

Package ‘popgen’

February 25, 2013

Version 0.2

Date 2013-2-25

Title Population genetic simulations & numerical analysis

Author Liam J. Revell

Maintainer Liam J. Revell <liam.revell@umb.edu>

Depends R (>= 2.6)

Imports

Suggests

ZipData no

Description popgen conducts various numerical analyses and simulations in population genetics and evolutionary theory.

License GPL (>= 2)

URL <http://faculty.umb.edu/liam.revell/>

Repository

Date/Publication 2013-2-25 21:00:00 EST

R topics documented:

founder.event	2
freqdep	3
genetic.drift	4
hawk.dove	5
mutation.selection	5
rcd	6
selection	7
sexratio	8

Index	10
--------------	-----------

founder.event	<i>Simulation of a founder event</i>
---------------	--------------------------------------

Description

This function simulates genetic drift with a founding event at time `etime`.

Usage

```
founder.event(p0=0.5,Ne=1000,Nf=10,itime=100,etime=50,show="p")
```

Arguments

<code>p0</code>	Starting frequency for the A allele.
<code>Ne</code>	Effective population size at the start of the simulation and after the founding event.
<code>Nf</code>	Size of the founding population.
<code>itime</code>	Total time of the simulation.
<code>etime</code>	Time for the founding event.
<code>show</code>	Two different options for plotting. "p" shows the frequency of the A allele through time; "var" shows the genetic variation in the population, calculated as $p*(1-p)$. The default is <code>show="p"</code> .

Value

The function creates one of two different plots, depending on the value of `show`.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[genetic.drift](#)

Examples

```
founder.event()  
founder.event(show="variation")
```

Description

This function performs numerical analysis of a frequency dependent selection model based on Rice (*Evolutionary Theory*). The fitnesses of the three genotypes in the model are as follows, where $f(XX)$ denotes the frequency of the XX genotype: $w(AA)=1-3*f(Aa)+3*f(aa)$; $w(Aa)=1-s*f(Aa)$; $w(aa)=1-3*f(Aa)+3*f(AA)$.

Usage

```
freqdep(p0=0.01,s=0,time=100,show="p",pause=0)
```

Arguments

p0	Starting frequency for the A allele.
s	Parameter that determines the strength of selection against heterozygotes when they are common.
time	Number of generations to run the analysis.
show	Various options for plotting. "p" shows the frequency of A through time; "surface" plots the mean fitness as a function of p; "delta p" shows the change in p as a function of p; "cobweb" creates a cobweb plot showing p(t) by p(t+1). The default is show="p".
pause	Pause between generations. pause=0.01 (for instance) might smooth animation.

Value

The function creates one of several possible plots, depending on the value of show.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[sexratio](#)

Examples

```
freqdep(time=100)
freqdep(s=1.5,time=100)
freqdep(s=2,show="cobweb",time=100)
```

`genetic.drift`*Genetic drift simulation*

Description

This function simulates genetic drift at a biallelic genetic locus with no selection and no mutation in a sexually reproducing diploid population or set of populations.

Usage

```
genetic.drift(p0=0.5, Ne=20, nrep=10, time=100, show="p", pause=0.1)
```

Arguments

<code>p0</code>	Starting frequency for the A allele.
<code>Ne</code>	Effective population size.
<code>nrep</code>	Number of replicate simulations.
<code>time</code>	Total time, in number of generations, for the simulation.
<code>show</code>	Various options for plotting. "p" shows the frequency of the A allele through time; "genotypes" creates an animated histogram with the frequencies of each of the three genotypes through time; "fixed" shows the fraction of populations that have fixed for each allele, a or A; "heterozygosity" plots the mean heterozygosity and the expected heterozygosity through time. The default is show="p".
<code>pause</code>	Pause between generations. pause=0.01 (for instance) might smooth animation.

Value

The function creates one of several possible plots, depending on the value of show.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[founder.event](#), [selection](#)

Examples

```
genetic.drift()  
genetic.drift(p0=0.7, show="heterozygosity")
```

hawk.dove

Analysis of hawk-dove game theoretic model

Description

This function performs numerical analysis of a discrete-time hawk-dove model in which "payoff" determines relative fitness in the population.

Usage

```
hawk.dove(p=c(0.01,0.99),M=NULL,time=100)
```

Arguments

p	Starting frequency of hawk & dove phenotypes, respectively. Should correspond with the rows of M.
M	Payoff matrix. Should M[i, j] should contain the fitness of i interacting with j.
time	Number of generations.

Value

The function creates a two panel plot. The upper panel shows the relative frequencies of each of the two interacting phenotypes. The lower panel shows mean fitness of the population and of each morph through time.

Author(s)

Liam Revell <liam.revell@umb.edu>

Examples

```
hawk.dove(time=100)
```

mutation.selection

Gene frequencies over time under mutation-selection balance

Description

This function performs numerical analysis of mutation-selection balance with mutation from A to a and selection against (either or both of) Aa and aa.

