Package 'SSIMmap'

November 13, 2026

Title The Structural Similarity Index Measure for Maps

Version 0.4.3

Description Computes the Structural Similarity Index Measure (SSIM) and its components for raster images and irregular polygon maps. The raster method follows Wang et al. (2004) <doi:10.1109/TIP.2003.819861>. The polygon method uses geographically weighted summary statistics described by Brunsdon et al. (2002) <doi:10.1016/S0198-9715(01)00009-6> with an adaptive nearest-neighbor kernel. Provides bandwidth selection, value transformations, permutation-based inference, Fire Weather Index rasters, and Toronto census-derived example data.

Depends R (>= 3.6.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports stats, scales, terra, ggplot2 (>= 3.4.0), sf, knitr, FNN

Suggests magick, RColorBrewer, testthat (>= 3.0.0), rmarkdown, patchwork, tidyterra

URL https://github.com/Hailyee-Ha/SSIMmap

BugReports https://github.com/Hailyee-Ha/SSIMmap/issues

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Hui Jeong (Hailyee) Ha [aut, cre] (ORCID: <https://orcid.org/0000-0002-5229-4528>), Jed Long [aut] (ORCID: <https://orcid.org/0000-0003-3961-3085>)

Maintainer Hui Jeong (Hailyee) Ha <hha24@uwo.ca>

Repository CRAN

Date/Publication 2026-11-13 09:10:53 UTC

Contents

fwi_0816_bc . . . 2
fwi_0818_bc . . . 3
fwi_1101_bc . . . 3
fwi_1111_bc . . . 4
ssim_bandwidth . . . 5
ssim_polygon . . . 6
ssim_raster . . . 8
Toronto_SSIM . . . 11

Index **12**

`fwi_0816_bc` *British Columbia Fire Weather Index — 16 August 2023*

Description

Daily Canadian Fire Weather Index (FWI) for the province of British Columbia, Canada, on 16 August 2023, at a 2 km spatial resolution. This date represents conditions during the peak summer fire season, when fire danger levels in British Columbia are typically high.

Usage

```
fwi_0816_bc
```

Format

A packed `terra` raster object with one numeric layer (FWI values).

Details

The object is stored as a packed `terra` raster (created with `terra::wrap()`) so that it can be saved in an R `.rda` file. Use `terra::unwrap()` to convert it back to a regular `terra` raster before passing it to `ssim_raster` or to other `terra` functions.

Source

Canadian Forest Fire Weather Index System; FWI values are derived from meteorological inputs (temperature, wind speed, relative humidity, precipitation).

Examples

```
data("fwi_0816_bc")
fwi_0816_bc <- terra::unwrap(fwi_0816_bc)
terra::plot(fwi_0816_bc, main = "FWI - 16 August 2023")
```

fwi_0818_bc — *British Columbia Fire Weather Index — 18 August 2023*

Description

Daily Canadian Fire Weather Index (FWI) for the province of British Columbia, Canada, on 18 August 2023, at a 2 km spatial resolution. This date also represents conditions during the peak summer fire season and is typically expected to show high spatial similarity to `fwi_0816_bc`.

Usage

```
fwi_0818_bc
```

Format

A packed `terra` raster object with one numeric layer (FWI values).

Details

The object is stored as a packed `terra` raster (created with `terra::wrap()`). Use `terra::unwrap()` to convert it back to a regular `terra` raster before use.

Source

Canadian Forest Fire Weather Index System.

Examples

```
data("fwi_0818_bc")
fwi_0818_bc <- terra::unwrap(fwi_0818_bc)
terra::plot(fwi_0818_bc, main = "FWI - 18 August 2023")
```

fwi_1101_bc — *British Columbia Fire Weather Index — 1 November 2023*

Description

Daily Canadian Fire Weather Index (FWI) for the province of British Columbia, Canada, on 1 November 2023, at a 2 km spatial resolution. This date represents late-fall conditions, when fire danger across most of British Columbia is typically much lower than during the summer season due to cooler temperatures and increased precipitation. The spatial structure of this map is therefore expected to differ substantially from the summer FWI maps (`fwi_0816_bc`, `fwi_0818_bc`).

