Package 'waywiser'

August 10, 2022

Type Package

Title Yardstick Extensions for Measuring Spatial Structure in Model Residuals

Version 0.1.0

Description Predictive models of spatial data may have spatially structured errors, with ``hot spots'' of higher than expected error clustered geographically due to spatial structure in the underlying data. These functions provide methods for measuring the spatial structure of model errors for models, and are particularly useful for models fit using the 'tidymodels' framework. Methods include Moran's I ('Moran' (1950) <doi:10.2307/2332142>), Geary's C ('Geary' (1954) <doi:10.2307/2986645>) and Getis-Ord's G ('Ord' and 'Getis' (1995) <doi:10.1111/j.1538-4632.1995.tb00912.x>).

License MIT + file LICENSE

URL https://github.com/mikemahoney218/waywiser, https://mikemahoney218.github.io/waywiser/

BugReports https://github.com/mikemahoney218/waywiser/issues

Depends R (>= 3.5)

Imports rlang, sf, spdep, yardstick

Suggests covr, dplyr, ggplot2, sfdep, spelling, testthat (>= 3.0.0), tidymodels, tidyr

Config/testthat/edition 3

Config/testthat/parallel true

Encoding UTF-8

RoxygenNote 7.2.1

Language en-US

NeedsCompilation no

Author Michael Mahoney [aut, cre] (<https://orcid.org/0000-0003-2402-304X>), RStudio [cph, fnd]

Maintainer Michael Mahoney <mike.mahoney.218@gmail.com>

Repository CRAN

Date/Publication 2022-08-10 14:20:02 UTC

R topics documented:

`ww_build_neighbors` *Make 'nb' objects from sf objects*

Description

Make 'nb' objects from sf objects

Usage

```
ww_build_neighbors(data, nb = NULL, ..., call = rlang::caller_env())
```

Arguments

`data`	An sf object (of class "sf" or "sfc").
`nb`	An object of class "nb" (in which case it will be returned unchanged), or a function to create an object of class "nb" from `data` and `...`, or `NULL`. See details.
`...`	Arguments passed to the neighbor-creating function.
`call`	The execution environment of a currently running function, e.g. `call = caller_env()`. The corresponding function call is retrieved and mentioned in error messages as the source of the error. You only need to supply `call` when throwing a condition from a helper function which wouldn't be relevant to mention in the message. Can also be `NULL` or a defused function call to respectively not display any call or hard-code a code to display. For more information about error calls, see Including function calls in error messages.

Details

When nb = NULL, the method used to create neighbors from data is dependent on what geometry type data is:

- If nb = NULL and data is a point geometry (classes "sfc_POINT" or "sfc_MULTIPOINT") the "nb" object will be created using ww_make_point_neighbors().
- If nb = NULL and data is a polygon geometry (classes "sfc_POLYGON" or "sfc_MULTIPOLYGON") the "nb" object will be created using ww_make_polygon_neighbors().
- If nb = NULL and data is any other geometry type, the "nb" object will be created using the centroids of the data as points, with a warning.

Value

An object of class "nb".

---

ww_build_weights *Build "listw" objects of spatial weights*

Description

Build "listw" objects of spatial weights

Usage

```
ww_build_weights(x, wt = NULL, include_self = FALSE, ...)
```

Arguments

x	Either an sf object or a "nb" neighbors list object. If an sf object, will be converted into a neighbors list via ww_build_neighbors().
wt	Either a "listw" object (which will be returned unchanged), a function for creating a "listw" object from x, or NULL, in which case weights will be constructed via spdep::nb2listw().
include_self	Include each region itself in its own list of neighbors?
...	Arguments passed to the weight constructing function.

Value

A listw object.

ww_global_geary_c *Global Geary's C statistic*

Description

Calculate the global Geary's C statistic for model residuals. `ww_global_geary_c()` returns the statistic itself, while `ww_global_geary_pvalue()` returns the associated p value. `ww_global_geary()` returns both.

Usage

```
ww_global_geary_c(data, ...)

ww_global_geary_c_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)

ww_global_geary_pvalue(data, ...)

ww_global_geary_pvalue_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)

ww_global_geary(
  data,
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)
```

Arguments

data	A data.frame containing the columns specified by the truth and estimate arguments.
...	Additional arguments passed to spdep::geary.test().
truth	The column identifier for the true results (that is numeric). This should be an unquoted column name although this argument is passed by expression and supports quasiquotation (you can unquote column names). For _vec() functions, a numeric vector.
estimate	The column identifier for the predicted results (that is also numeric). As with truth this can be specified different ways but the primary method is to use an unquoted variable name. For _vec() functions, a numeric vector.
wt	A "listw" object, for instance as created with ww_build_weights().
alternative	a character string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".
randomization	variance of I calculated under the assumption of randomisation, if FALSE normality
na_rm	A logical value indicating whether NA values should be stripped before the computation proceeds.

Value

A tibble with columns .metric, .estimator, and .estimate and nrow(data) rows of values. For grouped data frames, the number of rows returned will be the same as the number of groups. For _vec() functions, a single value (or NA).

