Package: dartR.spatial (via r-universe)

November 20, 2026

Type Package

Title Applying Landscape Genomic Methods on 'SNP' and 'Silicodart' Data

Version 1.3.0

Date 2026-11-18

Revision Hardy Hedgehog

Description Provides landscape genomic functions to analyse 'SNP' (single nuclear polymorphism) data, such as least cost path analysis and isolation by distance. Therefore each sample needs to have coordinate data attached (lat/lon) to be able to run most of the functions. 'dartR.spatial' is a package that belongs to the 'dartRverse' suit of packages and depends on 'dartR.base' and 'dartR.data'.

Encoding UTF-8

Depends R (>= 3.5), dartR.base (>= 1.3.0), dartR.data (>= 1.2.5)

Imports adegenet (>= 2.0.0), methods, utils, crayon, StAMPP, raster, sp, tidyr, vegan, MASS, ggplot2, data.table, grDevices, Matrix, igraph

Suggests mmod, testthat (>= 3.0.0), withr, gdistance, gplots, rrBLUP, terra (>= 1.7-39), sf, PopGenReport

License GPL (>= 3)

RoxygenNote 7.3.3

NeedsCompilation no

Author Bernd Gruber [aut, cre], Arthur Georges [aut], Jose L. Mijangos [aut], Carlo Pacioni [aut], Peter J. Unmack [ctb], Oliver Berry [ctb]

URL https://green-striped-gecko.github.io/dartR/

BugReports https://groups.google.com/g/dartr?pli=1

Maintainer Bernd Gruber <bernd.gruber@canberra.edu.au>

Config/pak/sysreqs zlib1g-dev

Repository https://cran.r-universe.dev
Date/Publication 2026-11-19 23:50:02 UTC
RemoteUrl https://github.com/cran/dartR.spatial
RemoteRef HEAD
RemoteSha b7f25215ccb291c52a53398748f1d7214e0e93f8

Contents

gl.costdistances 2
gl.genleastcost 4
gl.grm2 6
gl.ibd 9
gl.kosman 11
gl.run.eems 12
gl.spatial.autoCorr 16
gl2shp 20
utils.spautocor 21

Index **24**

`gl.costdistances` *Calculates distances across a resistance landscape*

Description

Calculates pairwise least-cost, randomised shortest-path or random-walk commute distances between locations on a resistance raster.

Usage

```
gl.costdistances(landscape, locs, method, NN, verbose = NULL, theta = 1)
```

Arguments

`landscape`	Single-layer RasterLayer of resistance values [required].
`locs`	Genlight with named lon/lat columns in @other$latlon, or a numeric two-column matrix/data.frame in x/y order in the raster CRS. Supplied row names must be unique. For individual distances from a genlight with existing populations, set pop(x) <- indNames(x) [required].
`method`	One of 'leastcost', 'rSPDistance' or 'commute' [required].
`NN`	Neighbourhood: 4, 8, 16 or 'bishop' (diagonals only). Eight neighbours is a common choice [required].
`verbose`	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].
`theta`	RSP parameter, strictly between 0 and 20. Used only when method is 'rSPDistance'; changing it changes the path model [default 1].

Details

Adjacent cells have conductance equal to the reciprocal of their mean resistance. Connections are symmetric. Geographic correction uses inter-cell distance, with type 'c' for least-cost and randomised shortest paths (RSP), and type 'r' for commute. For geographic rasters, distances use metres; projected rasters use their coordinate units. Explicit x/y coordinates on rasters with no CRS use local grid units; no projection or metre units are inferred.

Least-cost and RSP distances combine resistance and distance units. Commute is expected random-walk travel time (steps there and back), equal to effective resistance multiplied by graph volume (the sum of conductance over both directions). It is not the unscaled effective resistance used by Circuitscape. RSP retains gdistance's default net-movement method. Theta controls the path model and interacts with cost scale. If theta underflows on large costs, the error names a smaller theta that gives finite distances; theta is not changed automatically. Type 'c' is the RSP correction convention here; gdistance has no unique geographic correction for intermediate RSP regimes.

NA and positive Inf raster values are barriers. Traversable cells must have strictly positive, finite resistance. Locations on barriers or outside the raster are rejected. Disconnected least-cost pairs return Inf; commute and RSP require a connected traversable graph. A single location, or locations in the same cell, have zero pairwise distance.

Genlight coordinates are interpreted as WGS84 longitude/latitude. Arithmetic population centres are calculated in those coordinates, then transformed to the raster CRS. Stored coordinate rows must follow individual order. With no population assignments, individuals are used instead. Missing assignments or coordinates are rejected; no observations are silently omitted.

Value

A numeric square matrix of pairwise distances, with location or population names on both axes when supplied. Input objects are unchanged.

See Also

costDistance, rSPDistance, commuteDistance, geoCorrection

Other spatial analysis functions: `gl.genleastcost()`, `gl.spatial.autoCorr()`

Examples

```
if (requireNamespace("gdistance", quietly = TRUE)) {
  landscape.sim <- readRDS(system.file("extdata", "landscape.sim.rdata",
                                       package = "dartR.data"))
  # This synthetic landscape uses local grid coordinates, not a named CRS.
  xy <- apply(possums.gl@other$xy, 2, function(a)
    tapply(a, pop(possums.gl), mean))
  cd <- gl.costdistances(landscape.sim, xy, method = "leastcost", NN = 8)
  round(cd, 3)
}
```

`gl.genleastcost` *Performs least-cost path analysis based on a friction matrix*

Description

This function calculates pairwise distances (Euclidean, cost path distances and genetic distances) between populations or between individuals using a friction matrix and a spatial genlight object. The output of this function can be used with the functions wassermann and lgrMMRR from package PopGenReport to test for the significance of a layer on the genetic structure.

