', 3)),
    x2 = c(rep('a', 2), rep('b', 2), rep('c', 3))))
```

```

type = 'interaction')

geodetector(y ~ x1 + x2,
  tibble::tibble(y = 1:7,
    x1 = c('x',rep('y',3),rep('z',3)),
    x2 = c(rep('a',2),rep('b',2),rep('c',3))),
  type = 'risk',alpha = 0.95)

geodetector(y ~ x1 + x2,
  tibble::tibble(y = 1:7,
    x1 = c('x',rep('y',3),rep('z',3)),
    x2 = c(rep('a',2),rep('b',2),rep('c',3))),
  type = 'ecological',alpha = 0.95)

```

---

gozh

*geographically optimal zones-based heterogeneity(GOZH) model*


---

## Description

Function for geographically optimal zones-based heterogeneity(GOZH) model

## Usage

```
gozh(formula, data, cores = 1, type = "factor", alpha = 0.95, ...)
```

## Arguments

formula	A formula of GOZH model.
data	A data.frame, tibble or sf object of observation data.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
type	(optional) The type of geographical detector, which must be factor (default), interaction, risk, ecological. You can run one or more types at one time.
alpha	(optional) Specifies the size of confidence level. Default is 0.95.
...	(optional) Other arguments passed to <code>rpart_disc()</code> .

## Value

A list.

factor the result of factor detector  
interaction the result of interaction detector  
risk the result of risk detector  
ecological the result of ecological detector

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Luo, P., Song, Y., Huang, X., Ma, H., Liu, J., Yao, Y., & Meng, L. (2022). Identifying determinants of spatio-temporal disparities in soil moisture of the Northern Hemisphere using a geographically optimal zones-based heterogeneity model. *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, 185, 111–128. <https://doi.org/10.1016/j.isprsjprs.2022.01.009>

**Examples**

```
data('ndvi')
g = gozh(NDVIchange ~ ., data = ndvi)
g
```

---

gozh\_detector

*geographically optimal zones-based heterogeneity detector*


---

**Description**

Function for geographically optimal zones-based heterogeneity detector.

**Usage**

```
gozh_detector(formula, data, cores = 1, type = "factor", alpha = 0.95, ...)
```

**Arguments**

formula	A formula of GOZH detector.
data	A data.frame or tibble of observation data.
cores	(optional) A positive integer(default is 1). If cores > 1, a 'parallel' package cluster with that many cores is created and used. You can also supply a cluster object.
type	(optional) The type of geographical detector, which must be one of factor(default), interaction, risk, ecological.
alpha	(optional) Confidence level of the interval, default is 0.95.
...	(optional) Other arguments passed to rpart_disc().

**Value**

A list of tibble with the corresponding result under different detector types.

factor the result of factor detector

interaction the result of interaction detector

risk the result of risk detector

ecological the result of ecological detector

**Note**

Only one type of detector is supported in a gozh\_detector() run at a time.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Luo, P., Song, Y., Huang, X., Ma, H., Liu, J., Yao, Y., & Meng, L. (2022). Identifying determinants of spatio-temporal disparities in soil moisture of the Northern Hemisphere using a geographically optimal zones-based heterogeneity model. *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, 185, 111–128. <https://doi.org/10.1016/j.isprsjprs.2022.01.009>

**Examples**

```
data('ndvi')
g = gozh_detector(NDVIchange ~ ., data = ndvi)
g
```

---

 idsa

---

*interactive detector for spatial associations(IDSA) model*


---

**Description**

Function for interactive detector for spatial associations model.

**Usage**

```
idsa(
  formula,
  data,
  wt = NULL,
  discnum = 3:22,
  discmethod = "quantile",
  overlay = "and",
```



```

strategy = 2L,
increase_rate = 0.05,
cores = 1,
seed = 123456789,
alpha = 0.95,
...
)

```

### Arguments

formula	A formula of IDSA model.
data	A data.frame, tibble or sf object of observation data.
wt	(optional) The spatial weight matrix. When data is not an sf object, must provide wt.
discnum	(optional) Number of multilevel discretization. Default will use 3:22.
discmethod	(optional) The discretization methods. Default all use quantile. Noted that robust will use robust_disc(); rpart will use rpart_disc(); Others use sdsfun::discretize_vector().
overlay	(optional) Spatial overlay method. One of and, or, intersection. Default is and.
strategy	(optional) Discretization strategy. When strategy is 1L, choose the highest SPADE model q-statistics to determinate optimal spatial data discretization parameters. When strategy is 2L, The optimal discrete parameters of spatial data are selected by combining LOESS model.
increase_rate	(optional) The critical increase rate of the number of discretization. Default is 5%.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
seed	(optional) Random number seed, default is 123456789.
alpha	(optional) Specifies the size of confidence level. Default is 0.95.
...	(optional) Other arguments passed to cpsd_disc().

### Value

A list.

interaction the interaction result of IDSA model

risk1 whether values of the response variable between a pair of overlay zones are significantly different

risk2 risk detection result of the input data

number\_individual\_explanatory\_variables the number of individual explanatory variables used for examining the interaction effects

number\_overlay\_zones the number of overlay zones

percentage\_finely\_divided\_zones the percentage of finely divided zones that are determined by the interaction of variables

**Note**

**Please note that all variables in the IDSA model need to be continuous data.**

The IDSA model requires at least  $2^n - 1$  calculations when has  $n$  explanatory variables. When there are more than 10 explanatory variables, carefully consider the computational burden of this model. When there are a large number of explanatory variables, the data dimensionality reduction method can be used to ensure the trade-off between analysis results and calculation speed.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Yongze Song & Peng Wu (2021) An interactive detector for spatial associations, International Journal of Geographical Information Science, 35:8, 1676-1701, DOI:10.1080/13658816.2021.1882680

**Examples**

```
data('sim')
sim1 = sf::st_as_sf(sim, coords = c('lo', 'la'))
g = idsa(y ~ ., data = sim1)
g
```

---

interaction\_detector    *interaction\_detector*

---

**Description**

Identify the interaction between different risk factors, that is, assess whether factors X1 and X2 together increase or decrease the explanatory power of the dependent variable Y, or whether the effects of these factors on Y are independent of each other.

**Usage**

```
interaction_detector(y, x1, x2)
```

**Arguments**

y	Dependent variable, continuous numeric vector.
x1	Covariate $X_1$ , factor, character or discrete numeric.
x2	Covariate $X_2$ , factor, character or discrete numeric.

**Value**

A list.

Variable1 Q-statistics Q-statistics for variable1

Variable2 Q-statistics Q-statistics for variable2

Variable1 and Variable2 interact Q-statistics Q-statistics for variable1 and variable2 interact

Interaction the interact result type

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
interaction_detector(y = 1:7,
                    x1 = c('x', rep('y', 3), rep('z', 3)),
                    x2 = c(rep('a', 2), rep('b', 2), rep('c', 3)))
```

---

inverse\_distance\_weight

*calculate inverse distance weight*

---

**Description**

Function for calculate inverse distance weight.

**Usage**

```
inverse_distance_weight(locx, locy, power = 1, is_arc = FALSE)
```

**Arguments**

locx	The x axis location.
locy	The y axis location.
power	(optional) Default is 1. Set to 2 for gravity weights.
is_arc	(optional) FALSE (default) or TRUE, whether to compute arc distance.

