Package ‘diveRsity’

April 4, 2017

Version 1.9.90

Date 2017-03-17

Title A Comprehensive, General Purpose Population Genetics Analysis Package

Author Kevin Keenan <kkeenan02@qub.ac.uk>

Maintainer Kevin Keenan <kkeenan02@qub.ac.uk>

Description Allows the calculation of both genetic diversity partition statistics, genetic differentiation statistics, and locus informativeness for ancestry assignment.
It also provides users with various option to calculate bootstrapped 95\{ }% confidence intervals both across loci, for pairwise population comparisons, and to plot these results interactively.
Parallel computing capabilities and pairwise results without bootstrapping are provided.
Also calculates F-statistics from Weir and Cockerham (1984).
Various plotting features are provided, as well as Chi-square tests of genetic heterogeneity.
Functionality for the calculation of various diversity parameters is possible for RAD-seq derived SNP data sets containing thousands of marker loci.
A shiny application for the development of microsatellite multiplexes is also available.

Suggests xlsx, sendplot, plotrix, parallel, HWxtest

Depends R (>= 3.0.0)

Imports ggplot2, shiny, qgraph, Rcpp, methods, stats, grid

LinkingTo Rcpp

Repository CRAN

License GPL (>= 2)

URL http://diversityinlife.weebly.com/

NeedsCompilation yes

Date/Publication 2017-04-04 10:59:38 UTC

R topics documented:

`arp2gen` *Fast and simple conversion of arlequin (.arp) genotype files to genepop files, form within the* R *environment.*

Description

`arp2gen` allows simple file conversion from the arlequin genotype format to the genepop genotype format. Arlequin files can gave 2-digit or 3-digit allele records. No file size limit is imposed, however, system RAM is a limiting factor.

Usage

```
arp2gen(infile)
```

Arguments

`infile` Specifying the name of the *‘.arp’* arlequin genotype file. The argument must be a character string of the file name if it is located in the current working directory, or the file path (relative or absolute) if not.

Details

Following the input of a .arp file, `arp2gen` will write a .gen file to the same directory as the original .arp file. The output file will have the same name as the original `infile`, with the exception of the file extension, which will be .gen following file conversion. The original .arp file will remain unmodified.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

`basicStats` *Claculate basic descriptive population parameters from a genepop genotype file*

Description

A new and improved version of `divBasic`, with a more comprehensive set of control arguments, as well as better memory and CPU efficiency. Additional options for testing homozygosity/heterozygosity excess are also provided.

Usage

```
basicStats(infile = NULL, outfile = NULL, fis_ci = FALSE,
           ar_ci = FALSE, fis_boots = NULL, ar_boots = NULL,
           mc_reps = 9999, rarefaction = TRUE, ar_alpha = 0.05,
           fis_alpha = 0.05)
```

Arguments

`infile` Specifying the name of the *‘genepop’*(Rousset, 2008) file from which the statistics are to be calculated. This file can be in either the 3 digit of 2 digit format, and must contain only one *whitespace* separator (e.g. “space” or “tab”) between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in “” or ‘’).

`outfile` A character string specifying the name to be given to the output tab delimited file. Default value of `NULL` means that no file is written.

`fis_ci` A logical argument specifying whether confidence limits should be calculated for the inbreeding coefficient.

`ar_ci` A logical argument specifying whether confidence limits should be calculated for resampleed allelic richness. Only valid when `rarefaction = FALSE`.

`fis_boots` An integer specifying the number of bootstrap replicate used to calculate confidence limits for the inbreeding coefficient if `fis_ci = TRUE`

`ar_boots` An integer specifying the number of bootstrap replicate used to calculate confidence limits for allelic richness if `ar_ci = TRUE` and `rarefaction = FALSE`

`mc_reps` An integer specifying the number of Monte Carlo replicates used when carrying out psudo-exact HWE tests

`rarefaction` A logical indicating whether allelic richness should be calculated using rarefaction. If `FALSE` user should specify the number of `ar_boots` to use when estimating allelic richness using the resample method

`ar_alpha` A numeric argument specifying the alpha used to calculate the confidence interval for resample based allelic richness. Valid if `rarefaction = FALSE`, `ar_boots > 2` and `ar_ci = TRUE`. Confidence limits are defined as alpha/2 and 1-(alpha/2).

`fis_alpha` A numeric argument specifying the alpha used to calculate the confidence interval for resample based allelic richness. Valid if `fis_boots > 2` and `fis_ci = TRUE`. Confidence limits are defined as alpha/2 and 1-(alpha/2).

Details

HWE tests are carried out using methods originally implemented in the package `HWxtest`. Test of HWE proportions are either carried out using either exact testing (when the number of genotype tables is less than 10 million), or through Monte Carlo simulations, where the number of random trials can be specified using the argument `mc_reps`. HWE tests will also be automatically carried out to assess homozygosity/hetrozygosity excess (i.e. one tailed tests). The test results returned is automatically determined by the proporties of the original data. See `?HWx.test` for more details.

Value

A list of results

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Bill Engels (2014). HWxtest: Exact Tests for Hardy-Weinberg Proportions. R package version 1.0.5. http://CRAN.R-project.org/package=HWxtest

Examples

```
## Not run:
# To run an example use the following format
library("diveRsity")
data(Test_data)
test_results <- basicStats(infile = Test_data, outfile = "test_run",
                           fis_ci = TRUE, ar_ci = TRUE, fis_boots = 999,
                           ar_boots = 999, mc_reps = 9999,
                           rarefaction = FALSE, ar_alpha = 0.05,
                           fis_alpha = 0.05)

## End(Not run)
```

`bigDivPart` *Genetic differentiation statistics and their estimators for high throughput data (e.g. RAD-seq derived SNPS)*

Description

`bigDivPart` allows for the calculation of four main diversity partition statistics and their respective estimators from large genepop files.

Usage

```
bigDivPart(infile = NULL, outfile = NULL,
           WC_Fst = FALSE, format = NULL)
```

Arguments

`infile` Specifying the name of the *‘genepop’* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format, and must contain only one whitespace separator (e.g. “space” or “tab”) between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in “” or ‘’).

`outfile` Allows users to specify a prefix for an output folder. Name must a character string enclosed in either “” or ‘’.

`WC_Fst` A Logical argument indicating whether Weir & Cockerham’s 1984 F-statistics should be calculated.

`format` A character string specifying the preferred output format for calculated results. The arguments `txt` or `xlsx` are valid when `outfile` is not `NULL`.

Details

All results are optionally written to a user defined directory in either .xlsx or .txt format. The function is identical to the `divPart` function without the pairwise or bootstrapping functionality.

Value

standard
: A matrix containing identical data to the `Standard_stats` worksheet in the `.xlsx` workbook.

estimate
: A matrix containing identical data to the `Estimated_stats` worksheet in the `.xlsx` workbook.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Dragulescu, A.D., "xlsx: Read, write, formal Excel 2007 and Excel 97/2000/xp/2003 files", R package version 0.4.2, url:http://CRAN.R-project.org/package=xlsx, (2012).

Guile, D.P., Shepherd, L.A., Sucheston, L., Bruno, A., and Manly, K.F., "sendplot: Tool for sending interactive plots with tool-tip content.", R package version 3.8.10, url: http://CRAN.R-project.org/package=sendplot, (2012).