Usage

```
gl.genleastcost(
  x,
  fric.raster,
  gen.distance = "Gst.Nei",
  NN = 8,
  pathtype = "leastcost",
  plotpath = TRUE,
  theta = 1,
  plot.colors.pop = gl.colors("dis", verbose = 0),
  raster.colors = rev(terrain.colors(255)),
  verbose = NULL
)
```

Arguments

`x`	Name of the genlight object containing SNP data and coordinates [required].
`fric.raster`	A friction matrix: file path, RasterLayer, RasterStack, RasterBrick or SpatRaster [required].
`gen.distance`	Genetic distance between populations ('D', 'Gst.Nei', 'Gst.Hedrick') or individuals ('kosman', 'propShared', 'dist') [default "Gst.Nei"].
`NN`	Number of neighbours used when calculating the cost distance (possible values 4, 8, 16 or 'bishop'). NN=8 is most commonly used. Be aware that linear structures may cause artefacts in the least-cost paths, therefore inspect the actual least-cost paths in the provided output [default 8].
`pathtype`	Type of cost distance to be calculated: 'leastcost', 'commute' or 'rSPDistance'. If the path type is set to 'leastcost' and plotpath is TRUE, paths and path lengths are also returned [default 'leastcost'].
`plotpath`	If TRUE, plots each friction layer with the individuals, population centres and (for 'leastcost') the least-cost paths, and returns the paths and their lengths. Calculating paths slows down the computation, but checking them visually is recommended [default TRUE].
`theta`	Value needed for pathtype 'rSPDistance', strictly between 0 and 20. See `gl.costdistances` [default 1].

`plot.colors.pop` A color palette for population plots or a list with as many colors as there are populations in the dataset [default gl.colors("dis")].

`raster.colors` The color palette to use to color the raster values [default rev(terrain.colors(255))].

`verbose` Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].

Details

The friction matrix can be a single raster or a stack of several layers, supplied as a file path, a RasterLayer, RasterStack or RasterBrick (package raster) or a SpatRaster (package terra). The specified cost distance is calculated for each layer.

Cost distances are calculated with `gl.costdistances`, so both functions return the same values for the same landscape and locations. See that function for how resistance, barriers and geographic correction are handled.

Coordinates are taken from x@other$xy, which must be in the same coordinate system as the friction matrix. If x@other$xy is missing, the lon and lat columns of x@other$latlon are used as they are, without projection, so they must also match the coordinate system of the friction matrix. For population distances, the arithmetic mean of the coordinates of each population is used.

Genetic distances between populations are 'D', 'Gst.Nei' and 'Gst.Hedrick' (package mmod). Genetic distances between individuals are 'kosman' (`gl.kosman`), 'propShared' (one minus the proportion of shared alleles, `gl.propShared`) and 'dist' (Euclidean distance between allele counts).

Value

A list with six elements:

- gen.mat – matrix of pairwise genetic distances.
- eucl.mat – matrix of pairwise Euclidean distances.
- cost.matnames – names of the friction layers.
- cost.mats – list of cost distance matrices, one per layer.
- pathlength.mats – list of path length matrices, one per layer (only for pathtype 'leastcost' with plotpath = TRUE).
- paths – list of least-cost paths as SpatialLines objects, one list per layer (only for pathtype 'leastcost' with plotpath = TRUE).

Author(s)

Author(s): Bernd Gruber. Custodian: Bernd Gruber – Post to https://groups.google.com/d/forum/dartr

References

- Cushman, S., Wasserman, T., Landguth, E. and Shirk, A. (2013). Re-Evaluating Causal Modeling with Mantel Tests in Landscape Genetics. Diversity, 5(1), 51-72.

- Landguth, E. L., Cushman, S. A., Schwartz, M. K., McKelvey, K. S., Murphy, M. and Luikart, G. (2010). Quantifying the lag time to detect barriers in landscape genetics. Molecular ecology, 4179-4191.
- Wasserman, T. N., Cushman, S. A., Schwartz, M. K. and Wallin, D. O. (2010). Spatial scaling and multi-model inference in landscape genetics: Martes americana in northern Idaho. Landscape Ecology, 25(10), 1601-1612.

See Also

gl.costdistances

Other spatial analysis functions: gl.costdistances(), gl.spatial.autoCorr()

Examples

```
if (requireNamespace("gdistance", quietly = TRUE) &&
    requireNamespace("mmod", quietly = TRUE)) {
  landscape.sim <- readRDS(system.file("extdata", "landscape.sim.rdata",
                                       package = "dartR.data"))
  # three populations and a coarser landscape to keep the example fast
  x <- possums.gl[pop(possums.gl) %in% c("A", "B", "C"), ]
  landscape <- raster::aggregate(landscape.sim, 5)
  glc <- gl.genleastcost(x, fric.raster = landscape, gen.distance = "D",
                         NN = 8, pathtype = "leastcost", plotpath = TRUE)
  glc$cost.mats
  # test the significance of the layer (package PopGenReport)
  if (requireNamespace("PopGenReport", quietly = TRUE)) {
    PopGenReport::wassermann(eucl.mat = glc$eucl.mat,
                             cost.mat = glc$cost.mats,
                             gen.mat = glc$gen.mat)
  }
}
```

---

gl.grm2 *Calculates an identity by descent matrix*

Description

This function is the dartR.spatial copy of gl.grm (package dartR.captive), kept so that spatial analyses do not depend on dartR.captive. Both return the same matrix.