**Details**

The inverse distance weight formula is  $w_{ij} = 1/d_{ij}^\alpha$

**Value**

A inverse distance weight matrices with class of matrix.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
x = 1:10
y = 1:10
inverse_distance_weight(x,y)
inverse_distance_weight(x,y,is_arc = TRUE)
```

---

lesh

*locally explained heterogeneity(LESH) model*

---

**Description**

Function for locally explained heterogeneity model.

**Usage**

```
lesh(formula, data, cores = 1, ...)
```

**Arguments**

formula	A formula of LESH model.
data	A data.frame, tibble or sf object of observation data.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
...	(optional) Other arguments passed to rpart_disc().

**Value**

A list.

interaction the interaction result of LESH model

spd\_lesh a tibble of the SHAP power of determinants

**Note**

The LESH model requires at least  $2^n - 1$  calculations when has  $n$  explanatory variables. When there are more than 10 explanatory variables, carefully consider the computational burden of this model. When there are a large number of explanatory variables, the data dimensionality reduction method can be used to ensure the trade-off between analysis results and calculation speed.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Li, Y., Luo, P., Song, Y., Zhang, L., Qu, Y., & Hou, Z. (2023). A locally explained heterogeneity model for examining wetland disparity. *International Journal of Digital Earth*, 16(2), 4533–4552. <https://doi.org/10.1080/17538947.2023.2271883>

## Examples

```
data('ndvi')
g = llesh(NDVIchange ~ ., data = ndvi)
g
```

---

loess_optscale	<i>determine optimal spatial data analysis scale</i>
----------------	--

---

## Description

Function for determining optimal spatial data analysis scale based on locally estimated scatter plot smoothing (LOESS) model.

## Usage

```
loess_optscale(qvec, spscalevec, increase_rate = 0.05)
```

## Arguments

qvec	A numeric vector of q statistics.
spscalevec	A numeric vector of spatial scales corresponding to qvec.
increase_rate	(optional) The critical increase rate of the number of discretization. Default is 5%.

## Value

A numeric vector about optimal number of spatial scale and the critical increase rate of q value.

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## Examples

```
## Not run:
## The following code takes a long time to run:
library(tidyverse)
fvcpath = "https://github.com/SpatLyu/rdevdata/raw/main/FVC.tif"
fvc = terra::rast(paste0("/vsicurl/", fvcpath))
fvc1000 = fvc %>%
  terra::as.data.frame(na.rm = T) %>%
```

```

  as_tibble()
fvc5000 = fvc %>%
  terra::aggregate(fact = 5) %>%
  terra::as.data.frame(na.rm = T) %>%
  as_tibble()
qv1000 = factor_detector(fvc1000$fvc,
                        sdsfun::discretize_vector(fvc1000$premax,10,'quantile'))[[1]]
qv5000 = factor_detector(fvc5000$fvc,
                        sdsfun::discretize_vector(fvc5000$premax,10,'quantile'))[[1]]
loess_optscale(c(qv1000,qv5000),c(1000,5000))

## End(Not run)

```

---

ndvi	<i>dataset of NDVI changes and its influencing factors</i>
------	--

---

**Description**

dataset of NDVI changes and its influencing factors, modified from GD package.

**Usage**

```
ndvi
```

**Format**

ndvi: A tibble with 713 rows and 7 variables

**Author(s)**

Yongze Song <yongze.song@outlook.com>

---

NTDs	<i>NTDs data</i>
------	------------------

---

**Description**

The data were obtained by preprocessing use sf and tidyverse.

**Usage**

```
NTDs
```

**Format**

NTDs: A tibble with 185 rows and 4 variable columns and 2 location columns, modified from geodetector package.

opgd

*optimal parameters-based geographical detector(OPGD) model***Description**

Function for optimal parameters-based geographical detector(OPGD) model.

**Usage**

```
opgd(
  formula,
  data,
  discvar = NULL,
  discnum = 3:22,
  discmethod = c("sd", "equal", "geometric", "quantile", "natural"),
  cores = 1,
  type = "factor",
  alpha = 0.95,
  ...
)
```

**Arguments**

formula	A formula of OPGD model.
data	A data.frame, tibble or sf object of observation data.
discvar	Name of continuous variable columns that need to be discretized. Noted that when formula has discvar, data must have these columns. By default, all independent variables are used as discvar.
discnum	(optional) A vector of number of classes for discretization. Default is 3:22.
discmethod	(optional) A vector of methods for discretization, default is using c("sd", "equal", "geometric", "quantile", "natural"), by invoking sdsfun.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
type	(optional) The type of geographical detector, which must be factor(default), interaction, risk, ecological. You can run one or more types at one time.
alpha	(optional) Specifies the size of confidence level. Default is 0.95.
...	(optional) Other arguments passed to gd_bestunidisc(). A useful parameter is seed, which is used to set the random number seed.

**Value**

A list.

factor the result of factor detector

interaction the result of interaction detector

risk the result of risk detector  
 ecological the result of ecological detector

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

Song, Y., Wang, J., Ge, Y. & Xu, C. (2020) An optimal parameters-based geographical detector model enhances geographic characteristics of explanatory variables for spatial heterogeneity analysis: Cases with different types of spatial data, *GIScience & Remote Sensing*, 57(5), 593-610. doi: 10.1080/15481603.2020.1760434.

### Examples

```
data('sim')
opgd(y ~ xa + xb + xc, data = sim,
     discvar = paste0('x', letters[1:3]),
     discnum = 3:6)
```

---

pid\_idsa

*IDSa Q-saistics* PID

---

### Description

IDSa Q-saistics PID

### Usage

```
pid_idsa(formula, rawdata, discdata, wt, overlaymethod = "and")
```

### Arguments

formula	A formula for IDSa Q-saistics
rawdata	Raw observation data
discdata	Observed data with discrete explanatory variables
wt	Spatial weight matrix
overlaymethod	(optional) Spatial overlay method. One of and, or, intersection. Default is and.

### Details

$$Q_{IDSa} = \frac{\theta_r}{\phi}$$



**Value**

The value of IDSA Q-saistics PID.

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo, sim$la)
sim1 = dplyr::mutate(sim, dplyr::across(xa:xc, \(.x) sdsfun::discretize_vector(.x, 5)))
pid_idsa(y ~ xa + xb + xc, rawdata = sim,
         discdata = sim1, wt = wt)
```

---

```
plot.ecological_detector
      plot ecological detector
```

---

**Description**

S3 method to plot output for ecological detector in geodetector().

**Usage**

```
## S3 method for class 'ecological_detector'
plot(x, ...)
```

**Arguments**

x	Return by geodetector().
...	(optional) Other arguments passed to ggplot2::theme().

**Value**

A ggplot2 layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.factor\_detector *plot factor detector result*

---

### Description

S3 method to plot output for factor detector in geodetector().

### Usage

```
## S3 method for class 'factor_detector'
plot(x, slicenum = 2, alpha = 0.95, keep = TRUE, ...)
```

### Arguments

x	Return by geodetector().
slicenum	(optional) The number of labels facing inward. Default is 2.
alpha	(optional) Confidence level. Default is 0.95.
keep	(optional) Whether to keep Q-value results for insignificant variables, default is TRUE.
...	(optional) Other arguments passed to ggplot2::theme().

### Value

A ggplot2 layer.

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.gd\_result *plot GD result*

---

### Description

S3 method to plot output for GD model result in gd().

### Usage

```
## S3 method for class 'gd_result'
plot(x, ...)
```

### Arguments

x	Return by gd().
...	(optional) Other arguments passed to patchwork::wrap_plots().

