Package: wingen (via r-universe)

August 28, 2024

Title Continuous Mapping of Genetic Diversity

Version 2.1.2

Description Generate continuous maps of genetic diversity using moving windows with options for rarefaction, interpolation, and masking as described in Bishop et al. (2023) <doi:10.1111/2041-210X.14090>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports automap, crayon, dplyr, furrr, gdistance, graphics, grDevices, ggplot2, hierfstat, magrittr, pegas, purrr, raster, rlang, sf, terra, tidyr, tidyselect, utils, vcfR, viridis

Suggests adegenet, covr, devtools, future, knitr, MASS, rmarkdown, stringr, SpatialKDE, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

URL https://github.com/AnushaPB/wingen

BugReports https://github.com/AnushaPB/wingen/issues

Depends R (>= 3.5.0)

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

RemoteUrl https://github.com/anushapb/wingen

RemoteRef HEAD

RemoteSha 5d3a6e90e79fadcfbe0292be2c1c59899eec3dcc

Contents

`circle_gd` *Create a moving window map of genetic diversity using a circle window*

Description

Generate a continuous raster map of genetic diversity using circle moving windows

Usage

```
circle_gd(
  gen,
  coords,
  lyr,
  maxdist,
  distmat = NULL,
```

```
  stat = "pi",
  fact = 0,
  rarify = FALSE,
  rarify_n = 2,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05
)
```

Arguments

gen	genetic data either as an object of type vcf or a path to a vcf file (*note:* order matters! The coordinate and genetic data should be in the same order; there are currently no checks for this)
coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections)
maxdist	maximum geographic distance used to define neighborhood; any samples further than this distance will not be included (this can be thought of as the neighborhood radius) Can either be (1) a single numeric value or (2) a SpatRaster where each pixel is the maximum distance to be used for that cell on the landscape (must be the same spatial scale as `lyr`).
distmat	distance matrix output from get_geodist (optional; can be used to save time on distance calculations)
stat	genetic diversity statistic(s) to calculate (see Details, defaults to `"pi"`). Can be a single statistic or a vector of statistics
fact	aggregation factor to apply to `lyr` (defaults to 0; *note:* increasing this value reduces computational time)
rarify	if rarify = TRUE, rarefaction is performed (defaults to FALSE)
rarify_n	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
rarify_nit	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to `"all"` to use all possible combinations of samples of size `rarify_n` within the window.
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
fun	function to use to summarize rarefaction results (defaults to mean, must take `na.rm = TRUE` as an argument)
L	for calculating `"pi"`, L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen

default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)

`rarify_alleles` for calculating `"biallelic_richness"`, whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)

`sig` for calculating `"hwe"`, significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)

Details

Coordinates and rasters should be in a Euclidean coordinate system (i.e., UTM coordinates) such that raster cell width and height are equal distances. As such, longitude-latitude systems should be transformed before using dist_gd. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Coordinates and rasters should be in a projected (planar) coordinate system such that raster cells are of equal sizes. Therefore, spherical systems (including latitute-longitude coordinate systems) should be projected prior to use. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Current genetic diversity metrics that can be specified with `stat` include:

- `"pi"` for nucleotide diversity (default) calculated using `hierfstat` pi.dosage. Use the `L` argument to set the sequence length (defaults to dividing by the number of variants).
- `"Ho"` for average observed heterozygosity across all sites
- `"allelic_richness"` for average number of alleles across all sites
- `"biallelic_richness"` for average allelic richness across all sites for a biallelic dataset (this option is faster than `"allelic_richness"`)
- `"hwe"` for the proportion of sites that are not in Hardy–Weinberg equilibrium, calculated using `pegas` hw.test at the 0.05 level (other alpha levels can be specified by adding the sig argument; e.g., `sig = 0.10`).
- `"basic_stats"` for a series of statistics produced by `hierfstat` basic.stats including mean observed heterozygosity (same as Ho), mean gene diversities within population (Hs), Gene diversities overall (Ht), and Fis following Nei (1987). Population-based statistics (e.g., FST) normally reported by basic.stats are not included as they are not meaningful within the individual-based moving windows.

Value

SpatRaster that includes a raster layer of genetic diversity and a raster layer of the number of samples within the window for each cell

Examples

```
load_mini_ex()
cpi <- circle_gd(mini_vcf, mini_coords, mini_lyr, fact = 2, maxdist = 5)
```

`circle_general` *General function for making circular moving window maps*

Description

Generate a continuous raster map using circular moving windows. While resist_gd is built specifically for making maps of genetic diversity from vcfs,`circle_general` can be used to make maps from different data inputs. Unlike `resist_gd`, `resist_general` will not convert your data into the correct format for calculations of different diversity metrics. See details for how to format data inputs for different statistics.

