Package 'iMKT'

December 21, 2017

Title McDonald and Kreitman Test and its extensions calculation

Version 0.1.1

Date 2017-12-01

Description McDonald and Kreitman Test and its extensions.

Author Sergi Hervas <sergi.hervas@uab.cat>
Marta Coronado <marta.coronado@uab.cat>
Jesús Murga <jesus.murga@uab.cat>

Maintainer Jesús Murga <jesus.murga@uab.cat>

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Depends R (>= 3.3),
ggplot2,

Imports knitr,
utils,
stats,
cowplot,
reshape2,
nls2,
MASS,
ggthemes

VignetteBuilder knitr

R topics documented:

`asymptoticMK` *asymptoticMK*

Description

`asymptoticMK` developed in "Haller BC, Messer PW. asymptoticMK: A Web-Based Tool for the Asymptotic McDonald-Kreitman Test. G3 (Bethesda). 2017 May 5;7(5):1569-1575". Adapted from: http://github.com/MesserLab/asymptoticMK

Usage

```
asymptoticMK(daf, divergence, xlow, xhigh, seed)
```

Arguments

`daf`	data frame containing DAF, Pi and P0 values
`divergence`	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
`xlow`	lower limit for asymptotic alpha fit
`xhigh`	higher limit for asymptotic alpha fit
`seed`	seed value (optional). No seed by default

Details

The standard McDonald and Kreitman test (MKT) is used to detect the signature of selection at the molecular level. The MKT compares the amount of variation within a species (polymorphism, P) to the divergence (D) between species at two types of sites, one of which is putatively netral and used as the reference to detect selection at the other type of site. In the standard MKT, these sites are synonymous (putatively neutral, 0) and non-synonymous sites (selected sites, i) in a coding region. Under strict neutrality, the ratio of the number of selected and neutral polymorphic sites (Pi/P0) is equal to the ratio of the number of selected and neutral divergence sites (Di/D0).

Value

Estimation of alpha asymptotic value and details of the model fit

Examples

```
# asymptoticMK(mydafdata, mydivergencedata, 0, 0.9)
```

`check_input` *check_input*

Description

Check input data and return detailed errors when it is malformed.

Usage

```
check_input(daf, divergence, xlow, xhigh)
```

Arguments

`daf`	data frame containing DAF, Pi and P0 values
`divergence`	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
`xlow`	lower limit for asymptotic alpha fit
`xhigh`	higher limit for asymptotic alpha fit

Details

This function checks input data used in most package's functions (arguments daf, divergence, xlow and xhigh) and returns a brief description of the error(s) found. If data does not pass check_input() the requested analysis is not performed.

Value

None

`completeMKT` *completeMKT*

Description

`completeMKT()` put details here

Usage

```
completeMKT(daf, divergence, xlow, xhigh, seed)
```

Arguments

daf	data frame containing DAF, Pi and P0 values
divergence	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
xlow	lower limit for asymptotic alpha fit
xhigh	higher limit for asymptotic alpha fit
seed	seed value (optional). No seed by default

Details

put description here

Value

Execute all the MKT extensions

Examples

```
# completeMKT(mydafdata, mydivergencedata, 0, 0.9)
```

DGRP *DGRP*

Description

`DGRP()` Perform MKT corrected with DGRP method

Usage

```
DGRP(daf, divergence, list_cutoffs = c(0, 0.05, 0.2), plot = FALSE)
```

Arguments

daf	data frame containing DAF, Pi and P0 values
divergence	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
list_cutoffs	list of cutoffs to use (optional). Default cutoffs are: 0, 0.05, 0.2
plot	report plot (optional). Default is FALSE

