Package ‘genepop’

January 20, 2023

Type Package

Title Population Genetic Data Analysis Using Genepop

Version 1.2.2

Date 2023-01-19

Maintainer François Rousset <francois.rousset@umontpellier.fr>

Description Makes the Genepop software available in R. This software implements a mixture of traditional population genetic methods and some more focused developments: it computes exact tests for Hardy-Weinberg equilibrium, for population differentiation and for genotypic disequilibrium among pairs of loci; it computes estimates of F-statistics, null allele frequencies, allele size-based statistics for microsatellites, etc.; and it performs analyses of isolation by distance from pairwise comparisons of individuals or population samples.

License CeCILL-2

Encoding UTF-8

Imports Rcpp (>= 0.12.10), stringr

LinkingTo Rcpp, RcppProgress

Suggests testthat, knitr, shiny

Config/build/clean-inst-doc false

RoxygenNote 7.2.3

ByteCompile true

URL https://www.r-project.org,
https://kimura.univ-montp2.fr/~rousset/Genepop.htm

NeedsCompilation yes

Author François Rousset [aut, cre, cph]
(<https://orcid.org/0000-0003-4670-0371>),
Jimmy Lopez [ctb],
Alexandre Genin [ctb],
Khalid Belkhir [ctb]

Repository CRAN

Date/Publication 2023-01-19 23:30:17 UTC

R topics documented:

genepop-package . . . 2
basic_info . . . 4
clean_workdir . . . 4
Contingency-test . . . 5
conversion . . . 6
Differentiation . . . 7
Fst . . . 8
genedivFis . . . 9
genepopExample . . . 10
GUI . . . 10
Hardy-Weinberg . . . 11
IBD . . . 12
Linkage . . . 14
manipulation . . . 15
Nm_private . . . 16
nulls . . . 16
setMantelSeed . . . 17
setRandomSeed . . . 18
set_restriction . . . 18

Index **19**

---

genepop-package *Population genetic analyses using the Genepop software*

Description

A distribution of the Genepop software as an R package. The included C++ sources are suitable for compilation as a stand-alone executable. A shiny interface is also included. Genepop performs three main tasks: it computes **exact tests** for Hardy-Weinberg equilibrium (`test_HW`), for population differentiation (`test_diff`) and for genotypic disequilibrium among pairs of loci (`test_LD`); it computes **estimates** of F-statistics (`Fst`), null allele frequencies (`nulls`), allele size-based statistics for microsatellites, etc., and of number of immigrants by Barton & Slatkin's (1986) private allele method (`Nm_private`); It performs analyses of **isolation by distance** from pairwise comparisons of individuals or groups (`ibd`), including confidence intervals for "neighborhood size". It also provides various data `conversion` and `manipulation` utilities.

Author(s)

R package originally developed by Jimmy Lopez and Khalid Belkhir from the C++ sources of the Genepop executable version 4.6 (2016; Rousset 2008).

References

Main reference for current maintained version of Genepop:

Rousset, F. (2008). Genepop'007: a complete reimplementation of the Genepop software for Windows and Linux. Mol. Ecol. Res. 8: 103-106.

Original Genepop publication:

Raymond, M. & Rousset, F., 1995b. GENEPOP Version 1.2: population genetics software for exact tests and ecumenicism. J. Hered. 86: 248-249.

Methods implemented in Genepop:

Barton, N. H. & Slatkin, M., 1986. A quasi-equilibrium theory of the distribution of rare alleles in a subdivided population. Heredity 56: 409-415.

Brookfield, J. F. Y., 1996. A simple new method for estimating null allele frequency from heterozygote deficiency. Mol. Ecol. 5: 453-455.

Goudet, J., Raymond, M., de Meeus, T. & Rousset, F., 1996. Testing differentiation in diploid populations. Genetics 144: 1931-1938.

Guo, S. W. & Thompson, E. A., 1992. Performing the exact test of Hardy-Weinberg proportion for multiple alleles. Biometrics 48: 361-372.

Kalinowski, S. T. & Taper, M. L., 2006. Maximum likelihood estimation of the frequency of null alleles at microsatellite loci. Conserv. Genetics 7:991-995.

Louis, E. J. & Dempster, E. R., 1987. An exact test for Hardy-Weinberg and multiple alleles. Biometrics 43: 805-811.

Mantel, N., 1967. The detection of disease clustering and a generalized regression approach. Cancer Research 27: 209-220.