Usage

```
circle_general(
  x,
  coords,
  lyr,
  maxdist,
  distmat = NULL,
  stat,
  fact = 0,
  rarify = FALSE,
  rarify_n = 2,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05,
  ...
)
```

Arguments

`x`	data to be summarized by the moving window (*note:* order matters! `coords` should be in the same order, there are currently no checks for this). The class of `x` required depends on the statistic being calculated (see the `stat` argument and the function description for more details)
`coords`	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
`lyr`	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections)
`maxdist`	maximum geographic distance used to define neighborhood; any samples further than this distance will not be included (this can be thought of as the neighborhood radius) Can either be (1) a single numeric value or (2) a SpatRaster where

	each pixel is the maximum distance to be used for that cell on the landscape (must be the same spatial scale as `lyr`).
`distmat`	distance matrix output from get_geodist (optional; can be used to save time on distance calculations)
`stat`	moving window statistic to calculate (see details). `stat` can generally be set to any function that will take `x`as input and return a single numeric value (for example, `x` can be a vector and `stat` can be set equal to a summary statistic like `mean`, `sum`, or `sd`)
`fact`	aggregation factor to apply to `lyr` (defaults to 0; *note:* increasing this value reduces computational time)
`rarify`	if rarify = TRUE, rarefaction is performed (defaults to FALSE)
`rarify_n`	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
`rarify_nit`	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to `"all"` to use all possible combinations of samples of size `rarify_n` within the window.
`min_n`	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
`fun`	function to use to summarize rarefaction results (defaults to mean, must take `na.rm = TRUE` as an argument)
`L`	for calculating `"pi"`, L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)
`rarify_alleles`	for calculating `"biallelic_richness"`, whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)
`sig`	for calculating `"hwe"`, significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)
`...`	if a function is provided for `stat`, additional arguments to pass to the `stat` function (e.g. if `stat = mean`, users may want to set `na.rm = TRUE`)

Details

To calculate genetic diversity statistics with the built in wingen functions, data must be formatted as such:

- for `"pi"` or `"biallelic_richness"`, `x` must be a dosage matrix with values of 0, 1, or 2
- for `"Ho"`, `x` must be a heterozygosity matrix where values of 0 = homozygosity and values of 1 = heterozygosity
- for `"allelic_richness"` or `"hwe`, `x` must be a `genind` type object
- for `"basic_stats"`, `x` must be a `hierfstat` type object

Otherwise, `stat` can be any function that takes a matrix or data frame and outputs a single numeric value (e.g., a function that produces a custom diversity index); however, this should be attempted with caution since this functionality has not have been tested extensively and may produce errors.

Value

SpatRaster that includes a raster layer of genetic diversity and a raster layer of the number of samples within the window for each cell

---

`coords_to_raster` *Create a raster from coordinates*

Description

Generate a raster layer from coordinates which can be used in window_gd as the RasterLayer to move the window across

Usage

```
coords_to_raster(
  coords,
  buffer = 0,
  res = 1,
  agg = NULL,
  disagg = NULL,
  plot = FALSE
)
```

Arguments

`coords`	coordinates of samples as sf points, a SpatVector, a two-column matrix, or a data.frame with x and y coordinates
`buffer`	size of buffer to add to edge of raster (defaults to 0)
`res`	desired resolution of raster (defaults to 1). Can be a single value for square cells or a vector with two values representing x and y resolutions
`agg`	aggregation factor to apply to raster (defaults to NULL)
`disagg`	disaggregation factor to apply to raster (defaults to NULL)
`plot`	whether to plot resulting raster with coords (defaults to FALSE)

Value

RasterLayer

Examples

```
load_mini_ex()
coords_to_raster(mini_coords, buffer = 1, plot = TRUE)
```

get_geodist *Get a matrix of geographic distances for circle_gd*

Description

Create a distance matrix based on coordinates and a raster layer. The output is a distance matrix where rows represent cells on the landscape and columns represent individual locations on the landscape. Each value is the geographic distance between each individual and each cell calculated using st_distance. This matrix is used by circle_gd. If coords_only = TRUE, the result is a distance matrix for the sample coordinates only.

Usage

```
get_geodist(coords, lyr = NULL, fact = 0, coords_only = FALSE)
```

Arguments

coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	SpatRaster or RasterLayer for generating distances (not required if coords_only = TRUE)
fact	aggregation factor to apply to lyr (defaults to 0; *note:* increasing this value reduces computational time)
coords_only	whether to return distances only for sample coordinates

Value

a distance matrix used by circle_gd

Examples

```
load_mini_ex()
distmat <- get_geodist(mini_coords, mini_lyr)
```

get_resdist *Get a matrix of resistance distances for resist_gd*

Description

Create a distance matrix based on coordinates and a connectivity layer. The output is a distance matrix where rows represent cells on the landscape and columns represent individual locations on the landscape. Each value is the resistance distance between each sample and each cell calculated using the gdistance package. This matrix is used by resist_gd. If coords_only = TRUE, the result is a distance matrix for the sample coordinates only.

