

Package: TraitEnvMLMWA (via r-universe)

September 18, 2024

Title Weighted Averaging (WA) and Multi-Level Methods (MLM) for Trait-Environment Association in Functional and Community Ecology

Version 0.9001

Description This package grew out of the paper New robust weighted averaging- and model-based methods for assessing trait-environment relationships (Methods in Ecology and Evolution). The model-based methods are GLMMs, here called multi-level models (MLM), of which the novel MLM3 outperforms all existing methods. MLM3 has up to 30% more power, but the WA-methods are 500 times faster than MLM3. The main function for WA-methods is CWMSNC_regressions. The main functions for MLM3 are make_obj_for_traitenv, expand4glmm and MLM3_p_max.

Depends R (>= 3.5.0)

License GPL-3 | file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

NeedsCompilation no

Imports lme4, glmmTMB, ggplot2, dplyr, permute, stats

Suggests ade4, boot

Author Cajo ter Braak [aut, cre]

Maintainer Cajo ter Braak <Cajo.terBraak@wur.nl>

Repository <https://cajoterbraak.r-universe.dev>

RemoteUrl <https://github.com/CajoterBraak/TraitEnvMLMWA>

RemoteRef HEAD

RemoteSha 02216b3097078d0644c72aa656453f20c4fbf400

Contents

Bootstrap_test_prmtrc	2
CWMSNC_regressions	4
dat4MLM2TE_obj	7
expand4glmm	8
make_obj_for_traitenv	10
MLM3	11
MLM3_Aravo	11
MLM3_p_max	12
plot.CWMSNCr	14
plot_MLM3	16
Revisit	17
summary.CWMSNCr	18
WA_p_max	20
Index	24

Bootstrap_test_prmtrc *Parametric bootstrap test of trait-environment association using the MLM models*

Description

Bootstrap_test_prmtrc performs a parametric bootstrap test on trait-environment interaction, starting from a fitted MLM3 object (or any other MLM object with the trait:env term as last fixed parameter).

Usage

```
Bootstrap_test_prmtrc(
  MLM3,
  test_stat = "Wald",
  nrepet = 19,
  Binomial_total = 0,
  nAGQ = 0
)
```

Arguments

MLM3	the fitted MLM3 object, created by glmer (lme4) or glmmTMB.
test_stat	choice of test statistic; 'Wald' (default), 'LRT' or 'both'. The default is quicker.
nrepet	number of bootstraps
Binomial_total	scalar, 0 for count-like data and the binomial total for logit models (1 for presence-absence).
nAGQ	integer scalar (default 0), used only for an object created by glmer

Details

The code assumes that the parameter for trait-environment interaction (trait:env) is the last fixed parameter in `summary(MLM3)`. First, the formula of the null model is created by deleting the trait:env term from the formula of object MLM3 (the non-null model). Second, the null model (MLM0) is fitted. Third, sampling is from this null model. For each simulated data set, the model is refitted using the formula of the MLM3 object. The code works therefore also for MLM1 and MLM2, although their use is not recommended. See also Box A2 in Appendix A4 and Appendix A1.

Value

A named list,

p_values	the parametric and bootstrap p-values
MLM0	the fitted null model
obs	value of the test statistic(s)
nrepet	number of bootstraps
sim.boot	values of the test statistic(s) for the nrepet bootstrapped data
test_stat	the chosen test statistic(s)

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[expand4glmm](#).

Examples

```
## Not run:
#use a precomputed MLM3 model, e.g. from the Revisit data
data("MLM3")
## or compute the MLM3 model from the data
# data("Revisit")
# formula.MLM3 <- y ~ poly(env,2) + poly(trait,2) +
#   env : trait + (1 + env|species) + (1 + trait| site)
# MLM3 <- glmTMB(formula.MLM3, family = betabinomial, data=Revisit)
summary(MLM3)
res_boot <- Bootstrap_test_prmtrc(MLM3, test_stat = "Wald", nrepet = nrepet, Binomial_total = 100)
names(res_boot)
round(res_boot$p_values,3)

## End(Not run)
```

CWMSNC_regressions	<i>Weighted averaging methods for pairwise trait-environment associations</i>
--------------------	---

Description

CWMSNC_regressions carries out weighted or unweighted CWM/SNC regression for each combination of trait and environmental variable with permutation testing via the max test and computation of the (un)weighted fourth-corner correlations. The output can be summarized and plotted using `summary` and `plot`.

