

Population genetic offset under new environments.

Description

The function returns genetic offset estimates computed from user-specified population labels and new environments based on predictions of an `lfmm2` model. It takes as input an object of class `lfmm2Class` together with the data that were used to adjust the LFMM, and a matrix of new environmental variables in the same format as the original ones.

Usage

```
genetic.offset(object, input, env, new.env, pop.labels)
```

Arguments

<code>object</code>	An object of class <code>lfmm2Class</code> .
<code>input</code>	A genotypic matrix or a character string containing a path to the input file. The genotypic matrix must be in the <code>lfmm{lfmm_format}</code> format without missing values (9 or NA). See <code>impute</code> for completion based on nonnegative matrix factorization and consider R packages for reading large matrices.
<code>env</code>	A matrix of environmental covariates or a character string containing a path to the environmental file. The environment matrix must be in the <code>env</code> format without missing values. All variables must be encoded as <code>numeric</code> .
<code>new.env</code>	A matrix of new environmental covariates or a character string containing a path to the new environmental data file. The new environmental matrix must be in the <code>env</code> format without missing values, and of same dimension as the <code>env</code> matrix. All variables must be encoded as <code>numeric</code> .
<code>pop.labels</code>	A numeric or character vector providing population labels for all rows (individuals) of the response matrix.

Value

`offset` A matrix of genetic offset values computed for every population in `pop.labels`.

Author(s)

Olivier Francois

References

Gain C, Francois O (2020). LEA 3: Factor models for population and ecological genomics in R.

See Also

`lfmm.data lfmm2`

Examples

```
### Example of offset prediction using lfmm2 ###

# Simulation with 100 target loci, with effect sizes ranging between -10 an 10
# n = 100 individuals and L = 1000 loci

X <- as.matrix(rnorm(100)) # environmental variable
B <- rep(0, 1000)
target <- sample(1:1000, 100) # target loci
B[target] <- runif(100, -10, +10) # effect sizes

# Creating hidden factors and loadings
```

```
U <- t(tcrossprod(as.matrix(c(-1.25,0.5,1.25)), X)) + matrix(rnorm(300), ncol = 3)
V <- matrix(rnorm(3000), ncol = 3)

# Simulating a binarized matrix containing haploid genotypes
# Simulation performed with a generative LFMM

Y <- tcrossprod(as.matrix(X), B) + tcrossprod(U, V) + matrix(rnorm(100000, sd = .5), nrow = 100)
Y <- matrix(as.numeric(Y > 0), ncol = 1000)

#####
# Fitting an LFMM with K = 3 factors #
#####

mod2 <- lfmm2(input = Y, env = X, K = 3)

# Computing genetic offset statistics for 2 populations, defined from latent factor 1
pop <- 1 + (U[,1] > 0)

g.offset <- genetic.offset(object = mod2, input = Y,
                           env = X, new.env = 2*X + 10,
                           pop.labels = pop)

round(g.offset, digit = 3)

#rm(list = ls())
```

[Package *LEA* version 3.2.0 Index]