Hedrick, P., "A standardized genetic differentiation measure," Evolution, vol. 59, no. 8, pp. 1633-1638, (2005).

Jost, L., "G ST and its relatives do not measure differentiation," Molec- ular Ecology, vol. 17, no. 18, pp. 4015-4026, (2008).

Manly, F.J., "Randomization, bootstrap and Monte Carlo methods in biology", Chapman and Hall, London, 1997.

Nei, M. and Chesser, R., "Estimation of fixation indices and gene diver- sities," Ann. Hum. Genet, vol. 47, no. Pt 3, pp. 253-259, (1983).

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Revolution Analytics (2012). doParallel: Foreach parallel adaptor for the parallel package. R package version 1.0.1. http://CRAN.R-project.org/package=doParallel

Revolution Analytics (2012). foreach: Foreach looping construct for R. R package version 1.4.0. http://CRAN.R-project.org/package=foreach

Rousset, F., "genepop'007: a complete re-implementation of the genepop software for Windows and Linux.," Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

Examples

```
## Not run:
# simply use the following format to run the function

test_result <- bigDivPart(infile = 'mydata', outfile = "myresults',
                          WC_Fst = TRUE, format = "txt")

## End(Not run)
```

`Big_data` *Example infile for diveRsity package*

Description

big_data is a typical genepop (Rousset, 2008) two digit format input file for the package diveRsity. The data was simulated using a hierarchical island model containing five island groups with ten local demes each. Migration within island groups was much larger that between.

Usage

```
Big_data
```

Format

genepop 2 digit.

References

Rousset, F., "genepop'007: a complete re-implementation of the genepop software for Windows and Linux.," Molecular Ecology Resources, vol. 8, no. 1, pp. 103-6, (2008).

`chiCalc` *Testing sample independence from genotype counts*

Description

`chiCalc` carries out Fisher's exact tests of sample independence using genotype data.

Usage

```
chiCalc(infile = NULL, outfile = NULL, pairwise = FALSE, mcRep = 2000)
```

Arguments

`infile`	A character string indicating the location and name of a genepop format file to be read. If the file is in the current working directory, only the name must be provided. If the file is in a directory other than the current working directory, either a relative or absolute path to the file must be provided. The genepop file can be in the 2-digit or 3-digit allele format.
`outfile`	A character string indicating the prefix to be added to the results directory created. All results files will be written to this directory.
`pairwise`	A logical argument indicating whether sample independence should be tested between all population pairs.
`mcRep`	An integer specifying the number Monte Carlo test replicates. See `?fisher.test` for more information.

Details

All results will be written to a user defined folder ("working_directory/outfile"), providing an argument is passed for 'outfile'. Otherwise, results will only be returned to the workspace.

Fisher's exact tests are carried out using the function `fisher.test`. Multilocus p values are calculated using Fisher's method for combining p value.

Value

`overall` A data frame containing p values calculated across all population samples, per locus and across all loci.

`multilocus_pw` Generated if `pairwise = TRUE`. The object is a data frame containing multilocus p value calculated for all population pairs.

`locus_pw` A dataframe containing locus p values calculated for all pairs of populations. Rows represent loci, while columns represent pairwise comparisons.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

Examples

```
## Not run:
# To run an example use the following format
library(diveRsity)
data(Test_data)
test_results <- chiCalc(infile = Test_data, outfile = NULL,
                        pairwise = TRUE, mcRep = 5000)

## End(Not run)
```

---

corPlot

A function to plot the relationship between Gst, G'st, Theta, D (Jost) and the mean number of alleles at a locus. A reimplementation of `corPlot`. *This function will replace the old function in later version of the package.*

Description

`corPlot` uses the information calculated by `diffCalc` to plot and calculate the relationship between Gst, G'st, Theta, D (Jost) and the mean number of alleles per locus. This information can then be used to assess the likelihood that data derived from the loci are suitable for the calculation of population demography using Fst or its analogues. This is based on the assumption that where $u > m$, demographic process are obscured by mutation.

Usage

```
corPlot(infile = NULL, write = FALSE, plot.format = NULL)
```

Arguments

`infile` A character string indicating the location and name of a genepop format file to be read. If the file is in the current working directory, only the name must be provided. If the file is in a directory other than the current working directory, either a relative or absolute path to the file must be provided. The genepop file can be in the 2-digit or 3-digit allele format.

`write` A logical argument indicating whether results should be written to file or not

`plot.format` A string indicating the format to which plots should be written. Either 'png' or 'eps' are accepted.

Details

This function returns four scatter plots showing the relationship between Fst (Weir and Cockerham 1984), Gst (Nei and Chesser 1983), G'st (Hedrick 2005) and D (Jost 2008) and the mean number of alleles per locus. The function allows user to write plots to file, and return a four panelled figure. Plots are generated using the `ggplot2` package and arranged in a grid using the `multiplot` function by Winston Chang (Chang 2012).

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Cheng, W., R Graphics Cookbook, O'Reilly Media inc. 2012.

Hedrick, P., "A standardized genetic differentiation measure," Evolution, vol. 59, no. 8, pp. 1633-1638, (2005).

Jost, L., "G ST and its relatives do not measure differentiation," Molec- ular Ecology, vol. 17, no. 18, pp. 4015-4026, (2008).

Nei, M. and Chesser, R., "Estimation of fixation indices and gene diver- sities," Ann. Hum. Genet, vol. 47, no. Pt 3, pp. 253-259, (1983).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

`diffCalc` *A faster function for calculating genetic differentiation statistics*

Description

This function allows the calculation of pairwise differentiation using a range of population statistics, such as Gst (Nei & Chesser, 1983), G'st (Hedrick, 2005), theta (Weir & Cockerham, 1984) and D (Jost, 2008). These parameters can also be calculated at the global and locus levels. Significance of differentiation can be assessed through the calculation of 95% confidence limits using a bias corrected bootstrapping method. The functionality `diffCalc` is similar to the `fastDivPart` function. However `diffCalc` is much faster and more memory efficient than `fastDivPart`. This function also only allows results to be written to text files rather than xlsx file (as in `fastDivPart`. No plotting options are provide in `diffCalc`.)

Usage

```
diffCalc(infile = NULL, outfile = NULL, fst = FALSE, pairwise = FALSE,
         bs_locus = FALSE, bs_pairwise = FALSE, boots = NULL,
         ci_type = "individuals", alpha = 0.05, para = FALSE)
```

Arguments

`infile` Specifying the name of the *'genepop'* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format. See `http://genepop.curtin.edu.au/help_input.html` for detail on the genepop file format.

`outfile` A character string specifying the name of the folder to which results should be written.

`fst` A Logical argument indicating whether Weir & Cockerham's 1984 F-statistics should be calculated. NOTE - Calculating these statistics adds significant time to analysis when carrying out pairwise comparisons.

`pairwise` A logical argument indicating whether standard pairwise diversity statistics should be calculated and returned as a diagonal matrix.

`bs_locus` Gives users the option to calculate bias corrected 95% confidence intervals for locus statistic species.