Usage

```
fwi_1101_bc
```

Format

A packed `terra` raster object with one numeric layer (FWI values).

Details

This legacy November 1 dataset is retained unchanged for compatibility. The package examples use `fwi_1111_bc`, which contains a different observation from November 11, not a renamed copy.

The object is stored as a packed `terra` raster (created with `terra::wrap()`). Use `terra::unwrap()` to convert it back to a regular `terra` raster before use.

Source

Canadian Forest Fire Weather Index System.

Examples

```
data("fwi_1101_bc")
fwi_1101_bc <- terra::unwrap(fwi_1101_bc)
terra::plot(fwi_1101_bc, main = "FWI - 1 November 2023")
```

---

fwi_1111_bc *British Columbia Fire Weather Index — 11 November 2023*

Description

Daily Canadian Fire Weather Index (FWI) for British Columbia, Canada, on 11 November 2023, at a 2 km spatial resolution. This is the late-fall observation used in the package summer-fall comparison. It is distinct from the unchanged legacy `fwi_1101_bc` data.

Usage

```
fwi_1111_bc
```

Format

A packed `terra` raster object with one numeric FWI layer, 895 rows and 510 columns, at 2,000 m resolution in EPSG:3978.

Details

The original FWI raster was cropped and masked to the 2021 British Columbia provincial boundary, with no resampling or value transformation. Its grid matches `fwi_0816_bc` and `fwi_0818_bc`. The data are stored using `terra::wrap()`; use `terra::unwrap()` before calculation. Use `transform = "none"` in `ssim_raster` to reproduce the manuscript's original-unit FWI analysis; `"none"` is also the function default from version 0.4.2. See the installed '`extdata/fwi_1111_bc_provenance.txt`' for source and preparation checksums.

Source

Natural Resources Canada, Canadian Forest Service, Canadian Wildland Fire Information System (CWFIS), https://cwfis.cfs.nrcan.gc.ca/. Historical FWI coverage for 2023-11-11, independently retrieved on 2026-10-27 and verified cell-for-cell against the original source raster 'fwi20231111.tif'. Contains information licensed under the Open Government Licence – Canada. Crop boundary: Statistics Canada, 2021 provincial boundary.

Examples

```
data("fwi_1111_bc")
fwi_1111 <- terra::unwrap(fwi_1111_bc)
terra::plot(fwi_1111, main = "FWI - 11 November 2023")
```

---

ssim_bandwidth *Bandwidth selection for polygon maps*

Description

Computes bias–variance trade-off curves for two polygon maps using an adaptive Gaussian kernel. The two input variables can be optionally transformed (e.g. normal scores or min–max normalization) before smoothing. A suggested bandwidth range is obtained by balancing standardized bias and variance for each variable, and a single bandwidth is chosen from this range.

Usage

```
ssim_bandwidth(
  shape,
  map1,
  map2,
  max_bandwidth,
  transform = c("none", "normal_score", "percentile", "minmax"),
  option = "midpoint"
)
```

Arguments

shape	An sf polygon object with attributes containing the two variables of interest.
map1	Character string; column name in shape for the first map.
map2	Character string; column name in shape for the second map.
max_bandwidth	Integer (at least 12). Upper bound of the bandwidth k for the adaptive k-nearest-neighbor kernel. Must not exceed the number of polygons.
transform	One of c("none","normal_score","percentile","minmax"). The default "none" retains the original values. "normal_score" applies a Blom normal scores transform; "percentile" maps values to empirical percentiles in (0, 1); "minmax" applies min–max normalization to [0, 1]; and "none" leaves the variables on their original scale.

option — Character string specifying how to choose a single bandwidth from the bias–variance trade-off range: one of "midpoint", "lower", or "upper" (default "midpoint"). The square-root of the number of polygons (sqrt(n)) is also shown on the plot as a reference.