Examples

```
data(guerry, package = "sfdep")

guerry_modeled <- guerry
guerry_lm <- lm(crime_pers ~ literacy, guerry_modeled)
guerry_modeled$predictions <- predict(guerry_lm, guerry_modeled)

## Not run:
ww_global_geary(guerry_modeled, crime_pers, predictions)

## End(Not run)
```

---

ww_global_moran_i *Global Moran's I statistic*

Description

Calculate the global Moran's I statistic for model residuals. `ww_global_moran_i()` returns the statistic itself, while `ww_global_moran_pvalue()` returns the associated p value. `ww_global_moran()` returns both.

Usage

```
ww_global_moran_i(data, ...)

ww_global_moran_i_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)

ww_global_moran_pvalue(data, ...)

ww_global_moran_pvalue_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)

ww_global_moran(
  data,
  truth,
  estimate,
  wt = NULL,
  alternative = "greater",
  randomization = TRUE,
  na_rm = TRUE,
  ...
)
```

Arguments

`data`	A `data.frame` containing the columns specified by the `truth` and `estimate` arguments.
`...`	Additional arguments passed to `spdep::moran.test()`.

truth	The column identifier for the true results (that is numeric). This should be an unquoted column name although this argument is passed by expression and supports quasiquotation (you can unquote column names). For _vec() functions, a numeric vector.
estimate	The column identifier for the predicted results (that is also numeric). As with truth this can be specified different ways but the primary method is to use an unquoted variable name. For _vec() functions, a numeric vector.
wt	A "listw" object, for instance as created with ww_build_weights().
alternative	a character string specifying the alternative hypothesis, must be one of greater (default), less or two.sided.
randomization	variance of I calculated under the assumption of randomisation, if FALSE normality
na_rm	A logical value indicating whether NA values should be stripped before the computation proceeds.

Value

A tibble with columns .metric, .estimator, and .estimate and nrow(data) rows of values. For grouped data frames, the number of rows returned will be the same as the number of groups. For _vec() functions, a single value (or NA).

Examples

```
data(guerry, package = "sfdep")

guerry_modeled <- guerry
guerry_lm <- lm(crime_pers ~ literacy, guerry_modeled)
guerry_modeled$predictions <- predict(guerry_lm, guerry_modeled)

ww_global_moran_i(guerry_modeled, crime_pers, predictions)
ww_global_moran(guerry_modeled, crime_pers, predictions)
```

---

ww_local_geary_c *Local Geary's C statistic*

Description

Calculate the local Geary's C statistic for model residuals. ww_local_geary_c() returns the statistic itself, while ww_local_geary_pvalue() returns the associated p value. ww_local_geary() returns both.

Usage

```
ww_local_geary_c(data, ...)

ww_local_geary_c_vec(truth, estimate, wt, na_rm = TRUE, ...)

ww_local_geary_pvalue(data, ...)

ww_local_geary_pvalue_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  nsim = 499,
  na_rm = TRUE,
  ...
)

ww_local_geary(
  data,
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  nsim = 499,
  na_rm = TRUE,
  ...
)
```

Arguments

data	A data.frame containing the columns specified by the truth and estimate arguments.
...	Additional arguments passed to spdep::localC_perm().
truth	The column identifier for the true results (that is numeric). This should be an unquoted column name although this argument is passed by expression and supports quasiquotation (you can unquote column names). For _vec() functions, a numeric vector.
estimate	The column identifier for the predicted results (that is also numeric). As with truth this can be specified different ways but the primary method is to use an unquoted variable name. For _vec() functions, a numeric vector.
wt	A "listw" object, for instance as created with ww_build_weights().
na_rm	A logical value indicating whether NA values should be stripped before the computation proceeds.
alternative	A character defining the alternative hypothesis. Must be one of "two.sided", "less" or "greater".
nsim	The number of simulations to be used for permutation test.

Value

A tibble with columns .metric, .estimator, and .estimate and `nrow(data)` rows of values. For grouped data frames, the number of rows returned will be the same as the number of groups. For `_vec()` functions, a numeric vector of `length(truth)` (or NA).

Examples

```
data(guerry, package = "sfdep")

guerry_modeled <- guerry
guerry_lm <- lm(crime_pers ~ literacy, guerry_modeled)
guerry_modeled$predictions <- predict(guerry_lm, guerry_modeled)

ww_local_geary_c(guerry_modeled, crime_pers, predictions)
ww_local_geary(guerry_modeled, crime_pers, predictions)
```

---

ww_local_getis_ord_g *Local Getis-Ord G and G* statistic*

Description

Calculate the local Getis-Ord G and G* statistic for model residuals. `ww_local_getis_ord_g()` returns the statistic itself, while `ww_local_getis_ord_pvalue()` returns the associated p value. `ww_local_getis_ord()` returns both.