Michalakis, Y. & Excoffier, L., 1996. A generic estimation of population subdivision using distances between alleles with special interest to microsatellite loci. Genetics 142: 1061-1064.

Raymond, M. & Rousset, F., 1995a. An exact test for population differentiation.Evolution 49: 1283-1286.

Robertson, A. & Hill, W. G., 1984. Deviations from Hardy-Weinberg proportions: sampling variances and use in estimation of inbreeding coefficients. Genetics 107: 703-718.

Rousset, F., 1996. Equilibrium values of measures of population subdivision for stepwise mutation processes. Genetics 142: 1357-1362.

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

Rousset, F., 2000. Genetic differentiation between individuals. J. Evol. Biol. 13:58-62.

Rousset, F. & Raymond, M., 1995. Testing heterozygote excess and deficiency. Genetics 140: 1413-1419.

Watts, P. C., Rousset, F., Saccheri, I. J., Leblois, R., Kemp, S. J. & Thompson, D. J., 2007. Compatible genetic and ecological estimates of dispersal rates in insect (Coenagrion mercuriale: Odonata: Zygoptera) populations: analysis of 'neighbourhood size' using a more precise estimator. Mol. Ecol. 16: 737-751.

Weir, B. S., 1996. Genetic Data Analysis II. Sinauer, Sunderland, Mass.

Weir, B. S. & Cockerham, C. C., 1984. Estimating F-statistics for the analysis of population structure. Evolution 38: 1358-1370.

basic_info — *Allele and genotype frequencies*

Description

Allele and genotype frequencies per locus and per sample. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
basic_info(inputFile, outputFile = "", verbose = interactive())
```

Arguments

inputFile	The path of the input file, in Genepop format
outputFile	character: The path of the output file
verbose	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
basic_info(locinfile,'sample.txt.INF')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

clean_workdir — *Removing files created by Genepop*

Description

This removes "temporary files" created by Genepop, but also output files, so it should be used only when one no longer needs the latter files. This function asumes that the input file name contains only alphanumeric, dot, or underscore characters.

Usage

```
clean_workdir(
  otherfiles = NULL,
  path = ".",
 suffixes = c("GRA", "ISO", "MIG", "PRI", "DAT", "DG", "DIV", "D", "DIS", "FST", "NUL",
    "RHO", "2G2", "G", "GE", "GE2", "INF", "MSD", "TAB", "ST2"),
  in. = TRUE,
  cmdline = TRUE
)
```

Arguments

`otherfiles`	Character vector(s): one or more names of files to be removed and not matched by the other arguments (such as the input file, or some output files not identified by their suffix, as shown in the Example).
`path`	character vector: path from where files should be removed.
`suffixes`	Character vector(s): suffixes of files to be removed (useful for output files with readily identifiable suffixes).
`in.`	boolean: whether to remove the `fichier.in` file created by Genepop.
`cmdline`	boolean: whether to remove the `cmdline.txt` file created by Genepop.

Examples

```
# Removing files possibly written by other examples in the documentation:
clean_workdir(otherfiles=c("sample.txt", "Dsample.txt", "w2.txt",
"PEL1600withCoord.txt", "Rhesus.txt", "structest.txt"))
```

`Contingency-test` *Exact test on a single contingency table*

Description

Performs an exact conditional contingency-table test. There are many other ways of doing this in R but this function replicates the functionality of earlier genepop code analysing a contingency table provided in a file with ad hoc format. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
struc(
  inputFile,
  settingsFile = "",
  dememorization = 10000,
  batches = 100,
  iterations = 5000,
  verbose = interactive()
)
```

