

Package ‘wingen’

August 19, 2025

Title Continuous Mapping of Genetic Diversity

Version 2.2.0

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](https://doi.org/10.1111/2041-210X.14090)>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

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

Suggests adegenet, automap, 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 (>= 4.1.0)

NeedsCompilation no

Author Anusha Bishop [aut, cre] (ORCID: <<https://orcid.org/0000-0003-1731-8683>>),
Anne Chambers [aut] (ORCID: <<https://orcid.org/0000-0002-7369-0108>>),
Ian Wang [aut] (ORCID: <<https://orcid.org/0000-0003-2554-9414>>)

Maintainer Anusha Bishop <anusha.bishop@berkeley.edu>

Repository CRAN

Date/Publication 2025-08-18 23:50:09 UTC

Contents

circle_gd	2
circle_general	5
coords_to_raster	7
get_geodist	8
get_resdist	8
ggplot_count	9
ggplot_gd	10
krig_gd	11
load_middle_earth_ex	13
load_mini_ex	13
lotr_coords	14
lotr_lyr	14
lotr_range	15
lotr_vcf	15
mask_gd	16
mini_coords	16
mini_lyr	17
mini_vcf	17
mini_vcf_NA	18
plot_count	18
plot_gd	19
preview_gd	20
resist_gd	22
resist_general	24
vcf_to_dosage	27
window_gd	27
window_general	29
wkrig_gd	32
Index	35

circle_gd	<i>Create a moving window map of genetic diversity using a circle window</i>
-----------	--

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 <code>vcf</code> or a path to a <code>vcf</code> file (<i>note</i> : 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 <code>sf</code> points, a two-column matrix, or a <code>data.frame</code> representing <code>x</code> and <code>y</code> coordinates (see Details for important information about projections).
lyr	<code>SpatRaster</code> or <code>RasterLayer</code> 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 <code>SpatRaster</code> where each pixel is the maximum distance to be used for that cell on the landscape (must be the same spatial scale as <code>lyr</code>).
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 <code>"pi"</code>). Can be a single statistic or a vector of statistics.
fact	Aggregation factor to apply to <code>lyr</code> (defaults to 0; <i>note</i> : increasing this value reduces computational time).
rarify	If <code>rarify = TRUE</code> , rarefaction is performed (defaults to <code>FALSE</code>).
rarify_n	If <code>rarify = TRUE</code> , number of points to use for rarefaction (defaults to <code>min_n</code>).
rarify_nit	If <code>rarify = TRUE</code> , number of iterations to use for rarefaction (defaults to 5). Can also be set to <code>"all"</code> to use all possible combinations of samples of size <code>rarify_n</code> 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 <code>NA</code> ; defaults to 2).
fun	Function to use to summarize rarefaction results (defaults to <code>mean</code> , must take <code>na.rm = TRUE</code> 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 (<i>note</i> : L = NULL is the pi.dosage default which wingen does not use).
rarefy_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 latitude-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

<code>x</code>	Data to be summarized by the moving window (<i>note:</i> order matters! <code>coords</code> should be in the same order, there are currently no checks for this). The class of <code>x</code> required depends on the statistic being calculated (see the <code>stat</code> argument and the function description for more details).
<code>coords</code>	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).
<code>lyr</code>	SpatRaster or RasterLayer to slide the window across (see Details for important information about projections).
<code>maxdist</code>	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 <code>lyr</code>).
<code>distmat</code>	Distance matrix output from get_geodist (optional; can be used to save time on distance calculations).
<code>stat</code>	Moving window statistic to calculate (see details). <code>stat</code> can generally be set to any function that will take <code>x</code> as input and return a single numeric value (for example, <code>x</code> can be a vector and <code>stat</code> can be set equal to a summary statistic like <code>mean</code> , <code>sum</code> , or <code>sd</code>).
<code>fact</code>	Aggregation factor to apply to <code>lyr</code> (defaults to 0; <i>note</i> : increasing this value reduces computational time).
<code>rarify</code>	If <code>rarify = TRUE</code> , rarefaction is performed (defaults to <code>FALSE</code>).
<code>rarify_n</code>	If <code>rarify = TRUE</code> , number of points to use for rarefaction (defaults to <code>min_n</code>).
<code>rarify_nit</code>	If <code>rarify = TRUE</code> , 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 <code>rarify_n</code> within the window.
<code>min_n</code>	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).
<code>fun</code>	Function to use to summarize rarefaction results (defaults to <code>mean</code> , must take <code>na.rm = TRUE</code> as an argument).
<code>L</code>	For calculating "pi", <code>L</code> argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length <code>L</code> of the sequence. The wingen default is <code>L = "nvariants"</code> , which sets <code>L</code> to the number of variants in the VCF. If <code>L = NULL</code> , returns the sum over SNPs of nucleotide diversity (<i>note</i> : <code>L = NULL</code> is the pi.dosage default which wingen does not use).
<code>rarify_alleles</code>	For calculating "biallelic_richness", whether to perform rarefaction of allele counts as in allelic.richness (defaults to <code>TRUE</code>).
<code>sig</code>	For calculating "hwe", significance threshold (i.e., alpha level) to use for Hardy-Weinberg equilibrium tests (defaults to 0.05).
<code>...</code>	If a function is provided for <code>stat</code> , additional arguments to pass to the <code>stat</code> function (e.g. if <code>stat = mean</code> , users may want to set <code>na.rm = TRUE</code>).