Usage

```
CWMSNC_regressions(
  E,
  L,
  T,
  weighing = "N2",
  cutoff = 0,
  nrepet = 499,
  how_to_permute = list(sites = how(), species = how())
)
```

Arguments

E	a data frame, numeric n-vector, factor or n-by-p matrix, containing the values of the environmental variables in n sites, or a TE_obj object containing the trait, environment and abundance data,
L	a n by m matrix or data frame of abundance values of m species in n sites
T	a data frame, numeric m-vector, factor or m-by-q matrix, containing the values of the trait(s) for m species
weighing	type of weights used in the regression of CWM on E and SNC on T; "unw", "FC" or "N2" (default)
cutoff	minimal number of occurrences of species
nrepet	number of permutations
how_to_permute	a list for two <code>how</code> calls from the <code>permute</code> package, the first for site permutation, the second for species permutation. Default: completely random permutation.

Details

CWMSNC_regressions is for pairwise association between traits and environmental variables. Factors in E and T data frames are expanded in to dummy variable format. For p-values of an entire factor, use `WA_p_max`. Entering E and T as matrices or data frames has the advantage the variable names are retained.

The permutation test(s) are carried out by the max test. This test takes the maximum of the p-values of the CWM-E regression(s) and the SNC-T regression(s). Both p-values are obtained by permutation. In the former, the rows of E are permuted (i.e. the sites), in the latter the rows of T are permuted (i.e. the species).

Details of the weighing options: "FC" give the results of the unembellished fourth-corner correlation (which uses weighting by site and species totals); "unw" gives each site and species equal weight in the regression; "N2" weighs each site and species in proportion to its Hill number of order 2 (N2); i.e. the effective richness of the site and effective number of occurrences of the species. All weighing variants use the same CWM and SNC values (the weights in calculating CWM and SNC being the abundances in L).

The option `nperm` in `how` is overruled by `nrepet`.

Value

An object of class `CWMSNCr` which is a named list, among which,

<code>p_values</code>	three permutational p-values: site-based and species-based and the maximum of these two values
<code>p_parametric</code>	parametric p-values: site-based and species-based and the maximum of these two values (can be trusted only for weighing <code>== 'unw'</code>). NA for more than one trait or one environmental variable.
<code>wFC</code>	(weighted) fourth-corner correlations for sites and species and its signed minimum.
<code>lm_CWMe</code>	lm-object of the (possibly multiple!) regression of CWM on E. This item is pairwise only for a single environmental variable.
<code>lm_SNCt</code>	lm-object of the (possibly multiple!) regression of SNC on T. This item is pairwise only for a single trait.

References

- ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)
- ter Braak, C.J.F., Peres-Neto, P.R. & Dray, S. (2018) Simple parametric tests for trait-environment association. *Journal of Vegetation Science*, 29, 801-811. <https://doi.org/10.1111/jvs.12666>
- Peres-Neto, P.R., Dray, S. & ter Braak, C.J.F. (2017) Linking trait variation to the environment: critical issues with community-weighted mean correlation resolved by the fourth-corner approach. *Ecography*, 40, 806-816. <https://doi.org/10.1111/ecog.02302>

See Also

[plot.CWMSNCr](#) and [summary.CWMSNCr](#).