Arguments

`inputFile`	character: The path of the input file. This file should be in an ad hoc format
`settingsFile`	character: The path of the settings file
`dememorization`	integer: length of dememorization step of Markov chain algorithm
`batches`	integer: Number of batches
`iterations`	integer: Iterations per batch
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('structest.txt')
struc(locinfile)
if ( ! interactive()) clean_workdir(otherfiles='structest.txt')
```

---

conversion *File conversions*

Description

Converts input files from genepop format to some other formats (some maybe only of historical interest): Fstat, two Biosys formats. and linkdos. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
conversion(inputFile, format, outputFile = "", verbose = interactive())
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`format`	Character string: must be one of `'Fstat'`, `'BiosysL'`, `'BiosysN'`, or `'Linkdos'`
`outputFile`	character: The path of the output file
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
conversion(locinfile, format='Fstat', 'sample.txt.DAT')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

`Differentiation` *Tests of genic and genotypic differentiation*

Description

Exact conditional contingency-table tests for genic or genotypic differentiation. A single test for all populations, or distinct tests for all pairs of populations, may be computed. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
test_diff(
  inputFile,
  genic = TRUE,
  pairs = FALSE,
  outputFile = "",
  settingsFile = "",
  dememorization = 10000,
  batches = 100,
  iterations = 5000,
  verbose = interactive()
)
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`genic`	logical: whether to perform genic or genotypic tests
`pairs`	logical: whether to test differentiation between all pairs of ppulation, or to perform a single global test
`outputFile`	character: The path of the output file
`settingsFile`	character: The path of the settings file
`dememorization`	integer: length of dememorization step of Markov chain algorithm
`batches`	integer: Number of batches
`iterations`	integer: Iterations per batch
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
test_diff(locinfile,outputFile='sample.txt.GE')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

`Fst` *Fst (or rho_ST) estimation*

Description

Evaluates Fst or related measures based on allele sizes, for all populations of for all pairs of populations. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
Fst(
  inputFile,
  sizes = FALSE,
  pairs = FALSE,
  outputFile = "",
  dataType = "Diploid",
  verbose = interactive()
)
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`sizes`	logical: whether to estimate allele-size based statistics, or identity-based Fst
`pairs`	whether to estimate differentiation between all pairs of populations, or to compute a global estimate for all populations
`outputFile`	character: The path of the output file
`dataType`	character: The haploid and diploid data
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
Fst(locinfile, outputFile= 'sample.txt.DIV')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

genedivFis *Gene diversities and Fis (or rho_IS)*

Description

Evaluates Fis and gene diversities, or related measures based on allele sizes. See this section of the Genepop executable documentation for more information on the identity-based statistical methods, and this one for allele-size based ones.

Usage

```
genedivFis(
  inputFile,
  sizes = FALSE,
  outputFile = "",
  dataType = "Diploid",
  verbose = interactive()
)
```

Arguments

inputFile	The path of the input file, in Genepop format
sizes	logical: whether to compute statistics based on allele size, or not.
outputFile	character: The path of the output file
dataType	character: The haploid and diploid data
verbose	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
genedivFis(locinfile,outputFile = 'sample.txt.DIV')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

genepopExample — *Copy an example file from the Genepop package distribution*

Description

This function is used to copy an example file to the user's directory. It should not be used when analysing one's own data!

Usage

```
genepopExample(filename)
```

Arguments

`filename`	The name of an example file from the Genepop distribution.

Value

Returns the `filename`

GUI — *Call an experimental GUI for Genepop*

Description

Call an experimental GUI for Genepop

Usage

```
GUI()
```

Value

The return value of a ‘shiny::runApp()‘ call.

Hardy-Weinberg *Tests of Hardy-Weinberg genotypic proportions*

Description

Compute variants of the exact conditional test for Hardy-Weinberg genotypic proportions. The tests differ by their test statistics. `HWtable_analysis` handles a single table of genotype counts, and `test_HW` requires a standard genepop input file. See this section of the Genepop-executable documentation for more information on the statistical methods.

Usage

```
test_HW(
  inputFile,
  which = "Proba",
  outputFile = "",
  settingsFile = "",
  enumeration = FALSE,
  dememorization = 10000,
  batches = 20,
  iterations = 5000,
  verbose = interactive()
)

HWtable_analysis(
  inputFile,
  which = "Proba",
  settingsFile = "",
  enumeration = FALSE,
  dememorization = 10000,
  batches = 20,
  iterations = 5000,
  verbose = interactive()
)
```

Arguments

`inputFile`	character: The path of the input file. For `test_HW`, this file should be in Genepop format. For `HWtable_analysis`, it should be in ad hoc format illustrated by sample file `Rhesus.txt` used in the Examples section, and further detailed in this section of the Genepop-executable documentation.
`which`	character: `'Proba'`, `'excess'`, and `'deficit'` to perform the probability test, score test for excess, and score tests for deficit, respectively, in each population and for each locus. `test_HW` additionally handles `'global excess'` and `'global deficit'` for global tests for all loci and/or all populations, and `HWtable_analysis` additionally handles `'Fis'` to report basic information (allele frequencies and Fis).

`outputFile`	character: The path of the output file
`settingsFile`	character: The path of the settings file
`enumeration`	logical: whether to compute the complete enumeration test for samples with less than 5 alleles
`dememorization`	integer: length of dememorization step of Markov chain algorithm
`batches`	integer: Number of batches
`iterations`	integer: Iterations per batch
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
test_HW(locinfile, which='deficit', 'sample.txt.D')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
# Example in Guo & Thompson 1992 Table 5
locinfile <- genepopExample('Rhesus.txt')
outfile <- HWtable_analysis(locinfile,which='Proba',batches = 1000,iterations = 1000)
readLines(outfile)[21]
#clean_workdir(otherfiles='Rhesus.txt')
```

IBD *Isolation by distance*

Description

Estimates isolation by distance by regression of genetic distance to geographical distance. See this section of the Genepop executable documentation for more information on individual-based analyses and this one for group-based analyses.