`bs_pairwise` Gives users the option to calculate bias corrected 95% confidence intervals for pairwise statistics.

`boots` Specified the number of bootstraps for the calculation of 95% confidence intervals.

`ci_type` A character string indicating whether bootstrapping should be carried out over individuals within samples (``` ``individuals'' ```.), or across loci (``` ``loci'' ```).

`alpha` A numeric argument, specifying the alpha value used to estimate confidence limits for relevant parameters. Both the alpha/2 and the 1-(alpha/2) quantiles will be returned. Default value results in 95% CI.

`para` A logical argument indicating whether computations should be carried out over multiple CPUs, if available.

Value

If `outfile` is given as a character string, all results will be written to text files. The files will be written to a directory under the current working directory. The number of files written depends on the options choose. As well as this a list object is returned to the R workspace, containing the following results:

`std_stats` A `data.frame`, containing locus estimates for Gst, G'st, G"st, D (and Weir and Cockerham's F-statistics, if `fst = TRUE`). The last row of this dataframe contains the global estimate for each statistic across all samples and loci.

`global_bs` If `bs_locus = TRUE`, this object is returned. It is a `dataframe` containing global estimates and lower and upper 95% CIs for all relevant statistics.

`bs_locus` If `bs_locus = TRUE`, this object is returned. It is a list of either 4 or 7 dataframes the number of which depend on `fst`. Each dataframe contains the locus statistics estimate across all samples along with lower and upper 95% confidence limits.

`pw_locus` If `pairwise = TRUE`, this list of dataframes is returned. The list contains a dataframe for each relevant statistics, where rows correspond to loci and columns correspond to all possible pairwise combinations of samples. If `outfile` is provided, these results are also written to file.

`pairwise` If `pairwise = TRUE`, this object is returned. It is a list of either 3 or 4 matrices (depending on the value of `fst`) containing pairwise estimate of relevant statistics.

`bs_pairwise` A list of either 4 or 5 (depending on the value of `fst`) containing pairwise estimates and lower and upper 95% confidence intervals for all relevant statistics.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Eddelbuettel, D., and Francois, R., (2011). Rcpp: Seamless R and C++ Integration. Journal of Statistical Software, 40(8), 1-18. URL http://www.jstatsoft.org/v40/i08/.

Hedrick, P., “A standardized genetic differentiation measure,” Evolution, vol. 59, no. 8, pp. 1633-1638, (2005).

Jost, L., “G ST and its relatives do not measure differentiation,” Molec- ular Ecology, vol. 17, no. 18, pp. 4015-4026, (2008).

Manly, F.J., “Randomization, bootstrap and Monte Carlo methods in biology”, Chapman and Hall, London, 1997.

Meirmans, P.G., and Hedrick, P.W., (2011), Assessing population structure: Fst and related measures., Molecular Ecology, Vol. 11, pp5-18. doi: 10.1111/j.755-0998.2010.02927.x

Nei, M. and Chesser, R., “Estimation of fixation indices and gene diver- sities,” Ann. Hum. Genet, vol. 47, no. Pt 3, pp. 253-259, (1983).

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

Examples

```
## Not run:
# simply use the following format to run the function
library(diveRsity)
data(Test_data)
Test_data[is.na(Test_data)] <- ""
```

```
test_result <- diffCalc(infile = Test_data, outfile = "myresults",
                        fst = TRUE, pairwise = TRUE, bs_locus = TRUE,
                        bs_pairwise = TRUE, boots = 1000, para = TRUE)

## End(Not run)
```

diffPlot *A function to plot pairwise statistics calculated by divPart.*

Description

This function uses results from `divPart` to plot pairwise statistic estimators.

Usage

```
diffPlot(x, outfile = NULL, interactive = FALSE)
```

Arguments

`x`	Results object returned from the function 'divPart'
`outfile`	A character string indication the folder location to which plot files should be written.
`interactive`	A logical argument indication whether the package 'sendplot' should be used to plot 'divPart' pairwise results.

Details

A number of heatmap style plots of pairwise differentiation are generated. The function takes the output from either `fastDivPart` or `diffPlot` and writes interactive `HTML` plots to file. The number of plots depends on the input structure. For instance, if `fst = FALSE` in either `fastDivPart` or `diffCalc`, then a plot containing pairwise Fst will be produced.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

divBasic *A function to calculate basic population parameters such as allelic richness, observed heterozygosity, as well as expected heterozygosity.*

Description

`divBasic` allows the calculation of locus and overall basic population parameters. `divBasic` will write results to a *.xlsx* workbook. The function accepts co-dominant genetic data in both 2 and 3 digit genepop formats.

Usage

```
divBasic(infile = NULL, outfile = NULL, gp = 3, bootstraps = NULL,
         HWEexact = FALSE, mcRep = 2000)
```

Arguments

`infile` Specifying the name of the *'genepop'*(Rousset, 2008) file from which the statistics are to be calculated. This file can be in either the 3 digit of 2 digit format, and must contain only one *whitespace* separator (e.g. "space" or "tab") between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in "" or '').

`outfile` Allows users to specify a prefix for an output folder. Name must a character string enclosed in either "" or ''.

`gp` Specifies the digit format of the `infile`. Either 3 (default) or 2.

`bootstraps` This argument specifies how many bootstrap iterations should be executed when calculating 95% confidence intervals for $F_is$. The argument should be an integer greater than 1. Setting `bootstrap = NULL` suppresses the calculation of F_is. Users should note that setting this argument to values greater than 1000 may result in longer executions times.

`HWEexact` A logical argument specifying if HWE testing should be carried out using Fisher's exact tests.

`mcRep` An integer specifying the number of replicates to use for the Monte Carlo tests if HWEexact is TRUE.

Details

All results will be written to a user defined folder ("working_directory/outfile"), providing an argument is passed for 'outfile'. Results will be written to `.xlsx` files, and multiple `R` objects are also written to the current environment.

HWE tests can be carried out using either a standard Chisq goodness of fit method, or using Fisher's exact method. The standard chisq test behave poorly when there are classes with low numbers of observations (e.g. hypervariable microsatellite loci). In such instance it is advisable to use exact testing. Multi-locus HWE is tested using the standard chisq method by summing chisq difference and degrees of freedom across loci, and using these parameter to derive a pvalue for the test. When using exact testing, the multi-locus pvalue is determined using Fisher's method for combining pvalue from independent tests. This process assumes that loci are unlinked.

Value

`locus_pop_size` A matrix containing the number of individuals typed per locus per population sample. Mean values across loci are also given.

`Allele_number` A matrix containing the number of alleles observed per locus per population sample. Mean values across loci are also given.

`proportion_Alleles`
A matrix containing the percentage of total alleles observed per locus per population sample. Mean values across loci are also given.

`Allelic_richness`
A matrix containing the allelic richness per locus per population sample. Allelic richness is calculated using 1000 re-samples (n = smallest sample in the input data file), with replacement per population sample locus per population sample. Mean values across loci are also given.

`Ho` A matrix containing observed heterozygosity per locus per population sample. Mean values across loci are also given.

`He` A matrix containing expected heterozygosity per locus per population sample. Mean values across loci are also given.