Usage

```
mutation.selection(p0=1.0,w=c(1,1),u=0.001,time=100,show="q",pause=0,ylim=c(0,1))
```

Arguments

p0	Starting frequency for the A allele.
w	Fitnesses of the heterozygote (Aa) and homozygote deleterious (aa) genotypes. The fitness of AA is assumed to be 1.0.
u	Rate at which A mutates to a.
time	Number of generations to run the analysis.
show	Two options for plotting. "q" shows the frequency of a through time; "fitness" plots the mean fitness over time. The default is show="q".
pause	Pause between generations. pause=0.01 (for instance) might smooth animation.
ylim	Limits on the y-axis for plotting.

Value

The function creates one of two possible plots, depending on the value of show.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[genetic.drift](#), [selection](#)

Examples

```
mutation.selection(w=c(1,0),time=100,ylim=c(0,0.1))
```

rcd

Simulation of reproductive character displacement in an ecological community

Description

This function conducts individual-based, genetically explicit numerical simulation of reproductive character displacement in an ecological community. The model is one of multiple species (with fixed relative abundance) competing to utilize the same signal space. There is both stabilizing selection on the signal trait for detectability, as well as (in multi-species simulations) countervailing selection for divergence due to the costs of erroneous mismatching attempts.

Usage

```
rcd(nsp=3,nindivs=c(700,400,100),w_t=10,gen=c(500,500),figs="on",pf=100)
```

Arguments

nsp	Number of species in the simulation. If <code>figs="on"</code> , nsp must be 1, 2, or 3.
nindivs	A vector of length nsp containing the integer number of individuals in each species of the simulation.
w_t	Shape parameter of the Gaussian selection surface for the male signalling trait.
gen	Vector containing the number of allopatric generations followed by the number of sympatric generations for simulation.
figs	Either "on" if plotting is turned on, or "off" to suppress plotting.
pf	Print frequency for the simulation status to screen.

Value

The function returns a list containing the mean male signal trait and the mean female preference over time. It also (optionally) plots these.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[genetic.drift.selection](#)

Examples

```
## Not run: X<-rcd(nsp=2,nindivs=c(500,500))
```

selection

Numerical analysis of biallelic locus frequency independent selection

Description

This function performs numerical analysis of a simple biallelic selection model.

Usage

```
selection(p0=0.01,w=c(1.0,0.9,0.8),time=100,show="p",pause=0)
```

Arguments

p0	Starting frequency for the A allele.
w	Fitnesses for the three genotypes in the following order: AA, Aa, aa.
time	Number of generations to run the analysis.
show	Various options for plotting. "p" shows the frequency of A through time; "surface" plots the mean fitness as a function of p; "delta p" shows the change in p as a function of p; "cobweb" creates a cobweb plot showing p(t) by p(t+1). The default is show="p".
pause	Pause between generations. pause=0.01 (for instance) might smooth animation.

Value

The function creates one of several possible plots, depending on the value of show.

Author(s)

Liam Revell <liam.revell@umb.edu>

See Also

[freqdep](#)

Examples

```
selection(w=c(1.0,0.8,0.8),time=500)
selection(w=c(1.0,1.0,0.0),show="surface")
```

sexratio

Hypothetical analysis of frequency dependent selection on a sex determining genetic locus

Description

This function performs numerical analysis of a frequency dependent selection model of a hypothetical diploid sexually reproducing population in which sex is determined by the genotype at a biallelic genetic locus. Genotype AA are male, genotype aa are female, and genotype Aa might be male or female with equal probability.

Usage

```
sexratio(p0=0.01,time=100,show="p",pause=0)
```

Arguments

p0	Starting frequency for the A allele. Individuals with AA genotypes are male, while individuals with Aa genotypes are male or female with equal probability.
time	Number of generations to run the analysis.
show	Two different options for plotting. "p" shows the frequency of A through time; "fitness" plots the mean fitness through time. The default is show="p".
pause	Pause between generations. pause=0.01 (for instance) might smooth animation.

Value

The function creates one of two possible plots, depending on the value of show. Numerical analysis of this model shows how frequency dependent selection should favor alleles that tend to produce the rarer sex in the population.

Author(s)

Liam Revell <liam.revell@umb.edu>

sexratio

9

See Also

[sexratio](#)

Examples

```
sexratio(time=100)
```

Index

*Topic **character displacement**

rcd, 6

*Topic **drift**

founder.event, 2

genetic.drift, 4

rcd, 6

*Topic **evolutionary theory**

freqdep, 3

hawk.dove, 5

mutation.selection, 5

sexratio, 8

*Topic **game theory**

hawk.dove, 5

*Topic **population genetics**

founder.event, 2

freqdep, 3

genetic.drift, 4

mutation.selection, 5

rcd, 6

selection, 7

sexratio, 8

*Topic **sexual selection**

rcd, 6

founder.event, 2, 4

freqdep, 3, 8

genetic.drift, 2, 4, 6, 7

hawk.dove, 5

mutation.selection, 5

rcd, 6

selection, 4, 6, 7, 7

sexratio, 3, 8, 9