Usage

```
ibd(
  inputFile,
  outputFile = "",
  settingsFile = "",
  dataType = "Diploid",
  statistic = "F/(1-F)",
  geographicScale = "2D",
  CIcoverage = 0.95,
  testPoint = 0,
  minimalDistance = 1e-04,
  maximalDistance = 1e+09,
```

```
  mantelPermutations = 1000,
  mantelRankTest = FALSE,
  bootstrapMethod = "ABC",
  bootstrapNsim = 999,
  verbose = interactive()
)
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`outputFile`	character: The path of the output file
`settingsFile`	character: The path of the settings file
`dataType`	character: `'haploid'` or `'diploid'`
`statistic`	character: The pairwise genetic distance, either `'a'` or `'e'` for diploid individual data, `'a-like'` for haploid individual data, and `'F/(1-F)'` or `'SingleGeneDiv'` for group data (haploid or diploid)
`geographicScale`	character: gives either the scale transformation `'Log'` or `'Linear'` for geographic distances, or the shape of the habitat `'2D'` or `'1D'`
`CIcoverage`	numeric: The coverage probability of confidence intervals
`testPoint`	numeric: Given value of the slope to be tested
`minimalDistance`	numeric: The minimal geographic distance
`maximalDistance`	numeric: The maximal geographic distance
`mantelPermutations`	numeric: The number of permutations may be specified
`mantelRankTest`	logical: whether to use ranks in the Mantel test
`bootstrapMethod`	character: which bootstrap method to use (one of "ABC", "BC" or "BCa").
`bootstrapNsim`	integer: the number of bootstrap simulations to use (has no effect if method is "ABC").
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
## Not run:
locinfile <- genepopExample('w2.txt')
outfile <- ibd(locinfile,'w2.txt.ISO', geographicScale = 'Log', statistic='e')
if ( ! interactive()) clean_workdir(otherfiles='w2.txt')

locinfile <- genepopExample('PEL1600withCoord.txt')
```

```
outfile <- ibd(locinfile,'PEL1600withCoord.ISO', statistic = 'SingleGeneDiv',
               geographicScale = '1D')
if ( ! interactive()) clean_workdir(otherfiles='PEL1600withCoord.txt')

## End(Not run)
```

Linkage *Tables and exact test for genotypic linkage disequilibrium*

Description

Exact test for each pair of loci in each population. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
test_LD(
  inputFile,
  outputFile = "",
  settingsFile = "",
  dememorization = 10000,
  batches = 100,
  iterations = 5000,
  verbose = interactive()
)

write_LD_tables(inputFile, outputFile = "", verbose = interactive())
```

Arguments

inputFile	The path of the input file, in Genepop format
outputFile	character: The path of the output file
settingsFile	character: The path of the settings file
dememorization	integer: length of dememorization step of Markov chain algorithm
batches	integer: Number of batches
iterations	integer: Iterations per batch
verbose	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
## Not run:  # 'dontrun' only because a bit too slow for CRAN checks
locinfile <- genepopExample('sample.txt')
test_LD(locinfile,'sample.txt.DIS')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')

## End(Not run)
locinfile <- genepopExample('sample.txt')
write_LD_tables(locinfile,'sample.txt.TAB')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

---

manipulation *Various data manipulation utilities*

Description

Various procedures described in the linked sections of the Genepop executable documentation: diploidize haploid data, relabel_alleles, sample_haploid, and pop_to_indiv. The latter procedure converts population samples (several individuals in each population) to individual data. The names given to the individuals in the new file created (names which are to be interpreted as coordinates in a spatial analysis) may be the population coordinates (given as the name of the last individual in the original data file), or each individual's coordinates (given as the name of each individual in the original data file).

Usage

```
diploidize(inputFile, outputFile = "", verbose = interactive())

relabel_alleles(inputFile, outputFile = "", verbose = interactive())

pop_to_indiv(inputFile, coordinates, outputFile = "", verbose = interactive())

sample_haploid(inputFile, outputFile = "", verbose = interactive())
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`outputFile`	character: The path of the output file
`verbose`	logical: whether to print some information
`coordinates`	character: either `'population'` (to use population coordinates) or any other charater string (to use individual coordinates).