Usage

```
get_resdist(
  coords,
  lyr,
  fact = 0,
  transitionFunction = mean,
  directions = 8,
  geoCorrection = TRUE,
  coords_only = FALSE
)
```

Arguments

coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	conductivity layer (higher values should mean greater conductivity) for generating distances. Can be either a SpatRaster or RasterLayer.
fact	aggregation factor to apply to lyr (defaults to 0; *note:* increasing this value reduces computational time)
transitionFunction	function to calculate transition values from grid values (defaults to mean)
directions	directions in which cells are connected (4, 8, 16, or other), see adjacent (defaults to 8)
geoCorrection	whether to apply correction to account for local distances (defaults to TRUE). Geographic correction is necessary for all objects of the class Transition that are either: (1) based on a grid in a geographic (lonlat) projection and covering a large area; (2) made with directions > 4 (see geoCorrection for more details).
coords_only	whether to return distances only for sample coordinates

Value

a distance matrix used by resist_gd

Examples

```
load_mini_ex()
distmat <- get_resdist(mini_coords, mini_lyr)
```

---

ggplot_count *Plot moving window map of sample counts*

Description

Plot sample counts layer produced by window_gd or krig_gd

Usage

```
ggplot_count(x, index = NULL, col = viridis::mako(100))
```

Arguments

x	single SpatRaster of counts or SpatRaster where indexed layer is sample counts
index	index of raster layers to plot (defaults to plotting the one called "sample_count", if more than one layer is provided)
col	color palette to use for plotting (defaults to viridis::mako palette)

Value

list of ggplots

Examples

```
data("mini_lyr")
ggplot_count(mini_lyr)
```

---

ggplot_gd *Plot moving window map of genetic diversity*

Description

Plot genetic diversity layer produced by window_gd or krig_gd

Usage

```
ggplot_gd(x, bkg = NULL, index = NULL, col = viridis::magma(100))
```

Arguments

x	output from window_gd or krig_gd (RasterStack where first layer is genetic diversity)
bkg	optional raster or sf polygon
index	index of raster layers to plot (defaults to plotting all of the layers except the one called "sample_count", if more than one layer is provided)
col	color palette to use for plotting (defaults to magma palette)

Value

list of ggplots

Examples

```
data("mini_lyr")
ggplot_gd(mini_lyr)
```

krig_gd *Krige moving window maps*

Description

Perform interpolation of the raster(s) produced by window_gd using autoKrige

Usage

```
krig_gd(
  r,
  grd = NULL,
  index = 1,
  coords = NULL,
  agg_grd = NULL,
  disagg_grd = NULL,
  agg_r = NULL,
  disagg_r = NULL,
  autoKrige_output = FALSE,
  lower_bound = TRUE,
  upper_bound = TRUE,
  krig_method = "ordinary",
  resample = FALSE,
  resample_first = TRUE
)
```

Arguments

r	SpatRaster produced by window_gd
grd	object to create grid for kriging; can be a SpatRaster or RasterLayer. If undefined, will use r to create a grid.
index	integer indices of layers in raster stack to krige (defaults to 1; i.e., the first layer)
coords	if provided, kriging will occur based only on values at these coordinates. Can be provided as an sf points, a two-column matrix, or a data.frame representing x and y coordinates
agg_grd	factor to use for aggregation of grd, if provided (this will decrease the resolution of the final kriged raster; defaults to NULL)
disagg_grd	factor to use for disaggregation of grd, if provided (this will increase the resolution of the final kriged raster; defaults to NULL)
agg_r	factor to use for aggregation of r, if provided (this will decrease the number of points used in the kriging model; defaults to NULL)
disagg_r	factor to use for disaggregation, of r if provided (this will increase the number of points used in the kriging model; defaults to NULL)

autoKrige_output
: whether to return full output from autoKrige including uncertainty rasters (defaults to FALSE). If TRUE, returns a list with the kriged input raster layer ("raster"), kriged variance ("var"), kriged standard deviation ("stdev"), and full autoKrige output ("autoKrige_output").

lower_bound
: if TRUE (default), converts all values in the kriged raster less than the minimum value of the input raster, to that minimum

upper_bound
: if TRUE (default), converts all values in the kriged raster greater than the maximum value of the input raster, to that maximum

krig_method
: method to use for kriging. If `ordinary`, ordinary/simple kriging is performed (formula: ~ 1; default). If `universal`, universal kriging is performed (formula = ~ x + y).