This function calculates the mean probability of identity by state (IBS) across loci that would result from all the possible crosses of the individuals analyzed. IBD is calculated by an additive relationship matrix approach developed by Endelman and Jannink (2012) as implemented in the function A.mat (package rrBLUP).

Usage

```
gl.grm2(
  x,
  plotheatmap = TRUE,
  palette_discrete = NULL,
  palette_convergent = NULL,
  legendx = 0,
  legendy = 0.5,
  label.size = 0.75,
  legend.title = "Populations",
  plot.file = NULL,
  plot.dir = NULL,
  verbose = NULL,
  ...
)
```

Arguments

x	Name of the genlight object containing the SNP data [required].
plotheatmap	A switch if a heatmap should be shown [default TRUE].
palette_discrete	the color of populations [gl.select.colors].
palette_convergent	A convergent palette for the IBD values or vector of colours [default RdYlBu from gl.select.colors].
legendx	x coordinates for the legend[default 0].
legendy	y coordinates for the legend[default 0.5].
label.size	Specify the size of the population labels [default 0.75].
legend.title	Legend title [default "Populations"].
plot.file	Name for the RDS binary file to save (base name only, exclude extension); saved only when plotheatmap = TRUE [default NULL]
plot.dir	Directory in which to save files [default = working directory]
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, progress log ; 3, progress and results summary; 5, full report [default 2 or as specified using gl.set.verbosity].
...	Parameters passed to function A.mat from package rrBLUP. Unless min.MAF is given, it is set to 1/(2n) - 1e-10, so loci with a single copy of the minor allele are kept on every platform.

Details

This function uses the A.mat function from the rrBLUP package. This method follows the approach developed by Endelman and Jannink (2012).

Two alleles are Identical by State (IBS) if they are the same in state, regardless of whether they come from a common ancestor. Two alleles are Identical by Descent (IBD) if they are inherited

from a common ancestor. While IBS does not necessarily imply IBD, using high-density SNP data improves the estimation of IBD probabilities from IBS measures.

This function also plots a heatmap, and a dendrogram, of IBD values where each diagonal element has a mean that equals 1+f, where f is the inbreeding coefficient (i.e. the probability that the two alleles at a randomly chosen locus are IBD from the base population). As this probability lies between 0 and 1, the diagonal elements range from 1 to 2. Because the inbreeding coefficients are expressed relative to the current population, the mean of the off-diagonal elements is -(1+f)/n, where n is the number of loci. Individual names are shown in the margins of the heatmap and colors represent different populations.

A file-backed (FBM) object is not decoded as a whole for A.mat's default method (mean imputation, no shrinkage; `min.MAF` and `max.missing` may be given): the matrix is computed in blocks of loci (`dartR.base::utils.fbm.grm`) and agrees with A.mat to about 1e-13. Other A.mat options (`shrink`, `impute.method = "EM"`, `return.imputed`) need the whole matrix: the object is then converted to a genlight, with a warning stating the memory A.mat uses (about four doubles per genotype). Gen-backed objects use A.mat as before.

Value

An identity by descent matrix

Author(s)

Author(s): Arthur Georges. Custodian: Arthur Georges – Post to https://groups.google.com/d/forum/dartr

References

- Endelman, J. B. (2011). Ridge regression and other kernels for genomic selection with r package rrblup. The Plant Genome 4, 250.
- Endelman, J. B. , Jannink, J.-L. (2012). Shrinkage estimation of the realized relationship matrix. G3: Genes, Genomics, Genetics 2, 1405.

See Also

`gl.grm` and `gl.grm.network` in package dartR.captive, which compute the same matrix.

Examples

```
if (isTRUE(getOption("dartR_fbm"))) platypus.gl <- gl.gen2fbm(platypus.gl)
gl.grm2(platypus.gl[1:10, 1:100])
```

gl.ibd *Performs isolation by distance analysis*

Description

This function performs an isolation by distance analysis based on a Mantel test and also produces an isolation by distance plot. If a genlight object with coordinates is provided, then geographic and genetic distance matrices are calculated.

Usage

```
gl.ibd(
  x = NULL,
  distance = "Fst",
  coordinates = "latlon",
  Dgen = NULL,
  Dgeo = NULL,
  Dgeo_trans = "Dgeo",
  Dgen_trans = "Dgen",
  permutations = 999,
  plot.out = TRUE,
  paircols = NULL,
  plot.theme = theme_dartR(),
  plot.file = NULL,
  plot.dir = NULL,
  verbose = NULL
)
```

Arguments

x	Genlight object used to calculate missing distances. Ignored when both Dgen and Dgeo are supplied [default NULL].
distance	Distance to calculate: population-based 'Fst' [stamppFst] or 'D' [stamppNeisD], or individual-based 'propShared' (1 - gl.propShared), 'euclidean' [dist] or 'kosman' [gl.kosman] [default "Fst"]. SilicoDArT data accept only 'euclidean' and 'kosman'.
coordinates	'latlon', 'xy', or a two-column data.frame named lat/lon or x/y. Stored coordinates in x@other$latlon or x@other$xy follow individual order. Explicit data.frames with row names are matched to indNames(x); automatic row names use positional order. All used coordinates must be numeric and finite. Longitude/latitude gives geodesic distances in metres (population centres are the mean longitude and latitude); x/y gives Euclidean distances in coordinate units [default "latlon"].
Dgen	Genetic distances as a dist object or symmetric numeric square matrix. If NULL, calculated from x [default NULL].