Examples

```
locinfile <- genepopExample('sample.txt')
outfile <- diploidize(inputFile = locinfile,outputFile="Dsample.txt")
if ( ! interactive()) clean_workdir(c("sample.txt", "Dsample.txt"))
```

Nm_private — *Private allele method*

Description

Estimation of Nm by private allele method of Slatkin and Barton. See this section of the Genepop executable documentation for more information on the statistical methods.

Usage

```
Nm_private(
  inputFile,
  outputFile = "",
  dataType = "Diploid",
  verbose = interactive()
)
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`outputFile`	character: The path of the output file
`dataType`	character: The haploid and diploid data
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

Examples

```
locinfile <- genepopExample('sample.txt')
Nm_private(locinfile,'sample.txt.PRI')
if ( ! interactive()) clean_workdir(otherfiles='sample.txt')
```

nulls — *Estimation of allele frequencies under genotyping failure.*

Description

Estimates allele frequencies (and failure rate if relevant) under dfferent assumptions: maximum likelihood assuming that there is null allele (default method), maximum likelihood assuming that apparent nulls are technical failures independent of genotype (`'ApparentNulls'`), and Brookfield's (1996) estimator (`'B96'`). See this section of the Genepop executable documentation for more information on the statistical methods. Genepop takes the allele with the highest number for a given locus across all populations as the null allele. For example, if you have 4 alleles plus a null allele, a null homozygote individual should be indicated as e.g. `0505` or `9999` in the input file.

Usage

```
nulls(
  inputFile,
  outputFile = "",
  settingsFile = "",
  nullAlleleMethod = "",
  CIcoverage = 0.95,
  verbose = interactive()
)
```

Arguments

`inputFile`	The path of the input file, in Genepop format
`outputFile`	character: The path of the output file
`settingsFile`	character: The path of the settings file
`nullAlleleMethod`	character: `'ApparentNulls'`, `'B96'` or anything else (default method).
`CIcoverage`	numeric: The coverage probability of confidence interval
`verbose`	logical: whether to print some information

Value

The path of the output file is returned invisibly.

`setMantelSeed` *Set random generator seed for Mantel test*

Description

Set random generator seed for Mantel test

Usage

```
setMantelSeed(seed)
```

Arguments

`seed`	integer: the new seed

setRandomSeed *Set random generator seed (except for Mantel test)*

Description

Set random generator seed (except for Mantel test)

Usage

```
setRandomSeed(seed)
```

Arguments

seed	integer: the new seed

set_restriction *Programming utilities*

Description

`getVersion` returns the version number of the C++ code (the same number that identifies the C++ executable). `set_restriction(TRUE)` sets the maximum number of populations and of loci to 300.

Usage

```
set_restriction(set = FALSE)

getVersion()
```

Arguments

set	logical: whether to set restrictions on number of populations and of loci

Index

∗ **package**
genepop-package, 2

basic_info, 4

clean_workdir, 4
Contingency-test, 5
conversion, *2*, 6

Differentiation, 7
diploidize *(*manipulation*)*, 15

Fst, *2*, 8

genedivFis, 9
genepop *(*genepop-package*)*, 2
genepop-package, 2
genepop-utils *(*set_restriction*)*, 18
genepopExample, 10
getVersion *(*set_restriction*)*, 18
GUI, 10

Hardy-Weinberg, 11
HWtable_analysis *(*Hardy-Weinberg*)*, 11

IBD, 12
ibd, *2*
ibd *(*IBD*)*, 12

Linkage, 14

manipulation, *2*, 15

Nm_private, *2*, 16
nulls, *2*, 16

pop_to_indiv *(*manipulation*)*, 15

relabel_alleles *(*manipulation*)*, 15

sample_haploid *(*manipulation*)*, 15
set_restriction, 18
setMantelSeed, 17
setRandomSeed, 18
struc *(*Contingency-test*)*, 5

test_diff, *2*
test_diff *(*Differentiation*)*, 7
test_HW, *2*
test_HW *(*Hardy-Weinberg*)*, 11
test_LD, *2*
test_LD *(*Linkage*)*, 14

write_LD_tables *(*Linkage*)*, 14