Details

This function does *not* compute SSIM; it is intended to be used in combination with ssim_polygon(), which can then be called with the selected bandwidth.

The kernel is Gaussian with adaptive bandwidth:

$$w_{ij} = \exp\left\{-\tfrac{1}{2}(d_{ij}/h_i)^2\right\},$$

where d_{ij} is the distance between polygon centroids and h_i is the distance from polygon i to its k-th nearest neighbor. Bandwidth selection is based purely on bias–variance trade-off and does not involve any SSIM computation or permutation testing.

Value

A list with components:

plot — A ggplot object showing the bias–variance trade-off curves for both variables, the suggested bandwidth range, the chosen bandwidth, and sqrt(n) as a reference.

bandwidth — The chosen bandwidth k_star.

tradeoff — A list with elements map1, map2, kA, kB, and k_sqrt, where map1 and map2 are data frames containing the standardized bias and variance across k for each variable.

---

ssim_polygon *Structural similarity index (SSIM) for polygon maps*

Description

Computes local and global SSIM (and its components SIM, SIV, SIP) for two polygon maps using an adaptive k-nearest-neighbor (k-NN) Gaussian kernel. Input variables can be optionally transformed (e.g. rank-based inverse normal scores or min–max normalization). Optionally performs permutation tests for global and local significance with BH-FDR correction. Under the null hypothesis for the permutation test, both variables are randomly permuted at each iteration, breaking spatial structure and association between them. Local p-values can be optionally adjusted using the Benjamini–Hochberg FDR procedure.

Usage

```
ssim_polygon(
  shape,
  map1,
  map2,
  global = TRUE,
  bandwidth = NULL,
  transform = c("none", "normal_score", "percentile", "minmax"),
  k1 = NULL,
  k2 = NULL,
  do_test = FALSE,
  R = 1000,
  fdr = TRUE,
  alpha = 0.05,
  seed = NULL
)
```

Arguments

shape	An sf polygon object.
map1	Character string; column name in shape for the first map.
map2	Character string; column name in shape for the second map.
global	Logical. If TRUE, compute and print a summary table of global SSIM, SIM, SIV, SIP. If FALSE, return an sf object with local metrics (and local p-/q-values if do_test = TRUE).
bandwidth	Integer; adaptive k-NN size (number of neighbors). The default is ceiling(sqrt(n)), where n is the number of polygons. Must be at least 3 and not exceed n.
transform	One of c("none", "normal_score", "percentile", "minmax"). The default "none" retains the original values. "normal_score" applies a Blom normal scores transform; "percentile" maps values to empirical percentiles in (0, 1); "minmax" applies min–max normalization to [0, 1]; and "none" leaves the variables on their original scale.
k1, k2	SSIM constants. If NULL, defaults are k1 = 0.01, k2 = 0.03.
do_test	Logical; if TRUE, perform permutation tests for global and local SSIM, SIM, SIV, and SIP. Tests are two-sided and centered on each statistic's permutation mean.
R	Integer; number of permutations for the significance tests (default R = 1000).
fdr	Logical; if TRUE (default), apply Benjamini–Hochberg FDR correction across polygons separately for each local statistic.
alpha	Numeric; significance threshold for local results (default 0.05). Local features with q <= alpha are flagged as significant when fdr = TRUE; otherwise use p <= alpha.
seed	Optional integer; random seed for reproducibility of the permutation tests.

Value

If `global = TRUE` and `do_test = FALSE`, the function prints a knitr table summarizing the global mean, minimum, maximum, and standard deviation of SSIM, SIM, SIV, and SIP, and returns this `data.frame` (invisibly).