**Value**

A ggplot2 layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

*plot.gozh\_result*      *plot GOZH result*

---

**Description**

S3 method to plot output for GOZH model result in `gozh()`.

**Usage**

```
## S3 method for class 'gozh_result'  
plot(x, ...)
```

**Arguments**

`x`                    Return by `gozh()`.  
`...`                 (optional) Other arguments passed to `patchwork::wrap_plots()`.

**Value**

A ggplot2 layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

*plot.idsa\_result*      *plot IDSA risk result*

---

**Description**

S3 method to plot output for IDSA risk result in `idsa()`.

**Usage**

```
## S3 method for class 'idsa_result'  
plot(x, ...)
```

**Arguments**

x                   Return by `idsa()`.  
...                   (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`plot.interaction_detector`  
*plot interaction detector result*

---

**Description**

S3 method to plot output for interaction detector in `geodetector()`.

**Usage**

```
## S3 method for class 'interaction_detector'  
plot(x, alpha = 1, ...)
```

**Arguments**

x                   Return by `geodetector()`.  
alpha               (optional) Picture transparency. Default is 1.  
...                 (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.lesh\_result      *plot LESH model result*

---

### Description

S3 method to plot output for LESH model interaction result in `lesh()`.

### Usage

```
## S3 method for class 'lesh_result'
plot(
  x,
  pie = TRUE,
  scatter = FALSE,
  scatter_alpha = 1,
  pieradius_factor = 15,
  pielegend_x = 0.99,
  pielegend_y = 0.1,
  pielegend_num = 3,
  ...
)
```

### Arguments

<code>x</code>	<code>x</code> Return by <code>lesh()</code> .
<code>pie</code>	(optional) Whether to draw the interaction contributions. Default is TRUE.
<code>scatter</code>	(optional) Whether to draw the interaction direction diagram. Default is FALSE.
<code>scatter_alpha</code>	(optional) Picture transparency. Default is 1.
<code>pieradius_factor</code>	(optional) The radius expansion factor of interaction contributions pie plot. Default is 15.
<code>pielegend_x</code>	(optional) The X-axis relative position of interaction contributions pie plot legend. Default is 0.99.
<code>pielegend_y</code>	(optional) The Y-axis relative position of interaction contributions pie plot legend. Default is 0.1.
<code>pielegend_num</code>	(optional) The number of interaction contributions pie plot legend. Default is 3.
<code>...</code>	(optional) Other arguments passed to <code>ggplot2::theme()</code> .

### Value

A `ggplot2` layer.

### Note

When both `scatter` and `pie` are set to TRUE in RStudio, enlarge the drawing frame for normal display.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.opgd\_result      *plot OPGD result*

---

**Description**

S3 method to plot output for OPGD model result in `opgd()`.

**Usage**

```
## S3 method for class 'opgd_result'  
plot(x, ...)
```

**Arguments**

`x`                      Return by `opgd()`.  
`...`                    (optional) Other arguments passed to `patchwork::wrap_plots()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.rgd\_result      *plot RGD result*

---

**Description**

S3 method to plot output for RGD model result in `rgd()`.

**Usage**

```
## S3 method for class 'rgd_result'  
plot(x, slicenum = 2, alpha = 0.95, keep = TRUE, ...)
```

**Arguments**

- x                   Return by `rgd()`.
- slicenum           (optional) The number of labels facing inward. Default is 2.
- alpha              (optional) Confidence level. Default is 0.95.
- keep               (optional) Whether to keep Q-value results for insignificant variables, default is TRUE.
- ...                 (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`plot.risk_detector`     *plot risk\_detector*

---

**Description**

S3 method to plot output for risk detector in `geodetector()`.

**Usage**

```
## S3 method for class 'risk_detector'  
plot(x, ...)
```

**Arguments**

- x                   Return by `geodetector()`.
- ...                 (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

plot.sesu\_gozh      *plot gozh sesu*

---

**Description**

S3 method to plot output for gozh sesu in `sesu_gozh()`.

**Usage**

```
## S3 method for class 'sesu_gozh'  
plot(x, ...)
```

**Arguments**

`x`                    Return by `sesu_gozh()`.  
`...`                 (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.sesu\_opgd      *plot opgd sesu*

---

**Description**

S3 method to plot output for opgd sesu in `sesu_opgd()`.

**Usage**

```
## S3 method for class 'sesu_opgd'  
plot(x, ...)
```

**Arguments**

`x`                    Return by `sesu_opgd()`.  
`...`                 (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer.



**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.spade\_result      *plot SPADE power of spatial and multilevel discretization determinant*

---

**Description**

S3 method to plot output for SPADE power of spatial and multilevel discretization determinant from spade().

**Usage**

```
## S3 method for class 'spade_result'  
plot(x, slicenum = 2, alpha = 0.95, keep = TRUE, ...)
```

**Arguments**

x	Return by spade().The number of labels facing inward.
slicenum	(optional) The number of labels facing inward. Default is 2.
alpha	(optional) Confidence level.Default is 0.95.
keep	(optional) Whether to keep Q-value results for insignificant variables, default is TRUE.
...	(optional) Other arguments passed to ggplot2::theme().

**Value**

A ggplot2 layer.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

plot.srsgd\_result      *plot SRS GD result*

---

**Description**

S3 method to plot output for SRS GD model result in `srsgd()`.

**Usage**

```
## S3 method for class 'srsgd_result'  
plot(x, ...)
```

**Arguments**

`x`                      Return by `srsgd()`.  
`...`                    (optional) Other arguments passed to `patchwork::wrap_plots()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

plot.srs\_ecological\_detector  
*plot spatial rough set-based ecological detector*

---

**Description**

S3 method to plot output for spatial rough set-based ecological detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_ecological_detector'  
plot(x, ...)
```

**Arguments**

`x`                      Return by `srsgd()`.  
`...`                    (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`plot.srs_factor_detector`  
*plot spatial rough set-based factor detector result*

---

**Description**

S3 method to plot output for spatial rough set-based factor detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_factor_detector'  
plot(x, slicenum = 2, ...)
```

**Arguments**

`x`                   Return by `srsgd()`.  
`slicenum`           (optional) The number of labels facing inward. Default is 2.  
`...`               (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`plot.srs_interaction_detector`  
*plot spatial rough set-based interaction detector result*

---

**Description**

S3 method to plot output for spatial rough set-based interaction detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_interaction_detector'  
plot(x, alpha = 1, ...)
```

**Arguments**

x                    Return by `srsgd()`.  
alpha                (optional) Picture transparency. Default is 1.  
...                   (optional) Other arguments passed to `ggplot2::theme()`.

**Value**

A `ggplot2` layer

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.ecological_detector`  
*print ecological\_detector*

---

**Description**

S3 method to format output for ecological detector in `geodetector()`.

**Usage**

```
## S3 method for class 'ecological_detector'  
print(x, ...)
```

**Arguments**

x                    Return by `geodetector()`.  
...                   (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

print.factor\_detector *print factor detector*

---

**Description**

S3 method to format output for factor detector in geodetector().

**Usage**

```
## S3 method for class 'factor_detector'  
print(x, ...)
```

**Arguments**

x                   Return by geodetector().  
...                  (optional) Other arguments passed to knitr::kable().

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

print.gd\_result       *print GD result*

---

**Description**

S3 method to format output for GD model from gd().

**Usage**