Examples

```
# Aravo data set -----
data("aravo", package = "ade4")
Y <- aravo$spe
```

```

SLA <- aravo$traits$SLA
Snow <- aravo$env$Snow
nrepet <- 19 # change to e.g. 499 or 999
result <- CWMSNC_regressions(Snow, Y, SLA, weighing = "N2", nrepet = nrepet)
result$p_values
summary(result)
plot(result)

# untransformed analysis of all pairs. See TutorialWA_Aravo_Multi.r
#                                     for transformations that appear useful
# data frames E,L,T
result <- CWMSNC_regressions(aravo$env, aravo$spe, aravo$traits, weighing = "N2", nrepet = nrepet)
#result$p_values # contains all pairwise p-values
#               for site-based, species-based and max-based permutations
#result$wFC     # contains all pairwise weighted fourth-corner correlations
# (site-based, species-based, and signed min-based)
summary(result, type = "max", p_value_adjust_method = "fdr", significance_level = 0.05)
# in the E or T matrix or data frame case, you cannot choose your own labels;
# names should refer to names of variables or labels of factors
(nam.list <- plot(result, trait = "", env = "" ))
plot(result, trait = "Spread", env = "Snow" )
## All plots
# for (trait in nam.list$trait.names){
#   for (env in nam.list$env.names){
#     print(plot(result, trait = trait, env = env ))
#   }
# }

# get Revisit data -----
data("Revisit")
## adapt dataframe dat for WA-based analyses
n_sites <- with(Revisit, nlevels(factor(site)))
n_species <- with(Revisit, nlevels(factor(species)))
species <- Revisit$species[seq(from = 1, by = n_sites, length.out = n_species)]
sites <- Revisit$site[1:n_sites]
trait <- Revisit$trait[seq(from = 1, by = n_sites, length.out = n_species)]
env <- Revisit$env[1:n_sites]
Y <- matrix(Revisit$value, nrow = n_sites, ncol = n_species,
            dimnames = list(sites= sites, species=species))
# #Alternative using the function dat4MLM2TE_obj
# TE_obj <- dat4MLM2TE_obj(Revisit)
# trait <- TE_obj$T; env <- TE_obj$E; Y <- TE_obj$L
# CWM/SNC regressions -----
set.seed(1231)
result <- CWMSNC_regressions(env, Y, trait, weighing = "N2", nrepet = nrepet)
summary(result)
plot(result)

# Comparison of two versions of the max test-----
obj <- make_obj_for_traitenv(env,Y, trait, cutoff=0)
set.seed(1231)

```

```

# slow
system.time(aa<- WA_p_max(obj, nrepet = nrepet, fast = FALSE))
round(aa$p_values,4)

set.seed(1231)
# default
system.time(bb<- WA_p_max(obj, nrepet = nrepet, fast = TRUE))
round(bb$p_values,4)
# illustrating their equality
or1<- order(aa$sim.row)
or2<- order(bb$sim.row)
all.equal(or1,or2)
or1<- order(aa$sim.col)
or2<- order(bb$sim.col)
all.equal(or1,or2)

# fast, all three weighings -----

system.time(bb<- WA_p_max(obj, weighing = "N2", nrepet = nrepet))
round(bb$p_values,5)

system.time(FC<- WA_p_max(obj, weighing = "FC", nrepet = nrepet))
round(FC$p_values,5)

system.time(cc<- WA_p_max(obj, weighing = "unw", nrepet = nrepet))
round(cc$p_values,5)

# comparion with fourtcorner in ade4 -----

ade4::fourthcorner(data.frame(env),as.data.frame(Y),data.frame(trait), nrepet = nrepet)
result <- CWMSNC_regressions(env, Y, trait, weighing = "FC", nrepet = nrepet)
summary(result,digits = 4)

```

dat4MLM2TE_obj

Utility function dat4MLM2TE_obj

Description

dat4MLM2TE_obj creates from data in a format for glm(m)/MLM a TE_obj object (see [make_obj_for_traitenv](#)), which consists of trait, environment and abundance data. The argument dat is usually created using [expand4glm](#). The function is used repeatedly in [MLM3_p_max](#).

Usage

```
dat4MLM2TE_obj(dat, cutoff = 0)
```

Arguments

dat	dataframe with names y, site, species, trait and env for values of abundance, identity of site and of species and the trait value and environmental value, respectively.
cutoff	minimal number of occurrences of species

Details

The order of data in dat should be in standard order, that is, first all data on species 1, then species 2, with all sites listed in identical order for each species (including zero abundances). **BEWARE:** The function currently works for a single trait and environmental variable only!!!!

Value

An object of class TE_obj

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

Examples

```
## Not run:
data("Revisit")
str(Revisit)
TE_obj <- dat4MLM2TE_obj(Revisit)
str(TE_obj)

## End(Not run)
```

expand4glmm

Utility function expand4glmm

Description

expand4glmm inflates a TE_obj object (see [make_obj_for_traitenv](#)), which consists of trait, environment and abundance data, into a format for glm, glmer and glmmTMB.