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	<i>Create a raster from coordinates</i>
------------------	---

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

<code>coords</code>	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).
<code>lyr</code>	SpatRaster or RasterLayer for generating distances (not required if <code>coords_only = TRUE</code>).
<code>fact</code>	Aggregation factor to apply to <code>lyr</code> (defaults to 0; <i>note</i> : increasing this value reduces computational time).
<code>coords_only</code>	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; <i>note</i> : 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

*Superseded: Krige moving window maps***Description**

This function has been superseded by [wkrig_gd](#) and may be removed in a future release. Please use `wkrig_gd()` instead, which provides improved performance and does not depend on the automap package.

Performs spatial interpolation (kriging) of the raster(s) produced by [window_gd](#) using the `autoKrige` function from `automap`.

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

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

disagg_r	Factor to use for disaggregation of <i>r</i> , 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 = $\sim x + y$).
resample	Whether to resample <i>grd</i> or <i>r</i> . Set to "r" to resample <i>r</i> to <i>grd</i> . Set to "grd" to resample <i>grd</i> to <i>r</i> (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).

Details

[Superseded]

Value

A SpatRaster object (if autoKrige_output = FALSE) or a list of autoKrige outputs.

See Also

[wkrig_gd](#) for the updated kriging function.

Examples

```
## Not run:
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")

## End(Not run)
```

load_middle_earth_ex	<i>Middle earth example</i>
----------------------	-----------------------------

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	<i>Mini middle earth example</i>
--------------	----------------------------------

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	<i>Middle earth example coordinates</i>
-------------	---

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	<i>Middle earth example raster</i>
----------	------------------------------------

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. [doi:10.21220/RKEZX707](https://doi.org/10.21220/RKEZX707)

lotr_range	<i>Middle earth example range polygon</i>
------------	---

Description

sf polygon of range map

Usage

lotr_range

Format

sf.

Source

Created from simulations in Bishop et al. (2023).

lotr_vcf	<i>Middle earth example vcf</i>
----------	---------------------------------

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	<i>Mask moving window maps</i>
---------	--------------------------------

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	<i>Mini middle earth example coordinates</i>
-------------	--

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) based on Rose, Robert A. (2020) GIS & Middle Earth Presentation & Data Set. William & Mary. doi:[10.21220/RKEZX707](https://doi.org/10.21220/RKEZX707)

`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"). You can also use sub="" for a subtitle. See title
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_1yr")
plot_count(mini_1yr)
```

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 <code>cex.main</code> , <code>font.main</code> , <code>col.main</code> to change the appearance; and <code>loc.main</code> to change the location of the main title (either two coordinates, or a character value such as "topleft"). You can also use <code>sub=""</code> for a subtitle. See title
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 <code>plot("SpatRaster", "numeric")</code> 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; <i>note</i> : 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 latitude-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	<i>Create a moving window map of genetic diversity based on resistance</i>
-----------	--

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 (<i>note:</i> 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; <i>note</i> : 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 (<i>note</i> : 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 latitude-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 (<i>note:</i> 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; <i>note:</i> 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 (<i>note</i> : 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	<i>Convert a vcf to a dosage matrix</i>
---------------	---

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	<i>Create a moving window map of genetic diversity</i>
-----------	--

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",
```

```

    rarefy_alleles = TRUE,
    sig = 0.05,
    crop_edges = FALSE,
    ...
)