Details

The standard McDonald and Kreitman test (MKT) is used to detect the signature of selection at the molecular level. The MKT compares the amount of variation within a species (polymorphism, P) to the divergence (D) between species at two types of sites, one of which is putatively netral and used as the reference to detect selection at the other type of site. In the standard MKT, these sites are synonymous (putatively neutral, 0) and non-synonymous sites (selected sites, i) in a coding region. Under strict neutrality, the ratio of the number of selected and neutral polymorphic sites (Pi/P0) is equal to the ratio of the number of selected and neutral divergence sites (Di/D0).The null hypothesis of neutrality is rejected in a MKT when Di/D0 > Pi/P0. The excess of divergence relative to polymorphism for class i, is interpreted as adaptive selection for a subset of sites i. The fraction of adaptive fixations, alpha.symbol, is estimated from 1-(Pi/P0)(D0/Di). The significance of the test can be assesed with a Fisher exact test. The estimate of alpha.symbol can be easily biased by the segregation of slightly deleterious non-synonymous substitutions. Specifically, slightly deleterious mutations tend to contribute more to polymorphism than to divergence, and thus, lead to an underestimation of alpha. Because adaptive mutations and weakly deleterious selection sct in opposite directions on the MKT, alpha and the fraction of substitutions that are sligholty deleterious, b, will be both underestimated when the two selection regimes occur. To take adaptive and slighlty deleterious mutations mutually into account, Pi, the count off segregatning sites in class i, should be seaprated into the number of neutral variants and the number of weakly deleterious variants, Pi = Pineutral + Pi weak del. Alpha is then estimated as 1-(Pineutral/P0)(D0/Di)

Value

MKT corrected by the DGRP method. List with alpha results, graph, divergence metrics, MKT tables and negative selection fractions

Examples

```
## Using default cutoffs
# DGRP(mydafdata, mydivergencedata)
## Using custom cutoffs
# DGRP(mydafdata, mydivergencedata, c(0.05, 0.1, 0.15))
```

`FWW` *FWW*

Description

`FWW()` Perform MKT corrected with FWW method

Usage

```
FWW(daf, divergence, list_cutoffs = c(0, 0.05, 0.1), plot = FALSE)
```

Arguments

daf	data frame containing DAF, Pi and P0 values
divergence	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
list_cutoffs	list of cutoffs to use (optional). Default cutoffs are: 0, 0.05, 0.2
plot	report plot (optional). Default is FALSE

Details

The standard McDonald and Kreitman test (MKT) is used to detect the signature of selection at the molecular level. The MKT compares the amount of variation within a species (polymorphism, P) to the divergence (D) between species at two types of sites, one of which is putatively netral and used as the reference to detect selection at the other type of site. In the standard MKT, these sites are synonymous (putatively neutral, 0) and non-synonymous sites (selected sites, i) in a coding region. Under strict neutrality, the ratio of the number of selected and neutral polymorphic sites (Pi/P0) is equal to the ratio of the number of selected and neutral divergence sites (Di/D0). The null hypothesis of neutrality is rejected in a MKT when Di/D0 > Pi/P0. The excess of divergence relative to polymorphism for class i, is interpreted as adaptive selection for a subset of sites i. The fraction of adaptive fixations, alpha.symbol, is estimated from 1-(Pi/P0)(Ds/Dn). The significance of the test can be assesed with a Fisher exact test.The estimate of alpha.symbol can be easily biased by the segregation of slightly deleterious non-synonymous substitutions. Specifically, slightly deleterious mutations tend to contribute more to polymorphism than to divergence, and thus, lead to an underestimation of alpha. Bevause they tend to segregate at lower frequencies than do neutral mutations, they can be apartially controled for by removing low frequency polymorphisms from the analysis (Fay et al. 2001). This is known as the FWW method.