`HWE` A matrix containing uncorrected *p*-values from chi-square test for goodness-of-fit to Hardy-Weinberg equilibrium. Overall *p*-values are also given per population sample.

`fis` A list of dataframes containing locus and global F_is values for each population sample. In each dataframe the actual F_is is listed in the first column of the matrix, lower and upper 95% confidence intervals are listed in the next two columns, while bias corrected 95% CI are listed in the last two columns. This object is only returned when `bootstraps` is not `NULL`

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Rousset, F., "genepop'007: a complete re-implementation of the genepop software for Windows and Linux.," Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Examples

```
## Not run:
# To run an example use the following format

test_results <- divBasic(infile = Test_data, outfile = 'out', gp = 3, bootstraps = 1000)

## End(Not run)
```

divMigrate — *An experimental function for the detection of directional differentiation from microsatellite data.*

Description

divMigrate uses the method described in Sundqvist *et al.,* 2013 to plot the relative migration levels between population samples, from microsatellite allele frequency data. The method is still in the experimental stages, and it is not clear how well it performs under most evolutionary scenarios. Caution should be exercised.

Usage

```
divMigrate(infile = NULL, outfile = NULL, boots = 0, stat = "all",
           filter_threshold = 0, plot_network = FALSE,
           plot_col = "darkblue", para = FALSE)
```

Arguments

infile	Input genepop file name or path.
outfile	String prefix to be added to the results directory. If left NULL, no networks will be written to file.
boots	The number of bootstrap iteration to carry out for calculating mean relative migration and 95% confidence intervals.
stat	A character string indicating which statistic should be used to estimate relative migration between populations. The argument accepts one of the following: "d" (Jost's D), "gst" (Nei's Gst), "Nm" (Alcala et al, 2014) or "all" (all of the preceeding statistics; default).
filter_threshold	The minimum relative migration value for which edges in the networks should be displayed.
plot_network	A logical argument, specifying whether migration results should be plotted in a network.
plot_col	Defines the colour of edges in networks. Default is set to "darkblue".
para	A logical argument, specifying if multiple CPUs should be used when available.

Details

The function will except both Arlequin (.arp) genotype and genepop (.gen/.txt) files, containing codominant diploid data. Using the method outlined in Sundqvist *et al.,* 2013, the relative migration levels between all pairs of populations is determined. A weighted network plot is returned, as well as four matrices containing the objects described below.

Value

`Relative migration`
: Relative migration matrices

`Significant directional migration`
: If `nbs > 0` the significance of pairwise relative migration is tested (i.e. non-overlapping 95% CIs). All non-significant values in the standard relative migration matrices are replaced with 0.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Lisa Sundqvist, M.Z. & Kleinhans, D., 2013. Directional genetic differentiation and asymmetric migration. arXiv pre-print: arXiv:1304.0118v2

Rousset, F., "genepop'007: a complete re-implementation of the genepop software for Windows and Linux.," Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Alcala, N., Goudet, J., Vuilleumier, S., 2014, On the transition of genetic differentiation from isolation to panmixia: What we can learn from Gst and D, Theoretical Population Biology, Vol 93, pp75-84.

`divOnline` *A function to launch a web app version of* `diveRsity` *from the local system.*

Description

web app build using shiny.

Usage

```
divOnline()
```

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

divPart *Genetic differentiation statistics and their estimators*

Description

divPart (diversity partition), allows for the calculation of three main diversity partition statistics and their respective estimators. The function can be used to mainly explore locus values to identify 'outliers' and also to visualise pairwise differentiation between populations. Bootstrapped confidence intervals are calculated also. Results can. be optionally plotted for data exploration.

Usage

```
divPart(infile = NULL, outfile = NULL, gp = 3,
        pairwise = FALSE, WC_Fst = FALSE,
        bs_locus = FALSE,
        bs_pairwise = FALSE,
        bootstraps = 0, plot = FALSE,
        parallel = FALSE)
```

Arguments

infile — Specifying the name of the *'genepop'* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format, and must contain only one whitespace separator (e.g. "space" or "tab") between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in "" or '').

outfile — Allows users to specify a prefix for an output folder. Name must a character string enclosed in either "" or ''.

gp — Specifies the digit format of the `infile`. Either 3 (default) or 2.

pairwise — A logical argument indicating whether standard pairwise diversity statistics should be calculated and returned as a diagonal matrix.

WC_Fst — A Logical argument indicating whether Weir & Cockerham's 1984 F-statistics should be calculated. NOTE - Calculating these statistics adds significant time to analysis when carrying out pairwise comparisons.

bs_locus — Gives users the option to bootstrap locus statistics. Results will be written to `.xlsx` workbook by default if the package `'xlsx'` is installed, and to a `.html` file if `plot=TRUE`. If `'xlsx'` is not installed, results will be written to `.txt` files.

bs_pairwise — Gives users the option to bootstrap statistics across all loci for each pairwise population comparison. Results will be written to a `.xlsx` file by default if the package `'xlsx'` is installed, and to a `.html` file if `plot=TRUE`. If `'xlsx'` is not installed, results will be written to `.txt` files.

`bootstraps` Determines the number of bootstrap iterations to be carried out. The default value is `bootstraps = 0`, this is only valid when all bootstrap options are false. There is no limit on the number of bootstrap iterations, however very large numbers of bootstrap iterations (< 1000) on even modest data sets (e.g. 265 individuals x 38 loci) will take over 30 minutes to run on a most PCs).

`plot` Optional interactive `.html` image file of the plotted bootstrap results for loci if `bs_locus = TRUE` and pairwise population comparisons if `bs_pairwise = TRUE`. The default option is `plot = FALSE`.

`parallel` A logical input, indicating whether your analysis should be run in parallel mode or sequentially. `parallel = TRUE` is only valid if the packages, `parallel`, `doParallel` and `foreach` are installed.

Details

All results will be written to a user defined folder ("working_directory/outfile"). The format of outputs will vary depending on the availability of the package `'xlsx'` in the users local package library. If `'xlsx'` is available, results will be written to an Excel workbook. If `'xlsx'` is not available, results will be written to `.txt` files.

Value

`standard` A matrix containing identical data to the `Standard_stats` worksheet in the `.xlsx` workbook.

`estimate` A matrix containing identical data to the `Estimated_stats` worksheet in the `.xlsx` workbook.

`pairwise` A group of six matrices containing population pairwise statistics. This object is identical to that written as ‘pairwise-stats‘ in the `.xlsx` workbook.

`bs_locus` A list containing six matrices of locus values for *Gst*, *G’st*, *D(Jost)*, *Gst-(est)*, *G’st-(est)*, and *D(Jost)-(est)* along with their respective 95% confidence interval.

`bs_pairwise` A list containing six matrices of pairwise values for *Gst*, *G’st*, *D(Jost)*, *Gst-(est)*, *G’st-(est)*, and *D(Jost)-(est)* along with their respective 95% confidence intervals.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Dragulescu, A.D., “xlsx: Read, write, formal Excel 2007 and Excel 97/2000/xp/2003 files”, R package version 0.4.2, url:http://CRAN.R-project.org/package=xlsx, (2012).