```
## S3 method for class 'gd_result'  
print(x, ...)
```

**Arguments**

x                   Return by gd().  
...                  (optional) Other arguments passed to knitr::kable().

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.gozh_result`      *print GOZH result*

---

**Description**

S3 method to format output for GOZH model from `gozh()`.

**Usage**

```
## S3 method for class 'gozh_result'  
print(x, ...)
```

**Arguments**

`x`                      Return by `gozh()`.  
`...`                    (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.idsa_result`      *print IDSA result*

---

**Description**

S3 method to format output for IDSA model from `idsa()`.

**Usage**

```
## S3 method for class 'idsa_result'  
print(x, ...)
```

**Arguments**

`x`                      Return by `idsa()`.  
`...`                    (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.interaction_detector`  
*print interaction detector*

---

**Description**

S3 method to format output for interaction detector in `geodetector()`.

**Usage**

```
## S3 method for class 'interaction_detector'  
print(x, ...)
```

**Arguments**

`x`                   Return by `geodetector()`.  
`...`               (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.lesh_result`       *print LESH model interaction result*

---

**Description**

S3 method to format output for LESH model interaction result in `lesh()`.

**Usage**

```
## S3 method for class 'lesh_result'  
print(x, ...)
```

**Arguments**

x                    Return by `lesh()`.  
...                   (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.opgd_result`      *print OPGD result*

---

**Description**

S3 method to format output for OPGD model from `opgd()`.

**Usage**

```
## S3 method for class 'opgd_result'  
print(x, ...)
```

**Arguments**

x                    Return by `opgd()`.  
...                   (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>



---

print.rgd\_result      *print RGD result*

---

**Description**

S3 method to format output for RGD model from `rgd()`.

**Usage**

```
## S3 method for class 'rgd_result'  
print(x, ...)
```

**Arguments**

`x`                    Return by `rgd()`.  
`...`                 (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

print.rid\_result      *print RID result*

---

**Description**

S3 method to format output for RID model from `rid()`.

**Usage**

```
## S3 method for class 'rid_result'  
print(x, ...)
```

**Arguments**

`x`                    Return by `rid()`.  
`...`                 (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

print.risk\_detector     *print risk detector*

---

**Description**

S3 method to format output for risk detector in geodetector().

**Usage**

```
## S3 method for class 'risk_detector'  
print(x, ...)
```

**Arguments**

x                    Return by geodetector().  
...                  (optional) Other arguments passed to knitr::kable().

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

print.sesu\_gozh         *print gozh sesu*

---

**Description**

S3 method to format output for gozh sesu from sesu\_gozh().

**Usage**

```
## S3 method for class 'sesu_gozh'  
print(x, ...)
```

**Arguments**

x                    Return by sesu\_gozh().  
...                  (optional) Other arguments passed to knitr::kable().

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.sesu_opgd`      *print opgd sesu*

---

**Description**

S3 method to format output for opgd sesu from `sesu_opgd()`.

**Usage**

```
## S3 method for class 'sesu_opgd'  
print(x, ...)
```

**Arguments**

`x`                      Return by `sesu_opgd()`.  
`...`                    (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.spade_result`      *print SPADE power of spatial and multilevel discretization determinant*

---

**Description**

S3 method to format output for SPADE power of spatial and multilevel discretization determinant from `spade()`.

**Usage**

```
## S3 method for class 'spade_result'  
print(x, ...)
```

**Arguments**

x                    Return by `spade()`.  
...                   Other arguments.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

`print.srsgd_result`     *print SRSGD result*

---

**Description**

S3 method to format output for SRSGD model from `srsgd()`.

**Usage**

```
## S3 method for class 'srsgd_result'  
print(x, ...)
```

**Arguments**

x                    Return by `srsgd()`.  
...                   (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

```
print.srs_ecological_detector
  print spatial rough set-based ecological detector
```

---

**Description**

S3 method to format output for spatial rough set-based ecological detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_ecological_detector'
print(x, ...)
```

**Arguments**

`x`                   Return by `srsgd()`.  
`...`               (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

```
print.srs_factor_detector
  print spatial rough set-based factor detector
```

---

**Description**

S3 method to format output for spatial rough set-based factor detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_factor_detector'
print(x, ...)
```

**Arguments**

`x`                   Return by `srsgd()`.  
`...`               (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

```
print.srs_interaction_detector
      print spatial rough set-based interaction detector
```

---

**Description**

S3 method to format output for spatial rough set-based interaction detector in `srsgd()`.

**Usage**

```
## S3 method for class 'srs_interaction_detector'
print(x, ...)
```

**Arguments**

`x`                    Return by `srsgd()`.  
`...`                 (optional) Other arguments passed to `knitr::kable()`.

**Value**

Formatted string output

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

---

```
psd_iev                    PSD of an interaction of explanatory variables (PSD-IEV)
```

---

**Description**

PSD of an interaction of explanatory variables (PSD-IEV)

**Usage**

```
psd_iev(disccdata, spzone, wt)
```

**Arguments**

disccdata	Observed data with discrete explanatory variables. A tibble or data.frame .
spzone	Fuzzy overlay spatial zones. Returned from st_fuzzyoverlay().
wt	Spatial weight matrix

**Details**

$$\phi = 1 - \frac{\sum_{i=1}^m \sum_{k=1}^{n_i} N_{i,k} \tau_{i,k}}{\sum_{i=1}^m N_i \tau_i}$$

**Value**

The Value of PSD-IEV

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Yongze Song & Peng Wu (2021) An interactive detector for spatial associations, International Journal of Geographical Information Science, 35:8, 1676-1701, DOI:10.1080/13658816.2021.1882680

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo, sim$la)
sim1 = dplyr::mutate(sim, dplyr::across(xa:xc, \(.x) sdsfun::discretize_vector(.x, 5)))
sz = sdsfun::fuzzyoverlay(y ~ xa + xb + xc, data = sim1)
psd_iev(dplyr::select(sim1, xa:xc), sz, wt)
```

---

psd_pseudop	<i>calculate power of spatial determinant(PSD) and the corresponding pseudo-p value</i>
-------------	---

---

**Description**

Function for calculate power of spatial determinant  $q_s$ .

**Usage**

```
psd_pseudop(y, x, wt, cores = 1, seed = 123456789, permutations = 0)
```

**Arguments**

y	Variable Y, continuous numeric vector.
x	Covariable X, factor, character or discrete numeric.
wt	The spatial weight matrix.
cores	(optional) A positive integer(default is 1). If cores > 1, use parallel computation.
seed	(optional) Random seed number, default is 123456789.
permutations	(optional) The number of permutations for the PSD computation. Default is 0, which means no pseudo-p values are calculated.

**Details**

The power of spatial determinant formula is  $q_s = 1 - \frac{\sum_{h=1}^L N_h \Gamma_h}{N\Gamma}$

**Value**

A tibble of power of spatial determinant and the corresponding pseudo-p value.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE), International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo, sim$la, power = 2)
psd_pseudop(sim$y, sdsfun::discretize_vector(sim$xa, 5), wt)
```

---

psd\_spade

*power of spatial determinant(PSD)*

---

**Description**

Function for calculate power of spatial determinant q\_s

**Usage**

```
psd_spade(y, x, wt)
```



**Arguments**

y	Variable Y, continuous numeric vector.
x	Covariable X, factor, character or discrete numeric.
wt	The spatial weight matrix.

**Details**

The power of spatial determinant formula is

$$q_s = 1 - \frac{\sum_{h=1}^L N_h \Gamma_h}{NT}$$

**Value**

A value of power of spatial determinant  $q_s$ .

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE), International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo, sim$la, power = 2)
psd_spade(sim$y, sdsfun::discretize_vector(sim$xa, 5), wt)
```

---

psmd_pseudop	<i>power of spatial and multilevel discretization determinant (PSMD) and the corresponding pseudo-p value</i>
--------------	---

---

**Description**

Function for calculate power of spatial and multilevel discretization determinant and the corresponding pseudo-p value.