Dgeo	Geographic distances as a dist object or symmetric numeric square matrix. If NULL, calculated from coordinates [default NULL].
Dgeo_trans	R expression transforming Dgeo, for example 'log(Dgeo)'. The default leaves distances unchanged [default "Dgeo"].
Dgen_trans	R expression transforming Dgen, for example 'Dgen/(1-Dgen)'. The default leaves distances unchanged [default "Dgen"].
permutations	Number of permutations in the Mantel test [default 999].
plot.out	Display the isolation by distance plot [default TRUE].
paircols	Colour points by 'pop'ulation or 'ind'ividual pairs. For individual distances calculated using x, 'pop' uses population membership. With both distance matrices supplied, x is ignored and distance labels are used. NULL uses uncoloured points [default NULL].
plot.theme	ggplot2 theme for the plot [default theme_dartR()].
plot.file	Name for the RDS binary file to save (base name only, exclude extension) [default NULL]
plot.dir	Directory in which to save files [default = working directory]
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].

Details

Both transformations default to the identity: Fst is not linearised and geographic distance is not logged unless requested. The Euclidean genetic option uses stats::dist on the genotype matrix. For pairs with missing loci, squared differences are scaled by the total number of loci divided by the number compared; this differs from gl.dist.ind. Population geographic distances use the mean coordinates of each population.

Labelled distance inputs must have unique, matching identities. Geographic distances are reordered to match genetic distances. When either input lacks labels, its supplied order is used; callers must ensure correspondence. At least three individuals or populations, finite distances for every pair, and variation in both distance vectors are required after transformation. Missing pairs are rejected, not omitted from the Mantel test. Finite negative Fst estimates and transformed distances are permitted.

The Mantel test uses vegan::mantel. Terra is needed for geodesic distances. If an explicitly requested log(Dgeo) produces log(0), choose a scientifically appropriate transformation such as log(Dgeo + 1). No offset is added automatically. A plot is constructed only when plot.out is TRUE or plot.file is supplied. plot.file saves the plot as RDS even when plot.out is FALSE.

Value

A list with Dgen and Dgeo (the transformed, aligned dist objects) and mantel (the Mantel test result). The plot is displayed or saved separately.

Author(s)

Bernd Gruber (bugs? Post to https://groups.google.com/d/forum/dartr)

References

Rousset, F. (1997). Genetic differentiation and estimation of gene flow from F-statistics under isolation by distance. Genetics, 145(4), 1219-1228.

See Also

mantel, stamppFst

Examples

```
#because of speed only the first 100 loci
#' if (isTRUE(getOption("dartR_fbm"))) bandicoot.gl <- gl.gen2fbm(bandicoot.gl)
ibd <- gl.ibd(bandicoot.gl[,1:100], Dgeo_trans='log(Dgeo)' ,
Dgen_trans='Dgen/(1-Dgen)')
#because of speed only the first 10 individuals)
ibd <- gl.ibd(bandicoot.gl[1:10,], distance='euclidean', paircols='pop',
Dgeo_trans='Dgeo')

#only first 100 loci
ibd <- gl.ibd(bandicoot.gl[,1:100], paircols='pop')
```

---

gl.kosman *Calculates a Kosman distance matrix for each pair of individuals*

Description

This function calculates the Kosman and Leonard (2005) distance between each pair of individuals.

Usage

```
gl.kosman(x, verbose = NULL)
```

Arguments

x	Name of the genlight object containing the SNP or SilicoDArT data, with a single ploidy across individuals [required].
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].

Details

For each locus, the distance between two individuals is the absolute difference in allele dosage divided by the ploidy (0, 0.5 or 1 for diploid SNPs). The Kosman distance is the mean of these values over the loci called in both individuals, so missing data are handled pair by pair.

All individuals must have the same ploidy. SilicoDArT data (ploidy 1) are treated as haploid, so the distance is the proportion of shared called loci at which presence/absence differs.

Pairs of individuals with no called loci in common get a distance of NaN.

Value

A list with two nInd(x) x nInd(x) matrices, with individual names as row and column names. Only the lower triangle and the diagonal are filled; the upper triangle is NA.

- kosman – Kosman distances between individuals.
- nloci – number of loci called in both individuals, used for each distance; the diagonal holds the number of loci called in that individual.

Use `as.dist(result$kosman)` to obtain a dist object.

Author(s)

Author(s): Bernd Gruber. Custodian: Bernd Gruber – Post to https://groups.google.com/d/forum/dartr

References

- Kosman, E., & Leonard, K. J. (2005). Similarity coefficients for molecular markers in studies of genetic relationships between individuals for haploid, diploid, and polyploid species. Molecular Ecology, 14(2), 415-424.