Guile, D.P., Shepherd, L.A., Sucheston, L., Bruno, A., and Manly, K.F., “sendplot: Tool for sending interactive plots with tool-tip content.”, R package version 3.8.10, url: http://CRAN.R-project.org/package=sendplot, (2012).

Hedrick, P., “A standardized genetic differentiation measure,” Evolution, vol. 59, no. 8, pp. 1633-1638, (2005).

Jost, L., “G ST and its relatives do not measure differentiation,” Molec- ular Ecology, vol. 17, no. 18, pp. 4015-4026, (2008).

Manly, F.J., “Randomization, bootstrap and Monte Carlo methods in biology”, Chapman and Hall, London, 1997.

Nei, M. and Chesser, R., “Estimation of fixation indices and gene diver- sities,” Ann. Hum. Genet, vol. 47, no. Pt 3, pp. 253-259, (1983).

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Revolution Analytics (2012). doParallel: Foreach parallel adaptor for the parallel package. R package version 1.0.1. http://CRAN.R-project.org/package=doParallel

Revolution Analytics (2012). foreach: Foreach looping construct for R. R package version 1.4.0. http://CRAN.R-project.org/package=foreach

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

Examples

```
## Not run:
# simply use the following format to run the function

test_result <- divPart(infile = 'mydata', outfile = "myresults',
                       gp = 3, pairwise = TRUE, bs_locus = TRUE,
                       bs_pairwise = TRUE, bootstraps = 1000,
                       plot = TRUE)

## End(Not run)
```

divRatio — *Calculates the standardised diversity ratios relative to a ’yardstick’ reference population following Skrbinsek et al., 2012.*

Description

Diversity ratios derived from allelic richness and expected heterozygosity are calculated from either a genepop file containing raw data for all populations of interest, or from a genepop file containing raw data for only the reference population and a data frame containing relevant information for the populations of interest. See below for more details the structure of this data frame.

Usage

```
divRatio(infile = NULL, outfile = NULL, gp = 3, pop_stats = NULL,
         refPos = NULL, boots = 1000, para = FALSE)
```

Arguments

`infile` A character string argument specifying the name of either a 3 digit or 2 digit genepop file containing the raw genotypes of at least the reference population sample.

`outfile` A character string specifying a prefix name for an automatically generated results folder, to which results file will be written.

`gp` Specifies the digit format of the `infile`. Either 3 (default) or 2.

`pop_stats` A character string indicating the name of the population statistics data frame file. This argument is required if only raw data for the reference population are give in `infile`. The data frame should be structured in a specific way. An example can be seen by typing `data(pop_stats)` into the console. The validloci column is only required if mean allelic richness and expected heterozygosity for populations of interest have been calculated from loci for which data is not present in the reference population. This column should contain a single character string of common loci between each population sample and the reference population sample.

`refPos` A numeric argument specifying the position of the reference population in `infile`. The argument is only valid when raw genotype data has been provided for the reference population sample and all other populations of interest and `pop_stats` is `NULL`.

`boots` Specifies the number of times the reference population should be resampled when calculating the sample size standardised allelic richness and expected heterozygosity for calculating the diversity ratios. The larger the number of bootstraps the longer the analysis will take to run. As an indication of runtime, running `divRatio` on the `Big_data` data set (type `?Big_data` for details), takes 10min 42s on a Toshiba Satellite R830 with 6GB RAM, and an Intel Core i5 - 2435M CPU running Linux.

`para` A logical argument indicating whether the analysis should make use of all available cores on the users system.

Details

All results will be written to a user defined folder, providing an argument is passed for 'outfile'. Results will be written to `.xlsx` files if the package `xlsx` and its dependencies are installed, or a `.txt` file otherwise.

Value

A data frame containing the following columns:

`pop` The names of each population of interest, including the reference population

`n` The sample size of each population

`alr` Mean allelic richness across loci

`alrSE` The standard error of the allelic richness across loci

`He` Mean expected heterozygosity across loci

`HeSE` Standard error of expected heterozygosity across loci

`alrRatio` The ratio of the allelic richness of the subject population sample and the sample size standardised reference population allelic richness

`alrSEratio` The standard error of divisions for the allelic richness ratio

`heRatio` The ratio of expected heterozygosity between the standardised reference population sample and subject population samples

`heSEratio` The standard error of divisions for the expected heterozygosity ratio

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Skrbinsek, T., Jelencic, M., Waits, L. P., Potocnik, H., Kos, I., & Trontelj, P. (2012). Using a reference population yardstick to calibrate and compare genetic diversity reported in different studies: an example from the brown bear. Heredity, 109(5), 299-305. doi:10.1038/hdy.2012.42

Examples

```
## Not run:
# To run an example use the following format

test_results <- divBasic(infile = Test_data, outfile = 'out',
                         gp = 3, pop_stats = NULL, refPos = NULL,
                         bootstraps = 1000, parallel = TRUE)

## End(Not run)
```

divSimCo *Similarity coefficients for co-dominant diploid genenotype data*

Description

Calculation of similarity coefficients for co-dominant genotype data, following Kosman and Leonard, (2005).

Usage

```
divSimCo(infile = NULL, loci = FALSE, boots = 0)
```

Arguments

infile	Specifying the name of the *'genepop'* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format. See http://genepop.curtin.edu.au/help_input.html for detail on the genepop file format.
loci	A logical argument, specifying whether similarity matrices for each locus should be calculated. If FALSE, only a global similarity matrix is given.
boots	An integer, specifying the number of bootstrapped matrices to generate. Loci are resampled with replacement.

Details

Similarity coefficients are calculated according to Kosman and Leonard, (2005). Multi-locus values are calculated from all loci without missing data.

Value

A list containing either one (global) if loci = FALSE, or two elements (global and loci) if loci = TRUE.

global	A pairwise matrix of multilocus similarity coefficients.
loci	A list of pairwise matrices of locus similarity coefficients.

Author(s)

Kevin Keenan

References

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

---

fastDivPart *Genetic differentiation statistics and their estimators*

Description

fastDivPart is identical to the divPart function in regards to what it calculates. The difference with this function is the speed with which it processes pairwise calculations. By using more efficient programming techniques, fastDivPart can execute commands up to 20X faster than divPart.

Usage

```
fastDivPart(infile = NULL, outfile = NULL, gp = 3, pairwise = FALSE,
            fst = FALSE, bs_locus = FALSE, bs_pairwise = FALSE,
            boots = 0, plot = FALSE, para = FALSE)
```

Arguments

`infile` Specifying the name of the *‘genepop’* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format, and must contain only one whitespace separator (e.g. “space” or “tab”) between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in “” or ‘’).

`outfile` Allows users to specify a prefix for an output folder. Name must a character string enclosed in either “” or ‘’.

`gp` Specifies the digit format of the `infile`. Either 3 (default) or 2.

`pairwise` A logical argument indicating whether standard pairwise diversity statistics should be calculated and returned as a diagonal matrix.

`fst` A Logical argument indicating whether Weir & Cockerham’s 1984 F-statistics should be calculated. NOTE - Calculating these statistics adds significant time to analysis when carrying out pairwise comparisons.

`bs_locus` Gives users the option to bootstrap locus statistics. Results will be written to `.xlsx` workbook by default if the package `'xlsx'` is installed, and to a `.html` file if `plot=TRUE`. If `'xlsx'` is not installed, results will be written to `.txt` files.