**Usage**

```
psmd_pseudop(
  yobs,
  xobs,
  wt,
  discnum = 3:22,
  discmethod = "quantile",
  cores = 1,
  seed = 123456789,
  permutations = 0,
  ...
)
```

**Arguments**

yobs	Variable Y
xobs	The original undiscretized covariable X.
wt	The spatial weight matrix.
discnum	(optional) Number of multilevel discretization. Default will use 3:22.
discmethod	(optional) The discretization methods. Default will use quantile. If discmethod is set to robust, the function robust_disc() will be used. Conversely, if discmethod is set to rpart, the rpart_disc() function will be used. Others use sdsfun::discretize_vector(). Currently, only one discmethod can be used at a time.
cores	(optional) A positive integer(default is 1). If cores > 1, use parallel computation.
seed	(optional) Random seed number, default is 123456789.
permutations	(optional) The number of permutations for the PSD computation. Default is 0, which means no pseudo-p values are calculated.
...	(optional) Other arguments passed to sdsfun::discretize_vector(),robust_disc() or rpart_disc().

**Details**

The power of spatial and multilevel discretization determinant formula is  $PSMDQ_s = MEAN(Q_s)$

**Value**

A tibble of power of spatial and multilevel discretization determinant and the corresponding pseudo-p value.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE),International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo,sim$la)
psmd_pseudop(sim$y,sim$xa,wt)
```

---

psmd_spade	<i>power of spatial and multilevel discretization determinant(PSMD)</i>
------------	---

---

**Description**

Function for calculate power of spatial and multilevel discretization determinant PSMDQ<sub>s</sub>.

**Usage**

```
psmd_spade(
  yobs,
  xobs,
  wt,
  discnum = 3:22,
  discmethod = "quantile",
  cores = 1,
  seed = 123456789,
  ...
)
```

**Arguments**

yobs	Variable Y
xobs	The original undiscretized covariable X.
wt	The spatial weight matrix.
discnum	(optional) Number of multilevel discretization. Default will use 3:22.
discmethod	(optional) The discretization methods. Default will use quantile. If discmethod is set to robust, the function robust_disc() will be used. Conversely, if discmethod is set to rpart, the rpart_disc() function will be used. Others use sdsfun::discretize_vector(). Currently, only one discmethod can be used at a time.
cores	(optional) A positive integer(default is 1). If cores > 1, use parallel computation.
seed	(optional) Random seed number, default is 123456789.
...	(optional) Other arguments passed to sdsfun::discretize_vector(),robust_disc() or rpart_disc().

**Details**

The power of spatial and multilevel discretization determinant formula is  $PSMDQ_s = MEAN(Q_s)$

**Value**

A value of power of spatial and multilevel discretization determinant PSMDQ\_s.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE), International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

**Examples**

```
data('sim')
wt = inverse_distance_weight(sim$lo,sim$la)
psmd_spade(sim$y,sim$xa,wt)
```

---

 rgd

---

*robust geographical detector(RGD) model*


---

**Description**

Function for robust geographical detector(RGD) model.

**Usage**

```
rgd(formula, data, discvar = NULL, discnum = 3:22, minsize = 1, cores = 1)
```

**Arguments**

formula	A formula of RGD model.
data	A data.frame, tibble or sf object of observation data.
discvar	Name of continuous variable columns that need to be discretized. Noted that when formula has discvar, data must have these columns. By default, all independent variables are used as discvar.
discnum	A numeric vector of discretized classes of columns that need to be discretized. Default all discvar use 3:22.
minsize	(optional) The min size of each discretization group. Default all use 1.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.

**Value**

A list.

factor the result of RGD model

disc robust discrete results

**Note**

Please set up python dependence and configure GDVERSE\_PYTHON environment variable if you want to run `rgd()`. See `vignette('rgdrid', package = 'gdverse')` for more details.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Zhang, Z., Song, Y.\*, & Wu, P., 2022. Robust geographical detector. *International Journal of Applied Earth Observation and Geoinformation*. 109, 102782. DOI: 10.1016/j.jag.2022.102782.

**Examples**

```
## Not run:
## The following code needs to configure the Python environment to run:
data('ndvi')
g = rgd(NDVIchange ~ ., data = ndvi,
        discvar = names(ndvi)[-1:-3],
        discnum = 3:8, cores = 6)

## End(Not run)
```

---

<code>rid</code>	<i>robust interaction detector(RID) model</i>
------------------	---

---

**Description**

Function for robust interaction detector(RID) model.

**Usage**

```
rid(
  formula,
  data,
  discvar = NULL,
  discnum = 10,
  overlay = "intersection",
  minsize = 1,
  cores = 1
)
```

**Arguments**

formula	A formula of RID model.
data	A data.frame, tibble or sf object of observation data.
discvar	Name of continuous variable columns that need to be discretized. Noted that when formula has discvar, data must have these columns. By default, all independent variables are used as discvar.
discnum	A numeric vector for the number of discretized classes of columns that need to be discretized. Default all discvar use 10.
overlay	(optional) Spatial overlay method. One of and, or, intersection. Default is intersection.
minsize	(optional) The min size of each discretization group. Default all use 1.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.

**Value**

A list.  
 interaction the result of RID model

**Note**

The RID model requires at least  $2^n - 1$  calculations when has  $n$  explanatory variables. When there are more than 10 explanatory variables, carefully consider the computational burden of this model. When there are a large number of explanatory variables, the data dimensionality reduction method can be used to ensure the trade-off between analysis results and calculation speed.

Please set up python dependence and configure GDVERSE\_PYTHON environment variable if you want to run rid(). See vignette('rgdrid', package = 'gdverse') for more details.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Zhang, Z., Song, Y., Karunaratne, L., & Wu, P. (2024). Robust interaction detector: A case of road life expectancy analysis. *Spatial Statistics*, 59(100814), 100814. <https://doi.org/10.1016/j.spasta.2024.100814>

**Examples**

```
## Not run:
## The following code needs to configure the Python environment to run:
data('sim')
g = rid(y ~ .,
        data = dplyr::select(sim, -dplyr::any_of(c('lo', 'la'))),
        discnum = 4, cores = 6)
g

## End(Not run)
```

---

risk_detector	<i>risk detector</i>
---------------	----------------------

---

**Description**

Determine whether there is a significant difference between the attribute means of two sub regions.

**Usage**

```
risk_detector(y, x, alpha = 0.95)
```

**Arguments**

y	Variable Y, continuous numeric vector.
x	Covariate X, factor, character or discrete numeric.
alpha	(optional) Confidence level of the interval,default is 0.95.

**Value**

A tibble. contains different combinations of covariate X level and student t-test statistics, degrees of freedom, p-values, and whether has risk (Yes or No).

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
risk_detector(y = 1:7,  
             x = c('x',rep('y',3),rep('z',3)))
```

---

robust_disc	<i>univariate discretization based on offline change point detection</i>
-------------	--

---

**Description**

Determines discretization interval breaks using an optimization algorithm for variance-based change point detection.