Examples

```
#use only five individuals and seven loci
gg <- gl.kosman(possums.gl[1:5,14:21])
gg$kosman
gg$nloci
```

`gl.run.eems` *Runs the EEMS algorithm (Estimating Effective Migration Surfaces) on a genlight object.*

Description

This function runs the EEMS algorithm on a genlight object. The EEMS algorithm is a spatially explicit model that estimates effective migration surfaces (EEMS) from genetic data. The EEMS algorithm is implemented in C++, hence it is necessary to have the binary downloaded and the function needs to point to this file via the path specified in eems.path. The binary is call runeems_snps[.exe] and can be downloaded from the github site of dartRverse (https://github.com/green-striped-gecko/dartRverse/tree/main/binar

Usage

```
gl.run.eems(
  x,
  eems.path = "./",
  buffer = 10000,
  nDemes = 500,
```

```
  diploid = TRUE,
  numMCMCIter = 10000,
  numBurnIter = 2000,
  numThinIter = 9,
  seed = NULL,
  dpi = 250,
  add_grid = FALSE,
  col_grid = "#BBBBBB",
  add_demes = FALSE,
  col_demes = "#000000",
  add_outline = FALSE,
  col_outline = "#FFFFFF",
  eems_colors = NULL,
  plot.colors.pop = gl.colors("dis"),
  out.dir = NULL,
  plot.dir = NULL,
  plot.file = NULL,
  verbose = NULL,
  cleanup = TRUE,
  habitat = NULL,
  ...
)
```

Arguments

x	Name of the genlight object containing the SNP data [required].
eems.path	Path to the folder containing the eems executable [default working directory ("./")].
buffer	Distance in metres by which the convex hull of the sampling locations is expanded to form the habitat. Measured on the ground at any latitude. Ignored when habitat is supplied [default 10000].
nDemes	The approximate number of demes in the population graph [default 500].
diploid	Whether the organism is diploid. TRUE requires ploidy 2 for every individual; FALSE requires ploidy 1. Input must represent SNP allele dosages, not dominant presence/absence markers [default TRUE].
numMCMCIter	Number of MCMC iterations [default 10000].
numBurnIter	Number of burn-in iterations to discard at the start [default 2000].
numThinIter	Number of iterations to thin between two writing steps [default 9].
seed	An integer used to seed the random number generator [default NULL].
dpi	Resolution of the contour raster [default 250].
add_grid	A logical value indicating whether to add the population grid or not [default FALSE].
col_grid	The color of the population grid [default gray ("#BBBBBB")].
add_demes	A logical value indicating whether to add the observed demes or not [default FALSE].

col_demes	The color of the demes [default black ("#000000")].
add_outline	A logical value indicating whether to add the habitat outline or not [default FALSE].
col_outline	The color of the habitat outline [default white ("#FFFFFF")].
eems_colors	The EEMS color scheme as a vector of colors, ordered from low to high [default NULL].
plot.colors.pop	A color palette for population plots or a list with as many colors as there are populations in the dataset [default gl.colors("dis")].
out.dir	Existing directory for output. Each call creates a unique eems-run-* subdirectory containing data_eems/ and eems.log. Relative paths are resolved from the calling directory. Results in tempdir() last only for the R session [default tempdir()].
plot.dir	Directory to save the plot RDS files [default as specified by the global working directory or tempdir()].
plot.file	Name for the RDS binary file to save (base name only, exclude extension); NULL disables RDS saving [default NULL].
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, progress log; 3, progress and results summary; 5, full report [default NULL, unless specified using gl.set.verbosity].
cleanup	Whether to delete this run's input and parameter files after success. Raw EEMS results and logs are retained; failed runs retain all files for diagnosis [default TRUE].
habitat	Habitat outline for EEMS: an sf or sfc polygon in any coordinate reference system (transformed to longitude/latitude), or a two-column matrix of longitude and latitude in decimal degrees. It must be a single polygon without holes. NULL uses the convex hull of the sampling locations expanded by buffer [default NULL].
...	Extra parameters to add to function reemsplots2::make_eems_plots.

Details

Set the Number of MCMC iterations to 2 million for an initial run. If the posterior trace is still trending, lengthen the chain. Set the number of initial burnin iterations at 50 thin between writes so that total/thin is around 100–200; a 10000-step thinning interval is a practical default.

Choose the number of demes to match geographic scale: 100–250 for local or island studies and 300–500 for continental datasets. Run time grows cubically with number of demes, so anything above 1000 rarely pays off.

Coordinates are passed to EEMS as longitude/latitude with great-circle distance, so maps are returned in degrees and the rdist03 plot shows distances in kilometres.

By default the habitat is the convex hull of the sampling locations, expanded by buffer (in metres), so every sample sits at least one grid spacing inside the edge. A 5–10 (or a few kilometers for fine-scale work) is usually adequate. When the hull covers unsuitable areas, for example land between sampling sites of a marine species, supply the habitat outline with habitat. EEMS does not check that samples fall inside the habitat: each sample is assigned to the closest deme, and the function warns when samples fall outside.

The dpi argument controls sampling of the contour grid in reemsplots2, not the dots per inch of an exported image. Higher values increase grid resolution and plotting work without changing EEMS inference. Set image dimensions and export DPI separately when saving a rendered plot. Routine messages are suppressed at verbose = 0; diagnostic plots are displayed at verbose >= 3. All eight plots are returned at every level.