If `global = TRUE` and `do_test = TRUE`, the function prints the same summary table plus a table of global permutation p-values (two-sided) for SSIM, SIM, SIV, and SIP, and returns (invisibly) a `list` with components:

- `summary`: `data.frame` with global SSIM/SIM/SIV/SIP summary statistics;
- `p_global`: `data.frame` with global means and permutation p-values for SSIM, SIM, SIV, SIP.

If `global = FALSE`, the function returns an `sf` object equal to `shape` with additional columns:

- `SSIM`, `SIM`, `SIV`, `SIP`: local similarity metrics for each polygon;
- With `do_test = TRUE`, each statistic has columns with suffixes `_p`, `_q`, and `_sig`, for example `SIM_p`, `SIM_q`, and `SIM_sig`. These contain its local permutation p-value, BH-adjusted q-value, and logical significance flag (`q <= alpha`). When `fdr = FALSE`, the `_q` columns equal the unadjusted p-values.
- `p_value`, `q_value`, `sig` remain aliases for `SSIM_p`, `SSIM_q`, `SSIM_sig`, respectively, for compatibility with earlier versions.

`ssim_raster` *SSIM index for raster images*

Description

This function calculates the Structural Similarity Index Measure (SSIM) between two raster images. It can return either global SSIM summaries or local SSIM raster layers. Optional permutation-based statistical inference can also be performed for global or local SSIM values.

Usage

```
ssim_raster(
  map1,
  map2,
  global = TRUE,
  w = 1,
  transform = c("none", "normal_score", "percentile", "minmax"),
  k1 = NULL,
  k2 = NULL,
  do_test = FALSE,
  local_test = FALSE,
  R = 1000,
  fdr = TRUE,
  alpha = 0.05,
  seed = NULL
)
```

Arguments

map1	A single-layer terra::SpatRaster object representing the first raster map.
map2	A single-layer terra::SpatRaster object representing the second raster map.
global	Logical. If TRUE, the function returns global summary statistics of SSIM, SIM, SIV, and SIP. If FALSE, it returns local raster layers for each cell. Default is TRUE.
w	Integer. Radius of the local moving window used to calculate local SSIM components. For example, w = 1 uses a 3 x 3 window, and w = 2 uses a 5 x 5 window. Default is 1.
transform	Character. Transformation applied to both raster maps before SSIM calculation. Options are "none", "normal_score", "percentile", and "minmax". Default is "none".
k1	Numeric. Constant used to stabilize the luminance component of SSIM. If NULL, the default value 0.01 is used.
k2	Numeric. Constant used to stabilize the contrast and structure components of SSIM. If NULL, the default value 0.03 is used.
do_test	Logical. If TRUE, a permutation-based statistical test is performed. Default is FALSE.
local_test	Logical. If TRUE and global = FALSE, local permutation tests are performed for each raster cell. If FALSE, local SSIM layers are returned without p-values. Default is FALSE.
R	Integer. Number of permutations used for statistical inference. Default is 1000.
fdr	Logical. If TRUE, p-values are adjusted using the Benjamini-Hochberg false discovery rate correction. Default is TRUE.
alpha	Numeric. Significance level used to determine significant SSIM values. Default is 0.05. Significance includes equality: q <= alpha with FDR adjustment, otherwise p <= alpha.
seed	Optional integer. Random seed used for permutation testing. Default is NULL.

Details

The SSIM index is calculated as the product of three components: similarity in mean intensity, similarity in variance, and similarity in spatial pattern. These components are returned as SIM, SIV, and SIP, respectively.

The local SSIM calculation uses a square moving window with size (2 * w + 1) x (2 * w + 1).

When do_test = TRUE, permutation testing is performed by randomly permuting the values of map2 over the overlapping non-NA cells and recalculating SSIM values while holding map1 fixed. Both global and local tests are two-sided, centered on the corresponding permutation mean: p = (1 + sum(abs(null - mean(null)) >= abs(observed - mean(null)))) / (R + 1). This tests departures in either direction from the value-randomization distribution, not similarity to zero. Local tests estimate a separate permutation mean for each cell and component. They replay the same permutations in a second pass to avoid storing all local null layers; the final random-number state is the same as after one pass. With fdr = TRUE, BH adjustment is applied across the four global components or across valid cells separately for each local component.