resample
: whether to resample `grd` or `r`. Set to `"r"` to resample `r` to `grd`. Set to `"grd"` to resample `grd` to `r` (defaults to FALSE for no resampling)

resample_first
: if aggregation or disaggregation is used in addition to resampling, specifies whether to resample before (resample_first = TRUE) or after (resample_first = FALSE) aggregation/disaggregation (defaults to TRUE)

Value

a SpatRaster object or a list of autoKrige outputs (if autoKrige_output = TRUE)

Examples

```
load_mini_ex()
wpi <- window_gd(mini_vcf, mini_coords, mini_lyr, L = 10, rarify = TRUE)
kpi <- krig_gd(wpi, mini_lyr)
plot_gd(kpi, main = "Kriged Pi")
```

---

load_middle_earth_ex *Middle earth example*

Description

Loads middle earth example data

Usage

```
load_middle_earth_ex(quiet = FALSE)
```

Arguments

quiet
: whether to hide message (defaults to FALSE)

Value

three objects are loaded (lotr_vcf, lotr_coords, and lotr_lyrs)

Examples

```
load_middle_earth_ex()
```

`load_mini_ex` *Mini middle earth example*

Description

Loads mini middle earth example data

Usage

```
load_mini_ex(quiet = FALSE)
```

Arguments

`quiet`	whether to hide message (defaults to FALSE)

Value

three objects are assigned in the GlobalEnv (vcf, coords, and lyr)

Examples

```
load_mini_ex()
```

`lotr_coords` *Middle earth example coordinates*

Description

Middle earth example coordinates

Usage

```
lotr_coords
```

Format

A data frame with 100 rows and 2 columns

x x coordinate

y y coordinate

Source

created from simulations in Bishop et al. (2023)

lotr_lyr *Middle earth example raster*

Description

RasterLayer of middle earth based on an example digital elevation model of Tolkien's Middle Earth produced by the Center for Geospatial Analysis at William & Mary (Robert, 2020).

Usage

```
lotr_lyr
```

Format

RasterLayer

Source

created from simulations in Bishop et al. (2023) based on Rose, Robert A. (2020) GIS & Middle Earth Presentation & Data Set. William & Mary. https://doi.org/10.21220/RKEZ-X707

lotr_range *Middle earth example range polygon*

Description

sf polygon of range map

Usage

```
lotr_range
```

Format

sf

Source

created from simulations in Bishop et al. (2023)

lotr_vcf *Middle earth example vcf*

Description

A Variant Call Format data set

Usage

```
lotr_vcf
```

Format

Object of class vcfR with 100 individuals and 1000 loci

Source

created from simulations in Bishop et al. (2023)

mask_gd *Mask moving window maps*

Description

Mask genetic diversity layer produced by window_gd or krig_gd

Usage

```
mask_gd(x, y, minval = NULL, maxval = NULL)
```

Arguments

x	Raster object to mask
y	Raster object or Spatial object to use as mask
minval	if y is a Raster object, value of y below which to mask
maxval	if y is a Raster object, value of y above which to mask

Value

RasterLayer

Examples

```
data("mini_lyr")
mpi <- mask_gd(mini_lyr, mini_lyr, minval = 0.01)
```

mini_coords *Mini middle earth example coordinates*

Description

Mini middle earth example coordinates

Usage

```
mini_coords
```

Format

A data frame with 10 rows and 2 columns

x x coordinate

y y coordinate

Source

created from simulations in Bishop et al. (2023)

mini_lyr *Mini middle earth example raster*

Description

Small RasterLayer of middle earth based on an example digital elevation model of Tolkien's Middle Earth produced by the Center for Geospatial Analysis at William & Mary (Robert, 2020).

Usage

```
mini_lyr
```

Format

A RasterLayer of middle earth

Source

created from simulations in Bishop et al. (2023)

mini_vcf *Mini middle earth example vcf*

Description

A Variant Call Format data set

Usage

```
mini_vcf
```

Format

Object of class vcfR with 10 individuals and 10 loci

Source

created from simulations in Bishop et al. (2023)

mini_vcf_NA *Mini middle earth example vcf with NA values*

Description

A Variant Call Format data set with NA values

Usage

```
mini_vcf_NA
```

Format

Object of class vcfR with 10 individuals and 10 loci

Source

created from simulations in Bishop et al. (2023)

plot_count *Plot moving window map of sample counts*

Description

Plot sample counts layer produced by window_gd or krig_gd

Usage

```
plot_count(
  x,
  index = NULL,
  breaks = 100,
  col = viridis::mako(breaks),
  main = NULL,
  box = FALSE,
  range = NULL,
  legend = TRUE,
  ...
)
```