Value

A list of contour plots of migration and diversity rates as well as several diagnostic plots. It is a good idea to examine all these figures, which is why they are generated by default. Please check the examples how to customise the figures.

mrates01 Effective migration surface. This contour plot visualises the estimated effective migration rates `m`, on the log10 scale after mean centering.

mrates02 Posterior probability contours `P(log(m) > 0) = p` and `P(log(m) < 0) = p` for the given probability level `p`. Since migration rates are visualised on the log10 scale after mean centering, 0 corresponds to the overall mean migration rate. This contour plot emphasizes regions with effective migration that is significantly higher/lower than the overall average.

qrates01 Effective diversity surface. This contour plot visualises the estimated effective diversity rates `q`, on the log10 scale after mean centering.

qrates02 Posterior probability contours `P(log(q) > 0) = p` and `P(log(q) < 0) = p`. Similar to `mrates02` but applied to the effective diversity rates.

rdist01 Scatter plot of the observed vs the fitted between-deme component of genetic dissimilarity, where one point represents a pair of sampled demes.

rdist02 Scatter plot of the observed vs the fitted within-deme component of genetic dissimilarity, where one point represents a sampled deme.

rdist03 Scatter plot of observed genetic dissimilarities between demes vs observed geographic distances between demes.

pilogl01 Posterior probability trace

Author(s)

Bernd Gruber & Robyn (bugs? Post to https://groups.google.com/d/forum/dartr)

References

- Petkova D (2024). _reemsplots2: Generate plots to inspect and visualize the results of EEMS_. R package version 0.1.0, https://github.com/dipetkov/eems.
- D Petkova, J Novembre, M Stephens. Visualizing spatial population structure with estimated effective migration surfaces. Nature Genetics 48, 94 – 100 (2016). doi:10.1038/ng.3464.

Examples

```
## Not run:
# This example needs a binary (runeems_snps[.exe]) specific to your
# operating system  to run
eems <- gl.run.eems(bandicoot.gl, eems.path = "d:/downloads/eems/")
print(eems[[1]])
```

```
## End(Not run)
```

`gl.spatial.autoCorr` *Spatial autocorrelation following Smouse and Peakall 1999*

Description

Global spatial autocorrelation is a multivariate approach combining all loci into a single analysis. The autocorrelation coefficient "r" is calculated for each pair of individuals in each specified distance class. For more information see Smouse and Peakall 1999, Peakall et al. 2003 and Smouse et al. 2008.

Usage

```
gl.spatial.autoCorr(
  x = NULL,
  Dgeo = NULL,
  Dgen = NULL,
  coordinates = "latlon",
  Dgen_method = "Euclidean",
  Dgeo_trans = "Dgeo",
  Dgen_trans = "Dgen",
  bins = 5,
  reps = 100,
  plot.pops.together = FALSE,
  permutation = TRUE,
  bootstrap = TRUE,
  plot.theme = theme_dartR(),
  plot.colors.pop = NULL,
  CI.color = "red",
  plot.out = TRUE,
  plot.file = NULL,
  plot.dir = NULL,
  verbose = NULL
)
```

Arguments

`x`	Name of the genlight object containing the SNP or SilicoDArT data. If provided, Dgen and Dgeo are ignored [default NULL].
`Dgeo`	Geographic distance matrix, dist object or list of them (one per population, in the same order as Dgen; list names are used as population names) if no genlight object is provided. This is typically an Euclidean distance but it can be any meaningful (geographical) distance metrics [default NULL].

Dgen	Genetic distance matrix, dist object or list of them (one per population) if no genlight object is provided [default NULL].
coordinates	Can be either 'latlon', 'xy' or a two column data.frame with column names 'lat','lon', 'x', 'y') Coordinates are provided via gl@other$latlon ['latlon'] or via gl@other$xy ['xy']. For latlon, geodesic distances in metres are calculated; for xy, Euclidean distances are calculated on the coordinates [default "latlon"].
Dgen_method	Method to calculate genetic distances. See details [default "Euclidean"].
Dgeo_trans	Transformation to be used on the geographic distances. See Dgen_trans [default "Dgeo"].
Dgen_trans	You can provide a formula to transform the genetic distance. The transformation can be applied as a formula using Dgen as the variable to be transformed. For example: Dgen_trans = 'Dgen/(1-Dgen)'. Any valid R expression can be used here [default 'Dgen', which is the identity function.]
bins	The number of bins for the distance classes (i.e. length(bins) == 1) or a vectors with the break points. See details [default 5].
reps	The number to be used for permutation and bootstrap analyses [default 100].
plot.pops.together	Plot all the populations in one plot. Confidence intervals from permutations are not shown [default FALSE].
permutation	Whether permutation calculations for the null hypothesis of no spatial structure should be carried out [default TRUE].
bootstrap	Whether bootstrap calculations to compute the 95% confidence intervals around r should be carried out [default TRUE].
plot.theme	Theme for the plot. See details [default theme_dartR()].
plot.colors.pop	A color palette for populations or a list with as many colors as there are populations in the dataset [default NULL].
CI.color	Color for the shade of the 95% confidence intervals around the r estimates [default "red"].
plot.out	Specify if plot is to be produced [default TRUE].
plot.file	Name for the RDS binary file to save (base name only, exclude extension) [default NULL].
plot.dir	Directory in which to save files [default = working directory]
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].

Details

This function executes a modified version of spautocorr from the package PopGenReport. Differently from PopGenReport, this function also computes the 95% confidence intervals around the r via bootstraps, the 95 null hypothesis of no spatial structure and the one-tail test via permutation, and the correction factor described by Peakall et al 2003.

The input can be i) a genlight object (which has to have the latlon slot populated), ii) a pair of Dgeo and Dgen, which have to be either matrix or dist objects, or iii) a list of the matrix or dist objects if the analysis needs to be carried out for multiple populations (in this case, all the elements of the list have to be of the same class (i.e. matrix or dist) and the population order in the two lists has to be the same.