**Usage**

```
robust_disc(formula, data, discnum, minsize = 1, cores = 1)
```

**Arguments**

formula	A formula of univariate discretization.
data	A data.frame or tibble of observation data.
discnum	A numeric vector of discretized classes of columns that need to be discretized.
minsize	(optional) The min size of each discretization group. Default all use 1.
cores	(optional) A positive integer(default is 1). If cores > 1, use python joblib package to parallel computation.

**Value**

A tibble of discretized columns which need to be discretized.

**Note**

Please set up python dependence and configure GDVERSE\_PYTHON environment variable if you want to run robust\_disc(). See vignette('rgdrid', package = 'gdverse') for more details.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**Examples**

```
## Not run:
## The following code needs to configure the Python environment to run:
data('ndvi')
robust_disc(NDVIchange ~ GDP,data = ndvi,discnum = 5)
robust_disc(NDVIchange ~ .,
            data = dplyr::select(ndvi,-c(Climatezone,Mining)),
            discnum = 10,cores = 6)

## End(Not run)
```

---

rpart\_disc

*discretization of variables based on recursive partitioning*

---

**Description**

discretization of variables based on recursive partitioning

**Usage**

```
rpart_disc(formula, data, ...)
```



**Arguments**

formula            A formula.  
 data                A data.frame or tibble of observation data.  
 ...                (optional) Other arguments passed to `rpart::rpart()`.

**Value**

A vector that being discretized.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Luo, P., Song, Y., Huang, X., Ma, H., Liu, J., Yao, Y., & Meng, L. (2022). Identifying determinants of spatio-temporal disparities in soil moisture of the Northern Hemisphere using a geographically optimal zones-based heterogeneity model. *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, 185, 111–128. <https://doi.org/10.1016/j.isprsjprs.2022.01.009>

**Examples**

```
data('ndvi')
rpart_disc(NDVIchange ~ ., data = ndvi)
```

---

sesu\_gozh

*comparison of size effects of spatial units based on GOZH*

---

**Description**

Function for comparison of size effects of spatial units in spatial heterogeneity analysis based on geographically optimal zones-based heterogeneity(GOZH) model.

**Usage**

```
sesu_gozh(
  formula,
  datalist,
  su,
  cores = 1,
  strategy = 2L,
  increase_rate = 0.05,
  alpha = 0.95,
  ...
)
```

### Arguments

formula	A formula of comparison of size effects of spatial units.
datalist	A list of data.frame or tibble.
su	A vector of sizes of spatial units.
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
strategy	(optional) Calculation strategies of Q statistics at different scales. Default is 2L, see details for more contents.
increase_rate	(optional) The critical increase rate of the number of discretization. Default is 5%.
alpha	(optional) Specifies the size of confidence level. Default is 0.95.
...	(optional) Other arguments passed to rpart_disc().

### Details

When strategy is 1, use the same process as sesu\_opgd(). If not, all explanatory variables are used to generate a unique Q statistic corresponding to the data in the datalist based on rpart\_disc() and gd(), and then loess\_optscale() is used to determine the optimal analysis scale.

### Value

A list.

- sesu a tibble representing size effects of spatial units
- optsu optimal spatial unit
- strategy the optimal analytical scale selection strategy
- increase\_rate the critical increase rate of q value

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

- Song, Y., Wang, J., Ge, Y. & Xu, C. (2020) An optimal parameters-based geographical detector model enhances geographic characteristics of explanatory variables for spatial heterogeneity analysis: Cases with different types of spatial data, *GIScience & Remote Sensing*, 57(5), 593-610. doi: 10.1080/15481603.2020.1760434.
- Luo, P., Song, Y., Huang, X., Ma, H., Liu, J., Yao, Y., & Meng, L. (2022). Identifying determinants of spatio-temporal disparities in soil moisture of the Northern Hemisphere using a geographically optimal zones-based heterogeneity model. *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, 185, 111–128. <https://doi.org/10.1016/j.isprsjprs.2022.01.009>

**Examples**

```
## Not run:
## The following code takes a long time to run:
library(tidyverse)
fvcpath = "https://github.com/SpatLyu/rdevdata/raw/main/FVC.tif"
fvc = terra::rast(paste0("/vsicurl/", fvcpath))
fvc1000 = fvc %>%
  terra::as.data.frame(na.rm = T) %>%
  as_tibble()
fvc5000 = fvc %>%
  terra::aggregate(factor = 5) %>%
  terra::as.data.frame(na.rm = T) %>%
  as_tibble()
sesu_gozh(fvc ~ .,
          datalist = list(fvc1000, fvc5000),
          su = c(1000, 5000),
          cores = 6)

## End(Not run)
```

sesu\_opgd

*comparison of size effects of spatial units based on OPGD***Description**

Function for comparison of size effects of spatial units in spatial heterogeneity analysis based on optimal parameters geographical detector(OPGD) model.

**Usage**

```
sesu_opgd(
  formula,
  datalist,
  su,
  discvar,
  discnum = 3:22,
  discmethod = c("sd", "equal", "geometric", "quantile", "natural"),
  cores = 1,
  increase_rate = 0.05,
  alpha = 0.95,
  ...
)
```

**Arguments**

formula	A formula of comparison of size effects of spatial units.
datalist	A list of data.frame or tibble.
su	A vector of sizes of spatial units.

discvar	Name of continuous variable columns that need to be discretized. Noted that when formula has discvar, data must have these columns.
discnum	(optional) A vector of number of classes for discretization. Default is 3:22.
discmethod	(optional) A vector of methods for discretization, default is using <code>c("sd", "equal", "geometric", "quantile")</code> by invoking <code>sdsfun</code> .
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
increase_rate	(optional) The critical increase rate of the number of discretization. Default is 5%.
alpha	(optional) Specifies the size of confidence level. Default is 0.95.
...	(optional) Other arguments passed to <code>gd_bestunidisc()</code> .

### Details

Firstly, the OPGD model is executed for each data in the datalist (all significant Q statistic of each data are averaged to represent the spatial association strength under this spatial unit), and then the `loess_optscale` function is used to select the optimal spatial analysis scale.

### Value

A list.

`sesu` a tibble representing size effects of spatial units

`optsu` optimal spatial unit

`increase_rate` the critical increase rate of q value

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

Song, Y., Wang, J., Ge, Y. & Xu, C. (2020) An optimal parameters-based geographical detector model enhances geographic characteristics of explanatory variables for spatial heterogeneity analysis: Cases with different types of spatial data, *GIScience & Remote Sensing*, 57(5), 593-610. doi: 10.1080/15481603.2020.1760434.

### Examples

```
## Not run:
## The following code takes a long time to run:
library(tidyverse)
fvcpath = "https://github.com/SpatLyu/rdevdata/raw/main/FVC.tif"
fvc = terra::rast(paste0("/vsicurl/", fvcpath))
fvc1000 = fvc %>%
  terra::as.data.frame(na.rm = T) %>%
  as_tibble()
fvc5000 = fvc %>%
```

```
terra::aggregate(fact = 5) %>%
terra::as.data.frame(na.rm = T) %>%
as_tibble()
sesu_opgd(fvc ~ .,
          datalist = list(fvc1000,fvc5000),
          su = c(1000,5000),
          discvar = names(select(fvc5000,-c(fvc,lulc))),
          cores = 6)

## End(Not run)
```

---

shuffle_vector	<i>randomly shuffling vector</i>
----------------	----------------------------------

---

## Description

randomly shuffling vector

## Usage

```
shuffle_vector(x, shuffle_rate, seed = 123456789)
```

## Arguments

x	A vector.
shuffle_rate	The shuffling rate.
seed	(optional) Random seed number. Default is 123456789.