Usage

```
ww_local_getis_ord_g(data, ...)

ww_local_getis_ord_g_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  nsim = 499,
  na_rm = TRUE,
  ...,
  include_self = FALSE
)

ww_local_getis_ord_pvalue(data, ...)

ww_local_getis_ord_pvalue_vec(
  truth,
  estimate,
```

```
  wt = NULL,
  alternative = "two.sided",
  nsim = 499,
  na_rm = TRUE,
  ...,
  include_self = FALSE
)

ww_local_getis_ord(
  data,
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  nsim = 499,
  na_rm = TRUE,
  ...,
  include_self = FALSE
)
```

Arguments

data	A data.frame containing the columns specified by the truth and estimate arguments.
...	Arguments passed to spdep::localG_perm()
truth	The column identifier for the true results (that is numeric). This should be an unquoted column name although this argument is passed by expression and supports quasiquotation (you can unquote column names). For _vec() functions, a numeric vector.
estimate	The column identifier for the predicted results (that is also numeric). As with truth this can be specified different ways but the primary method is to use an unquoted variable name. For _vec() functions, a numeric vector.
wt	A "listw" object, for instance as created with ww_build_weights().
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less".
nsim	default 499, number of conditonal permutation simulations
na_rm	A logical value indicating whether NA values should be stripped before the computation proceeds.
include_self	Include each region itself in its own list of neighbors? Only used when wt is NULL, and if TRUE means this function calculates G* instead of G.

Value

A tibble with columns .metric, .estimator, and .estimate and nrow(data) rows of values. For grouped data frames, the number of rows returned will be the same as the number of groups. For _vec() functions, a numeric vector of length(truth) (or NA).

Examples

```
data(guerry, package = "sfdep")

guerry_modeled <- guerry
guerry_lm <- lm(crime_pers ~ literacy, guerry_modeled)
guerry_modeled$predictions <- predict(guerry_lm, guerry_modeled)

ww_local_getis_ord_g(guerry_modeled, crime_pers, predictions)
ww_local_getis_ord(guerry_modeled, crime_pers, predictions)
ww_local_getis_ord(guerry_modeled, crime_pers, predictions, include_self = TRUE)
```

---

ww_local_moran_i *Local Moran's I statistic*

Description

Calculate the local Moran's I statistic for model residuals. `ww_local_moran_i()` returns the statistic itself, while `ww_local_moran_pvalue()` returns the associated p value. `ww_local_moran()` returns both.

Usage

```
ww_local_moran_i(data, ...)

ww_local_moran_i_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  na_rm = TRUE,
  ...
)

ww_local_moran_pvalue(data, ...)

ww_local_moran_pvalue_vec(
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  na_rm = TRUE,
  ...
)

ww_local_moran(
```

```
  data,
  truth,
  estimate,
  wt = NULL,
  alternative = "two.sided",
  na_rm = TRUE,
  ...
)
```

Arguments

data	A data.frame containing the columns specified by the truth and estimate arguments.
...	Additional arguments passed to spdep::localmoran().
truth	The column identifier for the true results (that is numeric). This should be an unquoted column name although this argument is passed by expression and supports quasiquotation (you can unquote column names). For _vec() functions, a numeric vector.
estimate	The column identifier for the predicted results (that is also numeric). As with truth this can be specified different ways but the primary method is to use an unquoted variable name. For _vec() functions, a numeric vector.
wt	A "listw" object, for instance as created with ww_build_weights().
alternative	a character string specifying the alternative hypothesis, must be one of greater, less or two.sided (default).
na_rm	A logical value indicating whether NA values should be stripped before the computation proceeds.

Value

A tibble with columns .metric, .estimator, and .estimate and nrow(data) rows of values. For grouped data frames, the number of rows returned will be the same as the number of groups. For _vec() functions, a numeric vector of length(truth) (or NA).

Examples

```
data(guerry, package = "sfdep")

guerry_modeled <- guerry
guerry_lm <- lm(crime_pers ~ literacy, guerry_modeled)
guerry_modeled$predictions <- predict(guerry_lm, guerry_modeled)

ww_local_moran_i(guerry_modeled, crime_pers, predictions)
ww_local_moran(guerry_modeled, crime_pers, predictions)
```

ww_make_point_neighbors

Make 'nb' objects from point geometries

Description

This function uses `spdep::knearneigh()` and `spdep::knn2nb()` to create a "nb" neighbors list.

Usage

```
ww_make_point_neighbors(data, k = 1, sym = FALSE, ...)
```

Arguments

`data`	An `sfc_POINT` or `sfc_MULTIPOINT` object.
`k`	How many nearest neighbors to use in `spdep::knearneigh()`.
`sym`	Force the output neighbors list (from `spdep::knn2nb()`) to symmetry.
`...`	Other arguments passed to `spdep::knearneigh()`.

Value

An object of class "nb"

ww_make_polygon_neighbors

Make 'nb' objects from polygon geometries

Description

This function is an extremely thin wrapper around `spdep::poly2nb()`, renamed to use the way-wiser "ww" prefix.

Usage

```
ww_make_polygon_neighbors(data, ...)
```

Arguments

`data`	An `sfc_POLYGON` or `sfc_MULTIPOLYGON` object.
`...`	Additional arguments passed to `spdep::poly2nb()`.

Value

An object of class "nb"

Index