Value

MKT corrected by the FWW method

Examples

```
## Using default cutoffs
# FWW(mydafdata, mydivergencedata)
## Using custom cutoffs and rendering plot
# FWW(mydafdata, mydivergencedata, c(0.05, 0.1, 0.15), plot=TRUE)
```

---

iMK *iMK*

Description

iMK: alpha asymptotic + negative selection (d,b,f)

#details details here

Usage

```
iMK(daf, divergence, xlow, xhigh, seed, plot = FALSE)
```

Arguments

daf	data frame containing DAF, Pi and P0 values
divergence	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes
xlow	lower limit for asymptotic alpha fit
xhigh	higher limit for asymptotic alpha fit
seed	seed value (optional). No seed by default
plot	report plots of daf, alpha and negative selection fractions (optional). Default is FALSE

Value

iMK

Examples

```
## Without plot
# iMK(mydafdata, mydivergencedata, 0, 0.9)
## With plot
# iMK(mydafdata, mydivergencedata, 0, 0.9, plot=TRUE)
```

---

loadPopFly *loadPopFly*

Description

Load PopFly dataset

Usage

```
loadPopFly()
```

Details

This function loads PopFly data (http://popfly.uab.cat/) into the current workspace. Data is stored in a dataframe named PopFlyData.

Value

None

Examples

```
# loadPopFly()
```

`loadPopHuman` *loadPopHuman*

Description

Load PopHuman dataset

Usage

```
loadPopHuman()
```

Details

This function loads PopHuman data (http://pophuman.uab.cat/) into the current workspace. Data is stored in a dataframe named PopFlyData.

Value

None

Examples

```
# loadPopHuman()
```

`multipleDatasets` *multipleDatasets*

Description

Perform any MK test using all files (or a subset of them) in a given directory.

Usage

```
multipleDatasets(directory = directory, test = c("standard", "DGRP", "FWW",
  "asymptotic", "iMK"), xlow = 0, xhigh = 1, fullAnalysis = TRUE/FALSE,
  idList = "NA")
```

Arguments

directory	directory (path/to/files/) where daf and divergence files are stored in your local machine
test	which test to perform. Options include: standard (default), DGRP, FWW, asymptotic, iMK
xlow	lower limit for asymptotic alpha fit (default=0)
xhigh	higher limit for asymptotic alpha fit (default=1)
fullAnalysis	decide whether to analyze all files in directory or not (default=TRUE)
idList	used when fullAnalysis = F, list of IDs to analyze

Details

Files in directory must be named: file1*daf*, file1*divergence*, file2*daf*, file2*divergence*, ...

Value

None

Examples

```
##example here
```

---

mydafdata *Example data frames*

Description

Containing basic information to execute the funcionts

- price. price in US dollars (\$326–\$18,823)
- carat. weight of the diamond (0.2–5.01)
- cut. quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- colour. diamond colour, from J (worst) to D (best)
- clarity. a measurement of how clear the diamond is (I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best))
- x. length in mm (0–10.74)
- y. width in mm (0–58.9)
- z. depth in mm (0–31.8)
- depth. total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)
- table. width of top of diamond relative to widest point (43–95)

Usage

```
mydafdata
```

Format

A data frame with x rows and y variables

`mydivergencedata` *Example data frames*

Description

Containing basic information to execute the funcionts

- price. price in US dollars (\$326–\$18,823)
- carat. weight of the diamond (0.2–5.01)
- cut. quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- colour. diamond colour, from J (worst) to D (best)
- clarity. a measurement of how clear the diamond is (I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best))
- x. length in mm (0–10.74)
- y. width in mm (0–58.9)
- z. depth in mm (0–31.8)
- depth. total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)
- table. width of top of diamond relative to widest point (43–95)

Usage

```
mydivergencedata
```

Format

A data frame with x rows and y variables

PopFlyAnalysis *PopFlyAnalysis*

Description

Perform any MK test using a subset of PopFly data defined by custom genes and populations lists

Usage

```
PopFlyAnalysis(genes = c("gene1", "gene2", "..."), pops = c("pop1", "pop2",
  "..."), recomb = TRUE/FALSE, bins = 0, test = c("standard", "DGRP",
  "FWW", "asymptotic", "iMK"), xlow = 0, xhigh = 1)
```

Arguments

genes	list of genes
pops	list of populations
recomb	group genes according to recombination values (must specify number of bins). TRUE/FALSE. Recomb values (cM/Mb) from Comeron et al. 2012.
bins	number of recombination bins to compute (mandatory if recomb = TRUE)
test	which test to perform. Options include: standard (default), DGRP, FWW, asymptotic, iMK
xlow	lower limit for asymptotic alpha fit (default=0)
xhigh	higher limit for asymptotic alpha fit (default=1)

Details

Recombination values (recomb=T) from Comeron et al. 2012 (reference!)