## Value

A shuffled vector.

## Examples

```
shuffle_vector(1:100,0.15)
```

---

sim	<i>Simulation data.</i>
-----	-------------------------

---

**Description**

Simulation data.

**Usage**

```
sim
```

**Format**

sim: A tibble with 80 rows and 6 variables, modified from IDSA package.

**Author(s)**

Yongze Song <yongze.song@outlook.com>

---

spade	<i>spatial association detector (SPADE) model</i>
-------	---

---

**Description**

Function for spatial association detector (SPADE) model.

**Usage**

```
spade(  
  formula,  
  data,  
  wt = NULL,  
  discvar = NULL,  
  discnum = 3:22,  
  discmethod = "quantile",  
  cores = 1,  
  seed = 123456789,  
  permutations = 0,  
  ...  
)
```

**Arguments**

formula	A formula of spatial association detector (SPADE) model.
data	A data.frame, tibble or sf object of observation data.
wt	(optional) The spatial weight matrix. When data is not an sf object, must provide wt.
discvar	(optional) Name of continuous variable columns that need to be discretized. Noted that when formula has discvar, data must have these columns. By default, all independent variables are used as discvar.
discnum	(optional) Number of multilevel discretization. Default will use 3:22.
discmethod	(optional) The discretization methods. Default all use quantile. Note that when using different discmethod for discvar, please ensure that the lengths of both are consistent. Noted that robust will use robust_disc(); rpart will use rpart_disc(); Others use sdsfun::discretize_vector().
cores	(optional) Positive integer (default is 1). When cores are greater than 1, use multi-core parallel computing.
seed	(optional) Random number seed, default is 123456789.
permutations	(optional) The number of permutations for the PSD computation. Default is 0, which means no pseudo-p values are calculated.
...	(optional) Other arguments passed to sdsfun::discretize_vector(), robust_disc() or rpart_disc().

**Value**

A list.

factor the result of SPADE model

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

**References**

Xuezhi Cang & Wei Luo (2018) Spatial association detector (SPADE), International Journal of Geographical Information Science, 32:10, 2055-2075, DOI: 10.1080/13658816.2018.1476693

**Examples**

```
data('sim')
sim1 = sf::st_as_sf(sim, coords = c('lo', 'la'))
g = spade(y ~ ., data = sim1)
g
```

---

spd\_lesh                      *SHAP power of determinants (SPD)*

---

### Description

Function for calculate SHAP power of determinants *SPD*.

### Usage

```
spd_lesh(formula, data, cores = 1, ...)
```

### Arguments

formula	A formula of calculate SHAP power of determinants <i>SPD</i> .
data	A data.frame or tibble of observation data.
cores	(optional) A positive integer(default is 1). If cores > 1, a 'parallel' package cluster with that many cores is created and used. You can also supply a cluster object.
...	(optional) Other arguments passed to rpart_disc().

### Details

The power of SHAP power of determinants formula is

$$\theta_{x_j}(S) = \sum_{s \in M \setminus \{x_j\}} \frac{|S|!(|M|-|S|-1)!}{|M|!} (v(S \cup \{x_j\}) - v(S)).$$

SHAP power of determinants (SPD) is the contribution of variable  $x_j$  to the power of determinants.

### Value

A tibble with variable and its corresponding *SPD* value.

### Note

The SHAP power of determinants (SPD) requires at least  $2^n - 1$  calculations when has  $n$  explanatory variables. When there are more than 10 explanatory variables, carefully consider the computational burden of this model. When there are a large number of explanatory variables, the data dimensionality reduction method can be used to ensure the trade-off between analysis results and calculation speed.

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

Li, Y., Luo, P., Song, Y., Zhang, L., Qu, Y., & Hou, Z. (2023). A locally explained heterogeneity model for examining wetland disparity. *International Journal of Digital Earth*, 16(2), 4533–4552. <https://doi.org/10.1080/17538947.2023.2271883>



**Examples**

```
data('ndvi')
g = spd_lesh(NDVIchange ~ ., data = ndvi)
g
```

---

srsgd

*spatial rough set-based geographical detector(SRSGD) model*


---

**Description**

Function for spatial rough set-based geographical detector model.

**Usage**

```
srsgd(formula, data, wt = NULL, type = "factor", alpha = 0.95)
```

**Arguments**

formula	A formula of spatial rough set-based geographical detector model.
data	A data.frame, tibble or sf object of observation data.
wt	Spatial adjacency matrix. If data is a sf polygon object, the queen adjacency matrix is used when no wt object is provided. In other cases, you must provide a wt object.
type	(optional) The type of geographical detector, which must be one of factor(default), interaction and ecological.
alpha	(optional) Specifies the size of the alpha (confidence level). Default is 0.95.

**Value**

A list.

factor the result of spatial rough set-based factor detector

interaction the result of spatial rough set-based interaction detector

ecological the result of spatial rough set-based ecological detector

**Note**

The Spatial Rough Set-based Geographical Detector Model (SRSGD) conducts spatial hierarchical heterogeneity analysis utilizing a geographical detector for data where *the dependent variable is discrete*. Given the complementary relationship between SRSGD and the native version of geographical detector, I strive to maintain consistency with gd() function when establishing srsgd() function. This implies that all input variable data in srsgd must *be discretized prior to use*.

**Author(s)**

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Bai, H., Li, D., Ge, Y., Wang, J., & Cao, F. (2022). Spatial rough set-based geographical detectors for nominal target variables. *Information Sciences*, 586, 525–539. <https://doi.org/10.1016/j.ins.2021.12.019>

## Examples

```
data('srs_table')
data('srs_wt')
srsigd(d ~ a1 + a2 + a3, data = srs_table, wt = srs_wt,
       type = c('factor', 'interaction', 'ecological'))
```

---

srs\_ecological\_detector

*spatial rough set-based ecological detector*

---

## Description

spatial rough set-based ecological detector

## Usage

```
srs_ecological_detector(y, x1, x2, wt, alpha = 0.95)
```

## Arguments

y	Dependent variable, factor, character or discrete numeric.
x1	Covariate $X_1$ , factor, character or discrete numeric.
x2	Covariate $X_2$ , factor, character or discrete numeric.
wt	Spatial adjacency matrix.
alpha	(optional) Confidence level of the interval, default is 0.95.

## Value

A list.

T-statistic the result of T statistic for spatial rough set-based ecological detector

P-value the result of P value for spatial rough set-based ecological detector

Ecological does one spatial feature  $X_1$  play a more important role than  $X_2$

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Bai, H., Li, D., Ge, Y., Wang, J., & Cao, F. (2022). Spatial rough set-based geographical detectors for nominal target variables. *Information Sciences*, 586, 525–539. <https://doi.org/10.1016/j.ins.2021.12.019>

## Examples

```
data('srs_table')
data('srs_wt')
srs_ecological_detector(srs_table$d, srs_table$a1, srs_table$a2, srs_wt)
```

---

srs\_factor\_detector     *spatial rough set-based factor detector*

---

## Description

spatial rough set-based factor detector

## Usage

```
srs_factor_detector(y, x, wt)
```

## Arguments

y	Variable Y, factor, character or discrete numeric.
x	Covariate X, factor, character or discrete numeric.
wt	Spatial adjacency matrix.