Value

If `global = TRUE` and `do_test = FALSE`, a list containing a summary table of global SSIM, SIM, SIV, and SIP values is returned.

If `global = TRUE` and `do_test = TRUE`, a list containing the summary table, permutation null distributions, p-values, and optionally q-values is returned.

If `global = FALSE` and `do_test = FALSE`, a `terra::SpatRaster` object with four layers is returned: `SSIM`, `SIM`, `SIV`, and `SIP`.

If `global = FALSE`, `do_test = TRUE`, and `local_test = TRUE`, a `terra::SpatRaster` object is returned with local SSIM component layers, p-value layers, optionally q-value layers, and significance layers.

Examples

```
## Not run:
library(terra)

# Global SSIM summary
g1 <- ssim_raster(
  map1,
  map2,
  global = TRUE,
  w = 1
)

# Local SSIM raster layers
l1 <- ssim_raster(
  map1,
  map2,
  global = FALSE,
  w = 1
)

# Local SSIM with permutation test
l2 <- ssim_raster(
  map1,
  map2,
  global = FALSE,
  w = 1,
  do_test = TRUE,
  local_test = TRUE,
  R = 999
)

## End(Not run)
```

Toronto_SSIM *Toronto neighborhood deprivation indices and commuting measure (2021)*

Description

An `sf` polygon object containing 279 Aggregate Dissemination Areas (ADAs) for the Toronto Census Subdivision, Ontario, Canada. It includes two neighborhood deprivation indices and a derived commuting measure, for use with `ssim_polygon` and `ssim_bandwidth`.

Usage

```
Toronto_SSIM
```

Format

An `sf` data frame with 279 rows and the following columns:

ADAUID 2021 Aggregate Dissemination Area identifier.

CIMD Canadian Index of Multiple Deprivation residential instability score.

PP Pampalon material and social deprivation index social dimension.

Commute Preserved derived commuting percentage. Its DA numerator is the Census count for commuting to a different CSD and CD within the province or territory of residence (characteristic 2601), divided by total DA Census population (characteristic 1), multiplied by 100, and rounded half-up to two decimals. The ADA value is the unweighted mean of all stored percentages for all intersecting DAs, including boundary-only intersections and stored zero percentages. Zero-population DAs also have stored zero percentages and are included. It is not a household or worker share, a pooled ADA population rate, or a count of all outside-CSD commuting. Some source numerator cells are suppressed in the official release but stored as zero in the preserved data; see the provenance note.

geometry `sfc` polygon geometry column.

Details

The numeric data are unchanged from version 0.4.0. Earlier documentation incorrectly described these polygons as 2016 Census Tracts. The preserved commuting variable is a derived DA-average measure, not a proportion of households. See '`extdata/Toronto_SSIM_provenance.txt`' for its calculation and the suppressed-value limitation.

Source

Statistics Canada, 2021 Census of Population; author-prepared neighborhood deprivation and commuting data.

Examples

```
data("Toronto_SSIM")
plot(Toronto_SSIM["CIMD"], border = NA)
```

Index

∗ **datasets**
fwi_0816_bc, 2
fwi_0818_bc, 3
fwi_1101_bc, 3
fwi_1111_bc, 4
Toronto_SSIM, 11

fwi_0816_bc, 2, *3, 4*
fwi_0818_bc, *3*, 3, *4*
fwi_1101_bc, 3, *4*
fwi_1111_bc, *4*, 4

ssim_bandwidth, 5, *11*
ssim_polygon, 6, *11*
ssim_raster, *2*, *4*, 8

Toronto_SSIM, 11