Value

None

Examples

```
## List of genes
# mygenes <- c("FBgn0053196", "FBgn0086906", "FBgn0261836", "FBgn0031617",
#              "FBgn0260965", "FBgn0028899", "FBgn0052580", "FBgn0036181",
#              "FBgn0263077", "FBgn0013733", "FBgn0031857", "FBgn0037836")
## Perform analyses
# PopFlyAnalysis(genes=mygenes , pops=c("RAL","ZI"), recomb=F, test="iMK", xlow=0, xhigh=0.9)
# PopFlyAnalysis(genes=mygenes , pops=c("RAL","ZI"), recomb=T, bins=3, test="DGRP")
```

standard *mkt_standard*

Description

`mkt_standard()` MKT standard formula

Usage

```
standard(daf, divergence)
```

Arguments

`daf`	data frame containing DAF, Pi and P0 values
`divergence`	data frame containing divergent and analyzed sites for selected (i) and neutral (0) classes

Details

The standard McDonald and Kreitman test (MKT) is used to detect the signature of selection at the molecular level. The MKT compares the amount of variation within a species (polymorphism, P) to the divergence (D) between species at two types of sites, one of which is putatively netral and used as the reference to detect selection at the other type of site. In the standard MKT, these sites are synonymous (putatively neutral, 0) and non-synonymous sites (selected sites, i) in a coding region. Under strict neutrality, the ratio of the number of selected and neutral polymorphic sites (Pi/P0) is equal to the ratio of the number of selected and neutral divergence sites (Di/D0).

Value

Standard McDonald and Kreitman Test

Examples

```
# standard(mydafdata, mydivergencedata)
```

subsetPopData *subsetPopData*

Description

Perform iMK using a subset of PopFly or PopHuman data defined by custom genes and populations lists

Usage

```
subsetPopData(data = c("PopFly", "PopHuman"), genes = c("gene1", "gene2",
  "..."), pops = c("pop1", "pop2", "..."), recomb = TRUE/FALSE, bins = 0,
  test = c("standard", "DGRP", "FWW", "asymptotic", "iMK"), xlow = 0,
  xhigh = 1)
```

Arguments

data	input PopFly or PopHuman data
genes	list of genes
pops	list of populations
recomb	group genes according to recombination values (must specify number of bins). TRUE/FALSE
bins	number of recombination bins to compute (mandatory if recomb = TRUE)
test	which test to perform. Options include: standard (default), DGRP, FWW, iMK, asymptotic, none.
xlow	lower limit for asymptotic alpha fit (default=0)
xhigh	higher limit for asymptotic alpha fit (default=1)

Details

put details here

Value

None

Examples

```
## Load PopFly data
# loadPopFly()
## Perform analysis
# mygenes <- c("FBgn0053196", "FBgn0000008")
# subsetPopData("PopFly", mygenes , c("RAL","ZI"), recomb=F)
```

---

theme_Publication *ggplot theme for publication ready Plots*

Description

Date = 04/07/2015 Author = Koundinya Desiraju

Usage

```
theme_Publication(base_size = 14, base_family = "sans")
```

Arguments

base_size	base size required from theme_Publication
base_family	font to load in theme_Publication

Details

Default theme used for plot images. From http://rpubs.com/Koundy/71792

Value

plot theme

Examples

```
# theme_Publication(14, "sans")
```

Index