Arguments

x	single SpatRaster of counts or SpatRaster where indexed layer is sample counts
index	if a raster stack is provided, index of the sample count layer to plot (defaults to plotting the layer named "sample_count" or the last layer of the stack)
breaks	number of breaks to use in color scale (defaults to 10)
col	color palette to use for plotting (defaults to viridis::magma palette)
main	character. Main plot titles (one for each layer to be plotted). You can use arguments cex.main, font.main, col.main to change the appearance; and loc.main to change the location of the main title (either two coordinates, or a character value such as "topleft")
box	whether to include a box around the raster plot (defaults to FALSE)
range	numeric. minimum and maximum values to be used for the continuous legend
legend	whether to include legend
...	arguments passed to plot("SpatRaster", "numeric") and additional graphical arguments

Value

plot of sample counts

Examples

```
data("mini_lyr")
plot_count(mini_lyr)
```

plot_gd *Plot moving window map of genetic diversity*

Description

Plot genetic diversity layer produced by window_gd or krig_gd

Usage

```
plot_gd(
  x,
  bkg = NULL,
  index = NULL,
  col = viridis::magma(breaks),
  breaks = 100,
  main = NULL,
  box = FALSE,
  range = NULL,
  legend = TRUE,
  ...
)
```

Arguments

x	output from window_gd or krig_gd (SpatRaster where first layer is genetic diversity)
bkg	optional SpatRaster or other spatial object that will be plotted as the "background" in gray
index	if a raster stack is provided, index of the layer to plot (defaults to plotting all layers except layers named "sample_count")
col	color palette to use for plotting (defaults to magma palette)
breaks	number of breaks to use in color scale (defaults to 100)
main	character. Main plot titles (one for each layer to be plotted). You can use arguments cex.main, font.main, col.main to change the appearance; and loc.main to change the location of the main title (either two coordinates, or a character value such as "topleft")
box	whether to include a box around the Raster plot (defaults to FALSE)
range	numeric. minimum and maximum values to be used for the continuous legend
legend	whether to include legend
...	arguments passed to plot("SpatRaster", "numeric") and additional graphical arguments

Value

plot of genetic diversity

Examples

```
data("mini_lyr")
plot_gd(mini_lyr)
```

`preview_gd` *Preview moving window and sample counts*

Description

Generate a preview of moving window size and sample counts based on the coordinates and parameters to be supplied to window_gd, circle_gd, or resist_gd. The method to be used should be specified with `method = "window"`, `"circle"`, or `"resist"`. For `method = "window"`, `wdim` must be specified. For `method = "circle"` or `"resist"`, `maxdist` must be specified and `distmat` can also optionally be specified.

Usage

```
preview_gd(
  lyr,
  coords,
  method = "window",
  wdim = 3,
  maxdist = NULL,
  distmat = NULL,
  fact = 0,
  sample_count = TRUE,
  min_n = 0,
  plot = TRUE
)
```

Arguments

`lyr`	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections). For `method = "resist"` this should also be the conductivity layer (see resist_gd)
`coords`	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
`method`	which method to use to create preview (`"window"` for window_gd, `"circle"` for circle_gd, or `"resist"` for resist_gd; defaults to `"window"`)
`wdim`	if `method = "window"`, dimensions (height x width) of window; if only one value is provided, a square window is created (defaults to 3 x 3 window)
`maxdist`	if `method = "circle"` or `method = "resist`, the maximum geographic distance used to define the neighborhood; any samples further than this distance will not be included (see get_geodist or get_resdist)

distmat	if method = "circle" or method = "resist", an optional distance matrix to be used output from either get_geodist or get_resdist, respectively. If not provided, one will be automatically calculated.
fact	aggregation factor to apply to lyr (defaults to 0; *note:* increasing this value reduces computational time)
sample_count	whether to create plot of sample counts for each cell (defaults to TRUE)
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA)
plot	whether to plot results (default = TRUE)

Details

Coordinates and rasters should be in a projected (planar) coordinate system such that raster cells are of equal sizes. Therefore, spherical systems (including latitute-longitude coordinate systems) should be projected prior to use. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Value

Plots preview of window and returns SpatRaster with sample counts layer (if sample_count = TRUE)

Examples

```
load_mini_ex()
preview_gd(mini_lyr, mini_coords, wdim = 3, fact = 3, sample_count = TRUE, min_n = 2)
```

---

resist_gd *Create a moving window map of genetic diversity based on resistance*

Description

Generate a continuous raster map of genetic diversity using resistance distances calculated with a conductivity surface

Usage

```
resist_gd(
  gen,
  coords,
  lyr,
  maxdist,
  distmat = NULL,
  stat = "pi",
  fact = 0,
  rarify = FALSE,
```

```
  rarify_n = 2,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05,
  transitionFunction = mean,
  directions = 8,
  geoCorrection = TRUE
)
```