Usage

```
expand4glmm(obj, K = 0)
```

Arguments

obj	an object of class TE_obj, usually the output of function make_obj_for_traitenv .
K	scalar, 0 for count-like data and the binomial total for data that you like to analyse using logit models (1 for presence-absence).

Details

With single trait and single environment variable data ($\text{ncol}(\text{obj}\$T) == \text{ncol}(\text{obj}\$E) == 1$), the names of the trait and environmental variables are changed to `trait` and `env` (to allow identical MLM formulas for different data and variables). In the multi-variable case, the original names are kept and all interactions are added. `expand4glmm` is used repeatedly in the model-based permutational max test using function `MLM3_p_max`. In this function, `obj` is extended with list elements `trait0` and `env0` to allow for the detail needed in the model-based permutational max test. If used in this way, the internal logical `inPermut_r_c` is set to `TRUE`.

Value

A data frame. For single trait and environmental variable data:

<code>y</code>	the response; the abundance values in <code>obj\$L</code>
<code>site</code>	a factor indicating the site corresponding to each value of <code>y</code>
<code>species</code>	a factor indicating the species corresponding to each value of <code>y</code>
<code>obs</code>	integer 1 to <code>N</code> , the length of <code>y</code>
<code>trait</code>	trait
<code>env</code>	environmental variable
<code>trait.env</code>	the product

In the multi-trait environment case: all traits and environmental variables and all pairwise interactions

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

ter Braak, C.J.F., Peres-Neto, P. & Dray, S. (2017) A critical issue in model-based inference for studying trait-based community assembly and a solution. *PeerJ*, 5, e2885. <https://doi.org/10.7717/peerj.2885>

Examples

```
## Not run:
data("aravo", package = "ade4")
Y <- aravo$spe
trait <- scale(aravo$traits$SLA)
env <- scale(aravo$env$Snow)
obj <- make_obj_for_traitenv(env, Y, trait, cutoff=0)
K = 0 # 0 for non-binomial models
dat <- expand4glmm(obj, K = K)
names(dat)
library(glmTMB)
formula.MLM3 <- y ~ poly(env,2) + poly(trait,2) +
  env : trait + (1 + env|species) + (1 + trait| site)
MLM3 <- glmTMB(formula.MLM3, family = betabinomial, data=dat)
summary(MLM3)
```

```
plot_MLM3(MLM3)
## End(Not run)
```

make_obj_for_traitenv *Utility function make_obj_for_traitenv*

Description

make_obj_for_traitenv combines the three data items needed for analyzing trait-environment association into a single object of class TE_obj. In the process, empty sites and non-occurring species are deleted. There is an option for stronger deleting criteria.

Usage

```
make_obj_for_traitenv(E, L, T, cutoff = 0)
```

Arguments

E	a numeric n-vector or n by p matrix containing p environmental variable(s) of n sites
L	a n by m matrix or data frame of abundance values of m species in n sites
T	a numeric m-vector or m by q matrix containing q traits of m species
cutoff	minimal number of occurrences of species

Details

If E or T is a matrix, the column names are retained. If E or T is a vector, E gets name "env" and T gets name "trait". If you wish to retain the original name, enter E and T as data frames or single column matrices with a column name.

Value

A named list,

L	a sites by species matrix of abundances, after removal of species and sites, based on cutoff
E	a matrix of environmental values (rows are the same sites as the rows of L)
T	a matrix of trait values (rows are the same as the column of L)

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[expand4g1mm](#) and [WA_p_max](#).

Examples

```
## Not run:
data("aravo", package = "ade4")
Y <- aravo$spe
trait <- scale(aravo$traits$SLA)
env <- scale(aravo$env$Snow)
obj <- make_obj_for_traitenv(env, Y, trait, cutoff=0)
str(obj)

## End(Not run)
```

MLM3

MLM3-fit to Revisit data using trait C:N ratio and environmental variable TMG

Description

MLM3 is a `glmmTMB` object consisting of the fit of the MLM3 model to the Revisit data using the code

```
formula.MLM3 <- y ~ trait*env +(1+trait|site)+(1+env|species)
MLM3 <- glmmTMB(formula.MLM3, family = betabinomial, data= Revisit)
```

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[Revisit.](#)