If the input is a genlight object, the function calculates the linear distance for Dgeo and the relevant Dgen matrix (see Dgen_method) for each population. Genetic distances are used as distances, so that positive autocorrelation coefficients indicate more related individuals, as in GenAlEx. The Euclidean distance is squared, as in Smouse and Peakall (1999) and GenAlEx. Similarity matrices are converted to distances: 'propShared' as 1 - Dgen and 'grm' (a relationship matrix G) as G[i,i] + G[j,j] - 2 * G[i,j]. If the user provides the matrices, Dgen must be a distance: a similarity matrix will generate negative values for closely related individuals.

With coordinates = "latlon" (or a lat/lon data.frame), geographic distances are geodesic distances in metres (package terra). With 'xy' or an x/y data.frame, they are Euclidean distances in the coordinate units.

If max(Dgeo)>1000 (e.g. the geographic distances are in thousands of metres), the distance class labels in the plot are divided by 1000 (in the example before these would then become km) to facilitate readability. The returned table keeps the original units.

If bins is of length = 1 it is interpreted as the number of (even) bins to use. In this case the first class starts at the minimum value in the distance matrix, the classes have equal width, and the last ends at the maximum. If it is a numeric vector of length>1, it is interpreted as the breaking points. In this case, the first has to be the lowest value, and the last has to be the highest. There are no internal checks for this and it is user responsibility to ensure that distance classes are properly set up. If that is not the case, data that fall outside the range provided will be dropped. The number of bins will be length(bins) - 1.

The permutation constructs the 95% confidence intervals around the null hypothesis of no spatial structure (this is a two-tail test). The same data are also used to calculate the probability of the one-tail test as (number of permutations at least as extreme + 1) / (reps + 1) (See references below for details).

Bootstrap calculations are skipped and NA is returned when the number of possible combinations given the sample size of any given distance class is < reps.

Methods available to calculate genetic distances for SNP data:

- "propShared" using the function gl.propShared.
- "grm" using the function gl.grm2.
- "Euclidean" using the function gl.dist.ind.
- "Simple" using the function gl.dist.ind.
- "Absolute" using the function gl.dist.ind.
- "Manhattan" using the function gl.dist.ind.

Methods available to calculate genetic distances for SilicoDArT data:

- "Euclidean" using the function gl.dist.ind.
- "Simple" using the function gl.dist.ind.
- "Jaccard" using the function gl.dist.ind.

- "Bray-Curtis" using the function gl.dist.ind.

Examples of other themes that can be used can be consulted in

- https://ggplot2.tidyverse.org/reference/ggtheme.html and
- https://yutannihilation.github.io/allYourFigureAreBelongToUs/ggthemes/

Value

Returns a list with one data frame per population (named by population), each with the following columns:

1. Bin The distance classes
2. N The number of pairwise comparisons within each distance class
3. r.uc The uncorrected autocorrelation coefficient
4. Correction the correction
5. r The corrected autocorrelation coefficient
6. L.r The corrected autocorrelation coefficient lower limit (if `bootstap = TRUE`)
7. U.r The corrected autocorrelation coefficient upper limit (if `bootstap = TRUE`)
8. L.r.null.uc The uncorrected lower limit for the null hypothesis of no spatial autocorrelation (if `permutation = TRUE`)
9. U.r.null.uc The uncorrected upper limit for the null hypothesis of no spatial autocorrelation (if `permutation = TRUE`)
10. L.r.null The corrected lower limit for the null hypothesis of no spatial autocorrelation (if `permutation = TRUE`)
11. U.r.null The corrected upper limit for the null hypothesis of no spatial autocorrelation (if `permutation = TRUE`)
12. p.one.tail The p value of the one tail statistical test

Plots and table are saved plot.file in plot.dir if specified. Bootstraps and permutations (if requested) are saved in a temporary directory

Author(s)

Author(s): Carlo Pacioni, Bernd Gruber & Luis Mijangos. Custodian: Carlo Pacioni – Post to https://groups.google.com/d/forum/dartr

References

- Smouse PE, Peakall R. 1999. Spatial autocorrelation analysis of individual multiallele and multilocus genetic structure. Heredity 82: 561-573.
- Double, MC, et al. 2005. Dispersal, philopatry and infidelity: dissecting local genetic structure in superb fairy-wrens (Malurus cyaneus). Evolution 59, 625-635.
- Peakall, R, et al. 2003. Spatial autocorrelation analysis offers new insights into gene flow in the Australian bush rat, Rattus fuscipes. Evolution 57, 1182-1195.

- Smouse, PE, et al. 2008. A heterogeneity test for fine-scale genetic structure. Molecular Ecology 17, 3389-3400.
- Gonzales, E, et al. 2010. The impact of landscape disturbance on spatial genetic structure in the Guanacaste tree, Enterolobium cyclocarpum (Fabaceae). Journal of Heredity 101, 133-143.
- Beck, N, et al. 2008. Social constraint and an absence of sex-biased dispersal drive fine-scale genetic structure in white-winged choughs. Molecular Ecology 17, 4346-4358.

See Also

Other spatial analysis functions: `gl.costdistances()`, `gl.genleastcost()`

Examples

```
require("dartR.data")
res <- gl.spatial.autoCorr(platypus.gl, bins=seq(0,10000,2000))
# using one population, showing sample size
test <- gl.keep.pop(platypus.gl,pop.list = "TENTERFIELD")
res <- gl.spatial.autoCorr(test, bins=seq(0,10000,2000),CI.color = "green")
```

`gl2shp` *Converts a genlight object to ESRI shapefiles or kml files*

Description

This function exports the coordinates in a genlight object to a point shape file or kml file, with the individual metrics as attributes.