Arguments

gen	genetic data either as an object of type vcf or a path to a vcf file (*note:* order matters! The coordinate and genetic data should be in the same order; there are currently no checks for this)
coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	conductivity layer (higher values should mean greater conductivity) to move window across. Can be either a SpatRaster or RasterLayer.
maxdist	maximum cost distance used to define neighborhood; any samples further than this cost distance will not be included (this can be thought of as the neighborhood radius, but in terms of cost distance). Can either be (1) a single numeric value or (2) a SpatRaster where each pixel is the maximum distance to be used for that cell on the landscape (must be the same spatial scale as lyr).
distmat	distance matrix output from get_resdist (optional; can be used to save time on distance calculations)
stat	genetic diversity statistic(s) to calculate (see Details, defaults to "pi"). Can be a single statistic or a vector of statistics
fact	aggregation factor to apply to lyr (defaults to 0; *note:* increasing this value reduces computational time)
rarify	if rarify = TRUE, rarefaction is performed (defaults to FALSE)
rarify_n	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
rarify_nit	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to "all" to use all possible combinations of samples of size rarify_n within the window.
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
fun	function to use to summarize rarefaction results (defaults to mean, must take na.rm = TRUE as an argument)
L	for calculating "pi", L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen

	default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)
`rarify_alleles`	for calculating `"biallelic_richness"`, whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)
`sig`	for calculating `"hwe"`, significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)
`transitionFunction`	function to calculate transition values from grid values (defaults to mean)
`directions`	directions in which cells are connected (4, 8, 16, or other), see adjacent (defaults to 8)
`geoCorrection`	whether to apply correction to account for local distances (defaults to TRUE). Geographic correction is necessary for all objects of the class Transition that are either: (1) based on a grid in a geographic (lonlat) projection and covering a large area; (2) made with directions > 4 (see geoCorrection for more details).

Details

Coordinates and rasters should be in a Euclidean coordinate system (i.e., UTM coordinates) such that raster cell width and height are equal distances. As such, longitude-latitude systems should be transformed before using dist_gd. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Coordinates and rasters should be in a projected (planar) coordinate system such that raster cells are of equal sizes. Therefore, spherical systems (including latitute-longitude coordinate systems) should be projected prior to use. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Current genetic diversity metrics that can be specified with `stat` include:

- `"pi"` for nucleotide diversity (default) calculated using `hierfstat` pi.dosage. Use the `L` argument to set the sequence length (defaults to dividing by the number of variants).
- `"Ho"` for average observed heterozygosity across all sites
- `"allelic_richness"` for average number of alleles across all sites
- `"biallelic_richness"` for average allelic richness across all sites for a biallelic dataset (this option is faster than `"allelic_richness"`)
- `"hwe"` for the proportion of sites that are not in Hardy–Weinberg equilibrium, calculated using `pegas` hw.test at the 0.05 level (other alpha levels can be specified by adding the sig argument; e.g., `sig = 0.10`).
- `"basic_stats"` for a series of statistics produced by `hierfstat` basic.stats including mean observed heterozygosity (same as Ho), mean gene diversities within population (Hs), Gene diversities overall (Ht), and Fis following Nei (1987). Population-based statistics (e.g., FST) normally reported by basic.stats are not included as they are not meaningful within the individual-based moving windows.

Value

SpatRaster that includes a raster layer of genetic diversity and a raster layer of the number of samples within the window for each cell

Examples

```
load_mini_ex()
rpi <- resist_gd(mini_vcf, mini_coords, mini_lyr, maxdist = 50)
```

`resist_general` *General function for making resistance-based maps*

Description

Generate a continuous raster map using resistance distances. While resist_gd is built specifically for making maps of genetic diversity from vcfs,`resist_general` can be used to make maps from different data inputs. Unlike `resist_gd`, `resist_general` will not convert your data into the correct format for calculations of different diversity metrics. See details for how to format data inputs for different statistics.

Usage

```
resist_general(
  x,
  coords,
  lyr,
  maxdist,
  distmat = NULL,
  stat,
  fact = 0,
  rarify = FALSE,
  rarify_n = 2,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05,
  transitionFunction = mean,
  directions = 8,
  geoCorrection = TRUE,
  ...
)
```