```

Arguments

gen	Genetic data either as an object of type <code>vcf</code> or a path to a <code>vcf</code> file (<i>note</i> : 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 <code>sf</code> points, a two-column matrix, or a <code>data.frame</code> representing <code>x</code> and <code>y</code> coordinates (see Details for important information about projections).
lyr	<code>SpatRaster</code> or <code>RasterLayer</code> to slide the window across (see Details for important information about projections).
stat	Genetic diversity statistic(s) to calculate (see Details, defaults to <code>"pi"</code>). 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 <code>lyr</code> (defaults to 0; <i>note</i> : increasing this value reduces computational time).
rarefy	If <code>rarefy = TRUE</code> , rarefaction is performed (defaults to <code>FALSE</code>).
rarefy_n	If <code>rarefy = TRUE</code> , number of points to use for rarefaction (defaults to <code>min_n</code>).
rarefy_nit	If <code>rarefy = TRUE</code> , number of iterations to use for rarefaction (defaults to 5). Can also be set to <code>"all"</code> to use all possible combinations of samples of size <code>rarefy_n</code> 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 <code>NA</code> ; defaults to 2).
fun	Function to use to summarize rarefaction results (defaults to <code>mean</code> , must take <code>na.rm = TRUE</code> as an argument).
L	For calculating <code>"pi"</code> , <code>L</code> argument in pi.dosage function. Return the average nucleotide diversity per nucleotide given the length <code>L</code> of the sequence. The <code>wingen</code> default is <code>L = "nvariants"</code> , which sets <code>L</code> to the number of variants in the VCF. If <code>L = NULL</code> , returns the sum over SNPs of nucleotide diversity (<i>note</i> : <code>L = NULL</code> is the pi.dosage default which <code>wingen</code> does not use).
rarefy_alleles	For calculating <code>"biallelic_richness"</code> , whether to perform rarefaction of allele counts as in allelic.richness (defaults to <code>TRUE</code>).
sig	For calculating <code>"hwe"</code> , 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 <code>FALSE</code>).

... [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 latitude-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 (<i>note</i> : 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 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).
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; <i>note</i> : 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).

rarity_nit	If rarity = 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 rarity_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 (<i>note</i> : L = NULL is the pi.dosage default which wingen does not use).
rarity_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

wkrig_gd

*Krige moving window maps with variogram selection***Description**

Perform ordinary kriging of the raster(s) produced by [window_gd](#) using the gstat package to fit variograms and perform model selection. This function replaces the older [krig_gd](#) function to provide more flexibility in variogram model selection. While the default parameters have not been formally validated, they have performed well in practice for kriging wigen outputs from both simulated and empirical datasets.

Usage

```
wkrig_gd(
  r,
  grd = NULL,
  weight_r = NULL,
  models = c("Sph", "Exp", "Gau", "Mat"),
  nmax = Inf,
  maxdist = Inf,
  psill_start = NULL,
  nugget_start = NULL,
  range_start = NULL,
  max_range_frac = 0.5,
  fit_method = 6,
  model_output = FALSE
)
```

Arguments

<code>r</code>	SpatRaster produced by window_gd . Only the first layer is used if multiple layers are present.
<code>grd</code>	Object to create grid for kriging; can be a SpatRaster or RasterLayer. If undefined, <code>r</code> is used to create a grid.
<code>weight_r</code>	Optional SpatRaster with sample counts per cell, used to compute location-specific measurement variance and weights for kriging. If NULL (default), no weighting is applied.
<code>models</code>	Character vector of variogram model names to try (default: <code>c("Sph", "Exp", "Gau", "Mat")</code>).
<code>nmax</code>	Integer. Maximum number of neighboring observations to use for kriging at each prediction location (default: <code>Inf</code>). Users are encouraged to experiment with <code>nmax</code> to balance smoothness and local detail; starting with a value of 30 is recommended to reduce computational cost while still capturing local variability.
<code>maxdist</code>	Maximum distance to consider for neighboring observations (default: <code>Inf</code>). If set together with <code>nmax</code> , both parameters limit the number of neighbors.

psill_start	Optional starting value for partial sill. If NULL (default), a heuristic value is used (see Note).
nugget_start	Optional starting value for nugget effect. If NULL (default), a heuristic value is used (see Note).
range_start	Optional starting value for range parameter. If NULL (default), a heuristic value is used (see Note).
max_range_frac	Numeric. Maximum fraction of the range parameter to consider for neighboring observations (default: 0.5). This can help to limit the influence of distant points.
fit_method	Integer. Variogram fitting method passed to fit.variogram : 1 = weights N_j ; 2 = weights $N_j / \gamma(h_j)^2$; 6 = Ordinary Least Squares (unweighted); 7 = weights N_j / h_j^2 . The default (6) uses OLS, which is generally more robust for small or noisy datasets (see Note).
model_output	Logical. If TRUE, returns a list with the prediction raster, variogram, and fitted variogram model. If FALSE (default), returns only the prediction raster.