## Value

A list.

PD the average local explanatory power

SE\_PD the degree of spatial heterogeneity of the local explanatory power

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## References

Bai, H., Li, D., Ge, Y., Wang, J., & Cao, F. (2022). Spatial rough set-based geographical detectors for nominal target variables. *Information Sciences*, 586, 525–539. <https://doi.org/10.1016/j.ins.2021.12.019>

## Examples

```
data('srs_table')
data('srs_wt')
srs_factor_detector(srs_table$d, srs_table$a1, srs_wt)
```

---

srs_geodetector	<i>spatial rough set-based geographical detector</i>
-----------------	--

---

## Description

spatial rough set-based geographical detector

## Usage

```
srs_geodetector(formula, data, wt = NULL, type = "factor", alpha = 0.95)
```

## Arguments

formula	A formula of spatial rough set-based geographical detector model.
data	A data.frame, tibble or sf object of observation data.
wt	Spatial adjacency matrix. If data is a sf polygon object, the queen adjacency matrix is used when no wt object is provided. In other cases, you must provide a wt object.
type	(optional) The type of geographical detector, which must be one of factor(default), interaction and ecological.
alpha	(optional) Specifies the size of the alpha (confidence level). Default is 0.95.

## Value

A list of tibble with the corresponding result under different detector types.

factor the result of spatial rough set-based factor detector

interaction the result of spatial rough set-based interaction detector

ecological the result of spatial rough set-based ecological detector

## Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

## Examples

```
data('srs_table')
data('srs_wt')
srs_geodetector(d ~ a1 + a2 + a3, data = srs_table, wt = srs_wt)
srs_geodetector(d ~ a1 + a2 + a3, data = srs_table,
                wt = srs_wt, type = 'interaction')
srs_geodetector(d ~ a1 + a2 + a3, data = srs_table,
                wt = srs_wt, type = 'ecological')
```

---

srs\_interaction\_detector

*spatial rough set-based interaction detector*

---

### Description

spatial rough set-based interaction detector

### Usage

```
srs_interaction_detector(y, x1, x2, wt)
```

### Arguments

y	Dependent variable, factor, character or discrete numeric.
x1	Covariate $X_1$ , factor, character or discrete numeric.
x2	Covariate $X_2$ , factor, character or discrete numeric.
wt	Spatial adjacency matrix.

### Value

A list.

Variable1 PD the average local explanatory power for variable1

Variable2 PD the average local explanatory power for variable2

Variable1 and Variable2 interact PD the average local explanatory power for variable1 and variable2 interact

Variable1 SE\_PD the degree of spatial heterogeneity of the local explanatory power for variable1

Variable2 SE\_PD the degree of spatial heterogeneity of the local explanatory power for variable2

Variable1 and Variable2 SE\_PD the degree of spatial heterogeneity of the local explanatory power for variable1 and variable2 interact

Interaction the interact result type

### Author(s)

Wenbo Lv <lyu.geosocial@gmail.com>

### References

Bai, H., Li, D., Ge, Y., Wang, J., & Cao, F. (2022). Spatial rough set-based geographical detectors for nominal target variables. *Information Sciences*, 586, 525–539. <https://doi.org/10.1016/j.ins.2021.12.019>

### Examples

```
data('srs_table')
data('srs_wt')
srs_interaction_detector(srs_table$d, srs_table$a1, srs_table$a2, srs_wt)
```

---

srs_table	<i>example of spatial information system table</i>
-----------	--

---

**Description**

example of spatial information system table

**Usage**

srs\_table

**Format**

srs\_table: A tibble with 11 rows and 5 variables(one ID column).

---

srs_wt	<i>example of spatial information system spatial adjacency matrix</i>
--------	---

---

**Description**

example of spatial information system spatial adjacency matrix

**Usage**

srs\_wt

**Format**

srs\_wt: A matrix with 11rows and 11cols.

---

weight_assign	<i>assign values by weight</i>
---------------	--------------------------------

---

**Description**

assign values by weight

**Usage**

weight\_assign(x, w, list = FALSE)

**Arguments**

- `x`                    A numeric value
- `w`                    A weight vector
- `list`                (optional) Return list or not. if `list` is TRUE, return a list, otherwise return a vector. Default is FALSE.

**Value**

A numeric Vector.

**Examples**

```
weight_assign(0.875,1:3)
```

# Index

- \* **NTDs**
  - NTDs, 22
- \* **dataset**
  - ndvi, 22
  - NTDs, 22
  - sim, 62
  - srs\_table, 70
  - srs\_wt, 70
- \* **ndvi**
  - ndvi, 22
- \* **sim**
  - sim, 62
- \* **srs\_table**
  - srs\_table, 70
- \* **srs\_wt**
  - srs\_wt, 70
- all2int, 4
- cpsd\_disc, 4
- cpsd\_spade, 6
- ecological\_detector, 7
- F\_informationloss, 9
- factor\_detector, 8
- gd, 10
- gd\_bestunidisc, 11
- generate\_subsets, 12
- geodetector, 13
- gozh, 14
- gozh\_detector, 15
- idsa, 16
- interaction\_detector, 18
- inverse\_distance\_weight, 19
- lesh, 20
- loess\_optscale, 21
- ndvi, 22
- NTDs, 22
- opgd, 23
- pid\_idsa, 24
- plot.ecological\_detector, 25
- plot.factor\_detector, 26
- plot.gd\_result, 26
- plot.gozh\_result, 27
- plot.idsa\_result, 27
- plot.interaction\_detector, 28
- plot.lesh\_result, 29
- plot.opgd\_result, 30
- plot.rgd\_result, 30
- plot.risk\_detector, 31
- plot.sesu\_gozh, 32
- plot.sesu\_opgd, 32
- plot.spade\_result, 33
- plot.srs\_ecological\_detector, 34
- plot.srs\_factor\_detector, 35
- plot.srs\_interaction\_detector, 35
- plot.srsgd\_result, 34
- print.ecological\_detector, 36
- print.factor\_detector, 37
- print.gd\_result, 37
- print.gozh\_result, 38
- print.idsa\_result, 38
- print.interaction\_detector, 39
- print.lesh\_result, 39
- print.opgd\_result, 40
- print.rgd\_result, 41
- print.rid\_result, 41
- print.risk\_detector, 42
- print.sesu\_gozh, 42
- print.sesu\_opgd, 43
- print.spade\_result, 43
- print.srs\_ecological\_detector, 45
- print.srs\_factor\_detector, 45
- print.srs\_interaction\_detector, 46



print.srsgd\_result, 44  
psd\_iev, 46  
psd\_pseudop, 47  
psd\_spade, 48  
psmd\_pseudop, 49  
psmd\_spade, 51  
  
rgd, 52  
rid, 53  
risk\_detector, 55  
robust\_disc, 55  
rpart\_disc, 56  
  
sesu\_gozh, 57  
sesu\_opgd, 59  
shuffle\_vector, 61  
sim, 62  
spade, 62  
spd\_lesh, 64  
srs\_ecological\_detector, 66  
srs\_factor\_detector, 67  
srs\_geodetector, 68  
srs\_interaction\_detector, 69  
srs\_table, 70  
srs\_wt, 70  
srsgd, 65  
  
weight\_assign, 70