`bs_pairwise` Gives users the option to bootstrap statistics across all loci for each pairwise population comparison. Results will be written to a `.xlsx` file by default if the package `'xlsx'` is installed, and to a `.html` file if `plot=TRUE`. If `'xlsx'` is not installed, results will be written to `.txt` files.

`boots` Determines the number of bootstrap iterations to be carried out. The default value is `boots = 0`, this is only valid when all bootstrap options are false. There is no limit on the number of bootstrap iterations, however very large numbers of bootstrap iterations can take some time to run. If time is a limiting factor, the function `diffCalc` is up to 10x faster than `fastDivPart`, but does not allow users to write results to xlsx workbooks or plot results from the function. If required, these must be done manually.

`plot` Optional interactive `.html` image file of the plotted bootstrap results for loci if `bs_locus = TRUE` and pairwise population comparisons if `bs_pairwise = TRUE`. The default option is `plot = FALSE`.

`para` A logical input, indicating whether your analysis should be run in parallel mode or sequentially.

Details

All results will be written to a user defined folder ("working_directory/outfile"). The format of outputs will vary depending on the availability of the package `'xlsx'` in the users local package library. If `'xlsx'` is available, results will be written to an Excel workbook. If `'xlsx'` is not available, results will be written to `.txt` files. Multi-locus estimates of Weir and Cockerham’s theta are calculated as per Weir and Cockerham, 1984.

Value

`standard` A matrix containing identical data to the `Standard_stats` worksheet in the `.xlsx` workbook.

`estimate` A matrix containing identical data to the `Estimated_stats` worksheet in the `.xlsx` workbook.

`pairwise` A group of six matrices containing population pairwise statistics. This object is identical to that written as ‘pairwise-stats‘ in the `.xlsx` workbook.

`bs_locus` A list containing six matrices of locus values for *Gst*, *G’st*, *D(Jost)*, *Gst-(est)*, *G’st-(est)*, and *D(Jost)-(est)* along with their respective 95% confidence interval.

`bs_pairwise` A list containing three-four matrices (depending on whether Weir & Cockerham’s Fst is calculated) of pairwise values for *Gst*, *G’st*, *D(Jost)*, *Gst-(est)*, *G’st-(est)*, and *D(Jost)-(est)* along with their respective 95% confidence intervals, including bias corrected 95% confidence intervals.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Dragulescu, A.D., “xlsx: Read, write, formal Excel 2007 and Excel 97/2000/xp/2003 files”, R package version 0.4.2, url:http://CRAN.R-project.org/package=xlsx, (2012).

Guile, D.P., Shepherd, L.A., Sucheston, L., Bruno, A., and Manly, K.F., “sendplot: Tool for sending interactive plots with tool-tip content.”, R package version 3.8.10, url: http://CRAN.R-project.org/package=sendplot, (2012).

Hedrick, P., “A standardized genetic differentiation measure,” Evolution, vol. 59, no. 8, pp. 1633-1638, (2005).

Jost, L., “G ST and its relatives do not measure differentiation,” Molec- ular Ecology, vol. 17, no. 18, pp. 4015-4026, (2008).

Manly, F.J., “Randomization, bootstrap and Monte Carlo methods in biology”, Chapman and Hall, London, 1997.

Nei, M. and Chesser, R., “Estimation of fixation indices and gene diver- sities,” Ann. Hum. Genet, vol. 47, no. Pt 3, pp. 253-259, (1983).

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Revolution Analytics (2012). doParallel: Foreach parallel adaptor for the parallel package. R package version 1.0.1. http://CRAN.R-project.org/package=doParallel

Revolution Analytics (2012). foreach: Foreach looping construct for R. R package version 1.4.0. http://CRAN.R-project.org/package=foreach

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

Examples

```
## Not run:
# simply use the following format to run the function

test_result <- fastDivPart(infile = 'mydata', outfile = "myresults',
                           gp = 3, pairwise = TRUE, bs_locus = TRUE,
                           bs_pairwise = TRUE, boots = 1000,
                           plot = TRUE)

## End(Not run)
```

fstOnly

A minimal function for the calculate of Weir & Cockerham's (1984) F_ST and F_IT from codominant molecular data in the genepop format.

Description

This function calculates locus and pairwise confidence intervals for Weir & Cockerham's (1984) F_ST and F_IT. These statistics can also be calculated using the `divPart` function, however, `fstOnly` is designed to be more memory efficient for larger datasets (e.g. SNPs).

Usage

```
fstOnly(infile = NULL, outfile = NULL, gp = 3, bs_locus = FALSE,
        bs_pairwise = FALSE, bootstraps = 0, parallel = FALSE)
```

Arguments

`infile` Specifying the name of the *'genepop'* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format, and must contain only one whitespace separator (e.g. “space” or “tab”) between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the `working directory` the file path must be given. The name must be a character string (i.e. enclosed in “” or ‘’).

`outfile` Allows users to specify a prefix for an output folder. Name must a character string enclosed in either “” or ‘’.

`gp` Specifies the digit format of the `infile`. Either 3 (default) or 2.

`bs_locus` Specifies whether locus bootstrapped confidence intervals should be calculated.

`bs_pairwise` Specified whether pairwise population bootstrapped confidence intervals should be calculated.

`bootstraps` Determines the number of bootstrap iterations to be carried out. The default value is `bootstraps = 0`, this is only valid when all bootstrap options are false. There is no limit on the number of bootstrap iterations, however very large numbers of bootstrap iterations (< 1000) on even modest data sets (e.g. 265 individuals x 38 loci) will take over 20 minutes to run on a most PCs).

parallel	A logical input, indicating whether your analysis should be run in parallel mode or sequentially. `parallel = TRUE` is only valid if the packages, `parallel`, `doParallel` and `foreach` are installed.

Details

Because `fstOnly` is intended to maximise memory (RAM) efficiency, the function does not provide many of the plotting utilities that `divPart` does.

Value

locus	A list object containing two matrices, F_ST and F_IT. These matrices contain actual, lower 95% confidence interval and upper 95% confidence interval per locus. Global values are also presented with their respective confidence intervals.
pairwise	A list object containing two matrices, F_ST and F_IT. These matrices contain actual, lower 95% confidence interval and upper 95% confidence interval per pairwise population comparison.

Note

This function has become obsolete following improvements to `fastDivPart` and `diffCalc`. The use of either of these two functions is recommended over `fstOnly`. `fstOnly` will be deprecated in future `diveRsity` releases.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Weir, B.S. & Cockerham, C.C., Estimating F-Statistics, for the Analysis of Population Structure, Evolution, vol. 38, No. 6, pp. 1358-1370 (1984).

gpSampler *Randomly sample a genepop file*

Description

Randomly re-samples population samples from a genepop file for user defined sizes and returns a new genepop file containing the sub-sample data. The function can easily be integrated into simulation pipelines when exploring sample size effects.

Usage

```
gpSampler(infile = NULL, samp_size = 10, outfile = NULL)
```

Arguments

`infile` A character string pointing to a genepop input file. If the file exists in the current working directory, only the file name is required. If the file resides elsewhere, the entire file path is required.