Arguments

`x`	data to be summarized by the moving window (*note:* order matters! `coords` should be in the same order, there are currently no checks for this). The class of `x` required depends on the statistic being calculated (see the `stat` argument and the function description for more details)

coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections)
maxdist	maximum cost distance used to define neighborhood; any samples further than this cost distance will not be included (this can be thought of as the neighborhood radius, but in terms of cost distance). Can either be (1) a single numeric value or (2) a SpatRaster where each pixel is the maximum distance to be used for that cell on the landscape (must be the same spatial scale as `lyr`).
distmat	distance matrix output from get_resdist (optional; can be used to save time on distance calculations)
stat	moving window statistic to calculate (see details). `stat` can generally be set to any function that will take `x`as input and return a single numeric value (for example, `x` can be a vector and `stat` can be set equal to a summary statistic like `mean`, `sum`, or `sd`)
fact	aggregation factor to apply to `lyr` (defaults to 0; *note:* increasing this value reduces computational time)
rarify	if rarify = TRUE, rarefaction is performed (defaults to FALSE)
rarify_n	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
rarify_nit	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to `"all"` to use all possible combinations of samples of size `rarify_n` within the window.
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
fun	function to use to summarize rarefaction results (defaults to mean, must take `na.rm = TRUE` as an argument)
L	for calculating `"pi"`, L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)
rarify_alleles	for calculating `"biallelic_richness"`, whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)
sig	for calculating `"hwe"`, significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)
transitionFunction	function to calculate transition values from grid values (defaults to mean)
directions	directions in which cells are connected (4, 8, 16, or other), see adjacent (defaults to 8)
geoCorrection	whether to apply correction to account for local distances (defaults to TRUE). Geographic correction is necessary for all objects of the class Transition that are either: (1) based on a grid in a geographic (lonlat) projection and covering a large area; (2) made with directions > 4 (see geoCorrection for more details).

... if a function is provided for stat, additional arguments to pass to the stat function (e.g. if stat = mean, users may want to set na.rm = TRUE)

Details

To calculate genetic diversity statistics with the built in wingen functions, data must be formatted as such:

- for "pi" or "biallelic_richness", x must be a dosage matrix with values of 0, 1, or 2
- for "Ho", x must be a heterozygosity matrix where values of 0 = homozygosity and values of 1 = heterozygosity
- for "allelic_richness" or "hwe, x must be a genind type object
- for "basic_stats", x must be a hierfstat type object

Otherwise, stat can be any function that takes a matrix or data frame and outputs a single numeric value (e.g., a function that produces a custom diversity index); however, this should be attempted with caution since this functionality has not have been tested extensively and may produce errors.

Value

SpatRaster that includes a raster layer of genetic diversity and a raster layer of the number of samples within the window for each cell

---

vcf_to_dosage *Convert a vcf to a dosage matrix*

Description

Convert a vcf to a dosage matrix

Usage

```
vcf_to_dosage(x)
```

Arguments

x can either be an object of class 'vcfR' or a path to a .vcf file

Value

dosage matrix

`window_gd` *Create a moving window map of genetic diversity*

Description

Generate a continuous raster map of genetic diversity using moving windows.

Usage

```
window_gd(
  gen,
  coords,
  lyr,
  stat = "pi",
  wdim = 3,
  fact = 0,
  rarify = FALSE,
  rarify_n = NULL,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05,
  crop_edges = FALSE,
  ...
)
```

Arguments

`gen`	genetic data either as an object of type vcf or a path to a vcf file (*note:* order matters! The coordinate and genetic data should be in the same order; there are currently no checks for this)
`coords`	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
`lyr`	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections)
`stat`	genetic diversity statistic(s) to calculate (see Details, defaults to `"pi"`). Can be a single statistic or a vector of statistics
`wdim`	dimensions (height x width) of window; if only one value is provided, a square window is created (defaults to 3 x 3 window)
`fact`	aggregation factor to apply to `lyr` (defaults to 0; *note:* increasing this value reduces computational time)
`rarify`	if rarify = TRUE, rarefaction is performed (defaults to FALSE)

rarify_n	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
rarify_nit	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to "all" to use all possible combinations of samples of size rarify_n within the window.
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
fun	function to use to summarize rarefaction results (defaults to mean, must take na.rm = TRUE as an argument)
L	for calculating "pi", L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)
rarify_alleles	for calculating "biallelic_richness", whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)
sig	for calculating "hwe", significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)
crop_edges	whether to remove cells on the edge of the raster where the window is incomplete (defaults to FALSE)
...	deprecated this was intended to be used to pass additional arguments to the stat function, however now formal arguments are used instead (see L, rarify_alleles, and sig). Passing additional arguments using ... is still possible with the *_general() functions.

Details

Coordinates and rasters should be in a projected (planar) coordinate system such that raster cells are of equal sizes. Therefore, spherical systems (including latitute-longitude coordinate systems) should be projected prior to use. Transformation can be performed using st_set_crs for coordinates or project for rasters (see vignette for more details).