Usage

```
gl2shp(
  x,
  type = "shp",
  proj4 = "+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs",
  outfile = "gl",
  outpath = tempdir(),
  verbose = NULL
)
```

Arguments

`x`	Name of the genlight object containing the SNP data and location data, lat longs [required].
`type`	Type of output 'kml' or 'shp' [default 'shp'].
`proj4`	Proj4string of data set (see spatialreference.org for projections) [default WGS84].

outfile	Name (path) of the output shape file [default 'gl']. shp extension is added automatically.
outpath	Path where to save the output file; the folder must exist [default tempdir(), mandated by CRAN]. Use outpath=getwd() or outpath='.' when calling this function to direct output files to your working directory.
verbose	Verbosity: 0, silent or fatal errors; 1, begin and end; 2, brief progress messages; 3, progress and results summary; 5, full report [default 2, unless specified using gl.set.verbosity].

Details

Coordinates are taken from x@other$latlon (columns lat and lon; a column named long is accepted for lon) and are assumed to be in WGS84 unless a proj4 string is provided.

Individuals with a missing lat or lon are removed. Missing values in other individual metrics are kept and written as NA.

The attribute table holds the columns of x@other$ind.metrics plus lat and lon. If ind.metrics has no id column, an id column with the individual names is added. ESRI shapefiles truncate column names to 10 characters.

Value

A terra SpatVector with the exported points. The file is written to outpath.

Author(s)

Author(s): Bernd Gruber. Custodian: Bernd Gruber – Post to https://groups.google.com/d/forum/dartr

Examples

```
out <- gl2shp(testset.gl, outpath=tempdir())
```

---

utils.spautocor *Spatial autocorrelation coefficient calculations*

Description

Carries out calculation for spatial autocorrelation coefficient starting from a genetic and geogaphic distance matrix.

Usage

```
utils.spautocor(
  GD,
  GGD,
  permutation = FALSE,
  bootstrap = FALSE,
  bins = 10,
  reps
)
```

Arguments

GD	Genetic distance matrix.
GGD	Geographic distance matrix.
permutation	Whether permutation calculations for the null hypothesis of no spatial structure should be carried out [default TRUE].
bootstrap	Whether bootstrap calculations to compute the 95% confidence intervals around r should be carried out [default TRUE].
bins	The number of bins for the distance classes (i.e. `length(bins) == 1`) or a vectors with the break points. See details [default 5].
reps	The number to be used for permutation and bootstrap analyses [default 100].

Details

The code of this function is based one `spautocorr` from the package `PopGenReport`, which has been modified to fix a few bugs (as of `PopGenReport v 3.0.4` and allow calculations of bootstraps estimates.

See details from `gl.spatial.autoCorr` for a detailed explanation.

Value

Returns a data frame with the following columns:

1. Bin The distance classes
2. N The number of pairwise comparisons within each distance class
3. r.uc The uncorrected autocorrelation coefficient

if both `bootstap` and `permutation` are `FALSE` otherwise only `r` estimates are returned

Author(s)

Carlo Pacioni & Bernd Gruber

References

- Smouse PE, Peakall R. 1999. Spatial autocorrelation analysis of individual multiallele and multilocus genetic structure. Heredity 82: 561-573.
- Double, MC, et al. 2005. Dispersal, philopatry and infidelity: dissecting local genetic structure in superb fairy-wrens (Malurus cyaneus). Evolution 59, 625-635.
- Peakall, R, et al. 2003. Spatial autocorrelation analysis offers new insights into gene flow in the Australian bush rat, Rattus fuscipes. Evolution 57, 1182-1195.
- Smouse, PE, et al. 2008. A heterogeneity test for fine-scale genetic structure. Molecular Ecology 17, 3389-3400.
- Gonzales, E, et al. 2010. The impact of landscape disturbance on spatial genetic structure in the Guanacaste tree, Enterolobium cyclocarpum(Fabaceae). Journal of Heredity 101, 133-143.
- Beck, N, et al. 2008. Social constraint and an absence of sex-biased dispersal drive fine-scale genetic structure in white-winged choughs. Molecular Ecology 17, 4346-4358.

See Also

gl.spatial.autoCorr

Examples

```
# See gl.spatial.autoCorr
```

Index

∗ **distance**
gl.kosman, 11
∗ **inbreeding functions**
gl.grm2, 6
∗ **linker**
gl2shp, 20
∗ **spatial analysis functions**
gl.costdistances, 2
gl.genleastcost, 4
gl.spatial.autoCorr, 16

A.mat, *6*, *8*

commuteDistance, *3*
costDistance, *3*

dist, *9*

geoCorrection, *3*
gl.costdistances, 2, *4–6*, *20*
gl.dist.ind, *18, 19*
gl.genleastcost, *3*, 4, *20*
gl.grm2, 6
gl.ibd, 9
gl.kosman, *5*, 11
gl.propShared, *5*, *18*
gl.run.eems, 12
gl.spatial.autoCorr, *3*, *6*, 16, *23*
gl2shp, 20

mantel, *11*

rSPDistance, *3*

stamppFst, *9*, *11*
stamppNeisD, *9*

utils.spautocor, 21