`samp_size` This argument specifies the number of individuals to be randomly sampled from each population sample in the input file. Either a single numeric argument can be passed or a numeric vector of length equal to the number of population samples in the genepop file.

`outfile` A character string specifying the prefix name for the resulting file. This string will be suffixed with ".gen". The file will be written to the working directory, unless a properly formatted path is passed to `outfile`

Author(s)

Kevin Keenan

`haploDiv` *A function, allowing the calculation of Weir & Cockerham's (1984) F_ST from haploid genotypes in the genepop*

Description

haploDiv allows users to calculate Weir & Cockerham's F_{ST} from haploid genotypes for locus, global and pairwise population levels.

Usage

```
haploDiv(infile = NULL, outfile = NULL, pairwise = FALSE,
         boots = 0)
```

Arguments

`infile` A genepop file/data frame containing haploid genotypes. This file/data frame should contain locus information in either the two digit or three digit format. The argument can be a character string indicating the name of a file or a data frame in the R workspace (e.g. see `data("Test_data")`).

`outfile` A character string specifying a prefix to be added to output files. A character string specifying a directory location will result in the output files being written to the specified location, rather than the current working directory. If `outfile = NULL`, no results will be written to disk.

`pairwise` Specifies whether a population pairwise matrix containing F_{ST} values should be calculated.

`boots` Specifies whether bootstrapped 95% confidence intervals should be calculated for each pairwise estimate of F_{ST}. If `bootstraps = 0` and `pairwise = TRUE`, only a pairwise matrix of F_{ST} will be returned.

Details

This function uses the same fundamental algorithms as `divPart` and `fastDivPart`, the only difference being that if *diploidizes* haploid genotypes before calculating statistics. The diploidization process has the effect of changing a haploid genotype into a homozygous diploid genotype for all individuals.

Value

`locus` A named vector of locus estimates of Weir & Cockerham's F_{ST} across all population samples.

`overall` A global estimate of Weir & Cockerham's F_{ST}.

`pairwise` A diagonal matrix containing pairwise estimates of Weir & Cockerham's F_{ST} across all loci. Returned when `pairwise = TRUE`.

`bs_pairwise` A data frame with three data columns containing bootstrapped mean, lower 95% confidence limit and upper 95% confidence limit for each population pair (rows). Returned when `bootstraps > 0` and `pairwise = TRUE`.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

---

inCalc *A function to calculate locus informative for the inference of ancestry*

Description

`inCalc` allows the calculation of locus informativeness for ancestry (*In*), (Rosenberg *et al.,* 2003), both across all population samples under consideration and for all pairwise combinations of population samples. These data can be bootstrapped using the same procedure as above, to obtain 95% confidence intervals.

Usage

```
inCalc(infile = NULL, outfile = NULL, pairwise = FALSE, xlsx = FALSE,
       boots = NULL, para = FALSE)
```

Arguments

`infile` Specifying the name of the *‘genepop’* (Rousset, 2008) file from which the statistics are to be calculated This file can be in either the 3 digit of 2 digit format. See `http://genepop.curtin.edu.au/help_input.html` for detail on the genepop file format.

`outfile` Allows users to specify a prefix for an output folder. Name must a character string enclosed in either “” or ‘’.

`pairwise` Specified whether pairwise I_n should be calculated.

`xlsx` A logical argument indicating whether results should be written to an *xlsx* file. If `xlsx = FALSE` (default), results will be written to text files.

`boots` Determines the number of bootstrap iterations to be carried out. The default value is `boots = 0`, this is only valid when all bootstrap options are false.

`para` Allows for parallel computation of pairwise locus *In*. The number of available core is automatically detected if `para = TRUE`.

Details

All results will be written to a user defined folder ("working_directory/outfile"). The format of outputs will vary depending value of the `xlsx` argument. If `xlsx = TRUE`, results will be written to a .xlsx workbook using the `xlsx` package. If `xlsx = FALSE`, results will be written to `.txt` files.

Value

`inCalc` return a list object to the R workspace, with elements described below. In addition to this results can be optionally written to file using the `outfile` argument. If `xlsx = TRUE` results will be written to a multi-sheet xlsx file. If `xlsx = FALSE` results are written to multiple text file, the number of which depends on the function arguments used.

`global` A `data.frame` containing the *In* values for each locus, calculated across all samples in `infile`. If `boots` is an integer greater than 0, this data.frame will also contain lower and upper 95% confidence limits for each locus.

`pairwise` A `data.frame` containing the pairwise locus *In* values for all possible pairwise population comparisons. This object is returned when `boots` is an integer greater than 0.

`lower_CI` If `pairwise = TRUE` and `boots` is an integer greater than 0, `lower_CI` is returned. It is a `data.frame` containing the lower 95% confidence limit for the corresponding pairwise estimate in the `pairwise` data.

`upper_CI` If `pairwise = TRUE` and `boots` is an integer greater than 0, `upper_CI` is returned. It is a `data.frame` containing the upper 95% confidence limit for the corresponding pairwise estimate in the `pairwise` data.

Note

Since version 1.9.0, the speed of this function has been greatly improved. Users can expect up to x10 speed up on previous versions. The output data structure is also slightly different from v1.9.0 onwards.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Dragulescu, A.D., “xlsx: Read, write, formal Excel 2007 and Excel 97/2000/xp/2003 files”, R package version 0.4.2, url:http://CRAN.R-project.org/package=xlsx, (2012).

Manly, F.J., “Randomization, bootstrap and Monte Carlo methods in biology”, Chapman and Hall, London, 1997.

Rosenberg, N., Li, L., Ward, R., and Pritchard, J., “Informativeness of genetic markers for inference of ancestry.,” American Journal of Human Genetics, vol. 73, no. 6, pp. 1402-22, (2003).

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Examples

```
## Not run:
# To run an example use the following format
library(diveRsity)
data(Test_data)
Test_data[is.na(Test_data)] <- ""

test_results<-inCalc(infile = Test_data, outfile = 'out', pairwise = TRUE,
                     xlsx = FALSE, boots = 1000, para = TRUE)

## End(Not run)
```

microPlexer *Launches a* shiny *app for the arrangement of microsatellite loci into size and fluorophore based multiplex groups.*

Description

This function will launch a browser based app designed using the package, shiny, which allows users to design microsatellite multiplex groups.

Usage

```
microPlexer()
```

Details

The application provides flexibility in marker organisation through the use of two distinct algorithms. The *high-throughput* algorithm will attempt to group as many loci into as few multiplex groups as allowable based on locus size and the available fluorophore labels, while the *balanced* algorithm attempts to organise loci into multiplex group of roughly equal density, so as to offset possible primer interactions etc. As input, the application accepts a .csv file with three named columns. The structure of this file is as follows:

nms - contains the names of loci

lower - contains the lower size range of loci

upper - contains the upper size range of loci

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

`polyIn`	*A function for calculating informativeness for the inference of ancestry from loci of any ploidy.*

Description

This function will calculate Rosenberg et al's, (2003) In for loci of any ploidy level. The statistic can be calculated across all samples or on a pairwise basis. The function is efficient for a large number of loci.