Current genetic diversity metrics that can be specified with stat include:

- "pi" for nucleotide diversity (default) calculated using hierfstat pi.dosage. Use the L argument to set the sequence length (defaults to dividing by the number of variants).
- "Ho" for average observed heterozygosity across all sites
- "allelic_richness" for average number of alleles across all sites
- "biallelic_richness" for average allelic richness across all sites for a biallelic dataset (this option is faster than "allelic_richness")
- "hwe" for the proportion of sites that are not in Hardy–Weinberg equilibrium, calculated using pegas hw.test at the 0.05 level (other alpha levels can be specified by adding the sig argument; e.g., sig = 0.10).
- "basic_stats" for a series of statistics produced by hierfstat basic.stats including mean observed heterozygosity (same as Ho), mean gene diversities within population (Hs), Gene diversities overall (Ht), and Fis following Nei (1987). Population-based statistics (e.g., FST) normally reported by basic.stats are not included as they are not meaningful within the individual-based moving windows.

Value

SpatRaster that includes raster layers of genetic diversity and a raster layer of the number of samples within the window for each cell

Examples

```
load_mini_ex()
wpi <- window_gd(mini_vcf, mini_coords, mini_lyr, rarify = TRUE)
```

`window_general` *General function for making moving window maps*

Description

Generate a continuous raster map using moving windows. While window_gd is built specifically for making moving window maps of genetic diversity from vcfs, `window_general` can be used to make moving window maps from different data inputs. See details for how to format data inputs for different statistics.

Usage

```
window_general(
  x,
  coords,
  lyr,
  stat,
  wdim = 3,
  fact = 0,
  rarify = FALSE,
  rarify_n = NULL,
  rarify_nit = 5,
  min_n = 2,
  fun = mean,
  L = "nvariants",
  rarify_alleles = TRUE,
  sig = 0.05,
  crop_edges = FALSE,
  ...
)
```

Arguments

`x`	data to be summarized by the moving window (*note:* order matters! `coords` should be in the same order, there are currently no checks for this). The class of `x` required depends on the statistic being calculated (see the `stat` argument and the function description for more details)

coords	coordinates of samples as sf points, a two-column matrix, or a data.frame representing x and y coordinates (see Details for important information about projections)
lyr	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections)
stat	moving window statistic to calculate (can either be "pi" for nucleotide diversity (x must be a dosage matrix), "Ho" for average observed heterozygosity across all loci (x must be a heterozygosity matrix) , "allelic_richness" for average allelic richness across all loci (x must be a genind type object), "biallelic_richness" to get average allelic richness across all loci for a biallelic dataset (x must be a dosage matrix). stat can also be set to any function that will take xas input and return a single numeric value (for example, x can be a vector and stat can be set equal to a summary statistic like mean, sum, or sd)
wdim	dimensions (height x width) of window; if only one value is provided, a square window is created (defaults to 3 x 3 window)
fact	aggregation factor to apply to lyr (defaults to 0; *note:* increasing this value reduces computational time)
rarify	if rarify = TRUE, rarefaction is performed (defaults to FALSE)
rarify_n	if rarify = TRUE, number of points to use for rarefaction (defaults to min_n)
rarify_nit	if rarify = TRUE, number of iterations to use for rarefaction (defaults to 5). Can also be set to "all" to use all possible combinations of samples of size rarify_n within the window.
min_n	minimum number of samples to use in calculations (any focal cell with a window containing less than this number of samples will be assigned a value of NA; defaults to 2)
fun	function to use to summarize rarefaction results (defaults to mean, must take na.rm = TRUE as an argument)
L	for calculating "pi", L argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length L of the sequence. The wingen default is L = "nvariants", which sets L to the number of variants in the VCF. If L = NULL, returns the sum over SNPs of nucleotide diversity (*note:* L = NULL is the pi.dosage default which wingen does not use)
rarify_alleles	for calculating "biallelic_richness", whether to perform rarefaction of allele counts as in allelic.richness (defaults to TRUE)
sig	for calculating "hwe", significance threshold (i.e., alpha level) to use for hardy-weinberg equilibrium tests (defaults to 0.05)
crop_edges	whether to remove cells on the edge of the raster where the window is incomplete (defaults to FALSE)
...	if a function is provided for stat, additional arguments to pass to the stat function (e.g. if stat = mean, users may want to set na.rm = TRUE)

Details

To calculate genetic diversity statistics with the built in wingen functions, data must be formatted as such:

- for "pi" or "biallelic_richness", x must be a dosage matrix with values of 0, 1, or 2
- for "Ho", x must be a heterozygosity matrix where values of 0 = homozygosity and values of 1 = heterozygosity
- for "allelic_richness" or "hwe, x must be a genind type object
- for "basic_stats", x must be a hierfstat type object

Otherwise, stat can be any function that takes a matrix or data frame and outputs a single numeric value (e.g., a function that produces a custom diversity index); however, this should be attempted with caution since this functionality has not have been tested extensively and may produce errors.

Value

SpatRaster that includes a raster layer of genetic diversity and a raster layer of the number of samples within the window for each cell

Index