Details

The function fits multiple variogram models (Spherical, Exponential, Gaussian, Matern by default) and selects the best fit based on SSErr. It also includes optional weighting to account for sample count variation.

By default, starting values for variogram parameters are set heuristically:

- partial sill = ~80% of global variance
- nugget = ~20% of global variance
- range = 50% of maximum pairwise distance

The variogram fitting method defaults to Ordinary Least Squares (`fit_method = 6`), which tends to produce more stable fits in noisy or irregular datasets. Weighted methods (1, 2, and 7) may offer improved accuracy in large, well-distributed datasets.

For more fine-scale control over variogram fitting and kriging, consult the `gstat` package documentation.

This function uses [gstat](#) for variogram fitting and kriging. Weights are computed as the inverse of estimated location-specific variance (σ^2/n) if sample counts are provided.

Value

A [SpatRaster](#) object of kriged predictions if `model_output = FALSE`. If `model_output = TRUE`, returns a list with:

raster Kriged prediction raster ([SpatRaster](#))

variogram Empirical variogram ([variogram](#))

model Best-fit variogram model ([vgm](#))

Note**Convergence warnings from `gstat::fit.variogram()`**

During variogram fitting, you may see:

Warning: No convergence after 200 iterations: try different initial values?

This means the optimizer reached its iteration limit before fully minimizing the error. Even in these cases, `gstat` returns the best-fit model found so far. Warnings often occur with small datasets or noisy empirical variograms. You can experiment with different `psill_start`, `range_start`, and `nugget_start` values, or increase the iteration limit using `options(gstat.fit.maxiter)`.

Examples

```
# Note: this toy example uses a very small dataset.
# Warnings may occur due to limited points for variogram fitting.
suppressWarnings({
  load_mini_ex()
  wpi <- window_gd(mini_vcf, mini_coords, mini_lyr, L = 10, rarify = TRUE)
  kpi <- wkrig_gd(wpi[["pi"]], grd = mini_lyr, nugget = 0)
  plot_gd(kpi, main = "Kriged Pi")
})
```

Index

* datasets

- lotr_coords, [14](#)
- lotr_lyr, [14](#)
- lotr_range, [15](#)
- lotr_vcf, [15](#)
- mini_coords, [16](#)
- mini_lyr, [17](#)
- mini_vcf, [17](#)
- mini_vcf_NA, [18](#)

adjacent, [9](#), [23](#), [26](#)

allelic.richness, [4](#), [6](#), [23](#), [26](#), [28](#), [31](#)

basic.stats, [4](#), [24](#), [29](#)

circle_gd, [2](#), [8](#), [20](#), [21](#)

circle_general, [5](#)

coords_to_raster, [7](#)

deprecated, [29](#)

fit.variogram, [33](#)

geoCorrection, [9](#), [23](#), [26](#)

get_geodist, [3](#), [6](#), [8](#), [21](#)

get_resdist, [8](#), [21](#), [23](#), [25](#)

ggplot_count, [9](#)

ggplot_gd, [10](#)

gstat, [33](#)

hw.test, [4](#), [24](#), [29](#)

krig_gd, [9](#), [10](#), [11](#), [16](#), [18–20](#), [32](#)

load_middle_earth_ex, [13](#)

load_mini_ex, [13](#)

lotr_coords, [14](#)

lotr_lyr, [14](#)

lotr_range, [15](#)

lotr_vcf, [15](#)

magma, [10](#), [20](#)

mask_gd, [16](#)

mini_coords, [16](#)

mini_lyr, [17](#)

mini_vcf, [17](#)

mini_vcf_NA, [18](#)

pi.dosage, [4](#), [6](#), [23](#), [24](#), [26](#), [28](#), [29](#), [31](#)

plot_count, [18](#)

plot_gd, [19](#)

preview_gd, [20](#)

project, [4](#), [21](#), [24](#), [29](#)

resist_gd, [5](#), [8](#), [9](#), [20](#), [21](#), [22](#), [24](#)

resist_general, [24](#)

SpatRaster, [32](#), [33](#)

st_distance, [8](#)

st_set_crs, [4](#), [21](#), [24](#), [29](#)

title, [19](#), [20](#)

variogram, [33](#)

vcf_to_dosage, [27](#)

vgm, [33](#)

window_gd, [7](#), [9–11](#), [16](#), [18–21](#), [27](#), [29](#), [32](#)

window_general, [29](#)

wkrig_gd, [11](#), [12](#), [32](#)