Usage

```
polyIn(infile = NULL, pairwise = FALSE, para = FALSE)
```

Arguments

`infile`	A character string pointing to an input file containing genotypes in a modified genepop format. Rather that genotypes being coded as either two digit or three digit numbers, polyploid genotypes are coded as `AABA, AAAA, ... ABAB` etc. for a tetraploid or `ABA, AAA, ... ABB` etc. for a triploid. The overall structure of the input file should be that of a genepop file.
`pairwise`	A logical argument specifying whether In should be calculated for pairwise population comparisons. Large number of population samples will result in long computation times.
`para`	A logical argument specifying whether multiple CPU cores should be used. If `para = TRUE`, all available system CPUs will be used.

Value

If `pairwise = TRUE`, a list of pairwise matrices for each locus is returned. These list elements are named as per loci names in `infile` to allow simple indexing for loci of interest. If `pairwise = FALSE` and named numeric vector is returned.

Author(s)

Kevin Keenan 2014

References

Rosenberg, N., Li, L., Ward, R., and Pritchard, J., (2003) 'Informativeness of genetic markers for inference of of ancestry.', American Journal of Human Genetics, vol. 73, no. 6 pp. 1402-22.

pop_stats — *Example pop_stats data frame for use in the function* divRatio

Description

pop_stats is an example data frame for the function divRatio. Users wishing to pass a value to the argument pop_stats in this function should ensure that the format is as given in this data set. Header names should be identical to those presented in the pop_stats data set. The inclusion of the column named validloci is optional.

Usage

```
pop_data
```

Format

data frame

readGenepop — *A function to calculate allele frequencies from genepop files.*

Description

readGenepop allows the calculation of various parameters from 3 digit and 2 digit genepop files. The purpose of the function is mainly as a data manipulation process to allow for easy downstream analysis.

Usage

```
readGenepop(infile = NULL, gp = 3, bootstrap = FALSE)
```

Arguments

infile	Specifies the name of the *'genepop'*(Rousset, 2008) file from which the statistics are to be calculated. This file can be in either the 3 digit of 2 digit format, and must contain only one *whitespace* separator (e.g. “space" or “tab") between each column including the individual names column. The number of columns must be equal to the number of loci + 1 (the individual names column). If this file is not in the working directory the file path must be given. The name must be a character string (i.e. enclosed in “" or ‘’).
gp	A numeric argument specifying the format of the infile. Either ‘3’ or ‘2’ are accepted as arguments. Default is gp = 3.
bootstrap	A logical argument specifying whether the user would like the infile data bootstrapped. If bootstrap = TRUE a genepop format object is returned. See bootstrap_file in the value section below.

Details

Results from this function allow for the calculation of various population genetics statistics, such as those calculated by `div.part` and `in.calc`. Users may find it useful for data exploration. For instance by employing the `plot {graphics}` function, an *'ad hoc'* assessment of allele size distribution can be carried out using the code in the example section below. From this example it is clear that the function will be particularly useful for those wishing to develop their own novel analysis methods.

Value

`npops` The number of population samples in `infile`.

`nloci` The number of loci in `infile`.

`pop_alleles` A list of matrices (n = 2 x `npops`) containing haploid allele designations. Every two matrices contain the two alleles per individual per population. For example `pop_alleles[[1]][1,1]` and `pop_alleles[[2]][1,1]` are the two alleles observed in individual '1' in population '1' at locus '1', whereas `pop_alleles[[3]][1,1]` and `pop_alleles[[4]][1,1]` are the two alleles observed in individual '1' in population '2' at locus '1'.

`pop_list` A list of matrices (n = `npops`) containing the diploid genotypes of individuals per locus.

`loci_lames` A character vector containing the names of loci from `infile`.

`pop_pos` A numeric vector or the row index locations of the first individual per population in `infile`.

`pop_sizes` A numeric vector of length `npops` containing the number of individuals per population sample in `infile`.

`allele_names` A list of `npops` lists containing `nloci` character vectors of alleles names per locus. Useful for identifying unique alleles.

`all_alleles` A list of `nloci` character vectors of all alleles observed across all population samples in `infile`.

`allele_freq` A list containing `nloci` matrices containing allele frequencies per alleles per population sample.

`raw_data` An unaltered data frame of `infile`.

`loci_harm_N` A numeric vector of length `nloci`, containing the harmonic mean number of individuals genotyped per locus.

`n_harmonic` A numeric value representing the harmonic mean of `npops`.

`pop_names` A character vector containing a four letter population sample name for each population in `infile` (the first four letter of the first individual).

`indtyp` A list of length `nloci` containing character vectors of length `npops`, indicating the number of individuals per population sample typed at each locus.

`nalleles` A vector of the total number of alleles observed at each locus.

`bs_file` A genepop format data frame of bootstrapped `infile`. This value is only returned if `bootstrap = TRUE`.

Author(s)

Kevin Keenan <kkeenan02@qub.ac.uk>

References

Rousset, F., “genepop’007: a complete re-implementation of the genepop software for Windows and Linux.,” Molecular ecology resources, vol. 8, no. 1, pp. 103-6, (2008).

Examples

```
# Code to plot ordered allele fragment sizes to assess mutation model.
data(Test_data, package = "diveRsity") # define data
x <- readGenepop(infile = Test_data, gp = 3, bootstrap = FALSE)
locus10_pop1 <- c(x$pop_alleles[[1]][[2]][,10],
                  x$pop_alleles[[1]][[2]][,10])
sort_order <- order(locus10_pop1, decreasing = FALSE) #sort alleles
plot(locus10_pop1[sort_order], col="red", ylab = "Allele size")
```

snp2gen *Conversion function for SNP nucleotide genotype matrix to a genepop file.*

Description

This function converts SNP nucleotide genotype to genepop file format.

Usage

```
snp2gen(infile = NULL, prefix_length = 2)
```

Arguments

`infile`	A character string indicating the name of the text file containing SNP genotypes.
`prefix_length`	This argument specifies the population specific prefix within individual names.

Value

A genepop file written to disk as "snp2gen-converted.gen"

Author(s)

Kevin Keenan, 2014

Examples

```
## Not run:
data(SNPs, package = "diveRsity")
snp2gen(infile = SNPs, prefix_length = 2)

## End(Not run)
```

SNPs *Example input format for* snp2gp *function*

Description

SNPs depicts a typical input layout for the function snp2gp. Data can be passed to this function as either a dataframe, or a file name for a tab delimited file with the same structure as SNPs.

Usage

```
SNPs
```

Format

SNP data

Test_data *Example infile for diveRsity package*

Description

Test_data is a typical genepop (Rousset, 2008) three digit format input file for the package diveRsity. This is empirical data for six brown trout population first presented in 'Beaufort Trout MicroPlex: A high throughput multiplex platform comprising 38 informative microsatellite loci for use in brown trout and sea trout (Salmo trutta L.) genetics studies. Journal of Fish Biology (2013)'. The data file contains genotypic information for 37 microsatellite loci from 265 individuals.

Usage

```
Test_data
```

Format

genepop 3 digit.

References

Rousset, F., "genepop'007: a complete re-implementation of the genepop software for Windows and Linux.," Molecular Ecology Resources, vol. 8, no. 1, pp. 103-6, (2008).

Index