MLM3_Aravo

quadratic MLM3-fit to Aravo data using SLA and Snow

Description

MLM3_Aravo is a `glmerMod` object from the `lme4` package consisting of the quadratic fit of the MLM3 model to the Aravo data from the `ade4` package using the trait SLA and the environmental variable Snow with code

```
formula.MLM3.quad <- y ~ poly(trait,2) + poly(env,2) +trait:env +
(1+trait|site)+(1+env|species)
MLM3_Aravo <- glmer(formula.MLM3.quad, data=Aravo_in_glmmformat,
family= poisson, nAGQ=0, control = glmerControl(calc.derivs=F))
To get the data in Aravo_in_glmmformat use expand4glmm.
```

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution*, Appendix A7 (<https://doi.org/10.1111/2041-210X.13278>)

MLM3_p_max	<i>Permutational max test of trait-environment association for MLM models</i>
------------	---

Description

MLM3_p_max performs the permutational max test of trait-environment interaction for MLM models, starting from a fitted MLM3 object.

Usage

```
MLM3_p_max(
  MLM3,
  nrepet = 19,
  Binomial_total = 0,
  test_stat = "Wald",
  how_to_permute = list(sites = how(), species = how()),
  print = 1,
  nAGQ = 0
)
```

Arguments

MLM3	the fitted MLM3 object, created by <code>glmer</code> (lme4) or <code>glmmTMB</code> .
nrepet	number of bootstraps
Binomial_total	scalar, 0 for count-like data and the binomial total for logit models (1 for presence-absence).
test_stat	choice of test statistic; 'Wald' (default) or 'LRT'.
how_to_permute	a list for two <code>how</code> calls, the first for site permutation, the second for species permutation.
print	integer; print progress every print iterations
nAGQ	integer scalar (default 0), used only for an object created by <code>glmer</code>

Details

The code assumes that the interaction parameter is the last fixed parameter in `summary(MLM3)`. The data used in the max test is extracted using `dat4MLM2TE_obj`. This generates an object of class `TE_obj` (see `make_obj_for_traitenv`). `dat4MLM2TE_obj` is limited for use with a single trait and single environmental variable, and so is therefore `MLM3_p_max`. In the model-based permutation tests, either the trait values or the environmental values in the interaction term T*E of the model

are permuted to yield a species- and site-level test, respectively. Main effects for the permuted trait and environmental variable are added to ensure that the interaction after permutation has a corresponding main effect. For further details, see Appendices A4 and A5 of ter Braak (2019).

Value

A named list, among which,

p_values	four p-values: one parametric p-value (Wald test) and three permutational p-values: site-based and species-based and the maximum of these two values
obs	values of the test statistic for sites (first row) and species (second row)
sim.row	values of the test statistic for the nrepet data in which the rows of E are permuted
sim.col	values of the test statistic for the nrepet data in which the rows of T are permuted
nrepet	number of permutations

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[expand4glm](#).

Examples

```
## Not run:
#use a precomputed MLM3 model for the Revisit data
data("MLM3")
## or compute an MLM3 model from the data
# data("Revisit")
# formula.MLM3 <- y ~ poly(env,2) + poly(trait,2) +
  env : trait + (1 + env|species) + (1 + trait| site)
# MLM3 <- glmTMB(formula.MLM3, family = betabinomial, data=Revisit)
summary(MLM3)
res_perm <- MLM3_p_max(MLM3, test_stat = "Wald", nrepet = nrepet, Binomial_total = 100)
names(res_perm)
round(res_perm$p_values,3)

## End(Not run)
```

plot.CWMSNCr	<i>Plotting the results of a weighed averaging-based analysis of pairwise trait-environment association(s)</i>
--------------	--

Description

plot.CWMSNCr plots the results of [CWMSNC_regressions](#) for the single quantitative trait and the single quantitative environmental variable, specified by the arguments `trait` and `env`. Both main effects (site totals vs E and species totals vs T) and the associations (CWM vs E and SNC vs T) are plotted.

Usage

```
## S3 method for class 'CWMSNCr'
plot(x, ..., title = NA, trait = 1, env = 1)
```

Arguments

<code>x</code>	an object of class <code>CWMSNCr</code> , created by CWMSNC_regressions .
<code>...</code>	other optional arguments
<code>title</code>	character text for the main title of the association graph (optional)
<code>trait</code>	trait name or level of a nominal trait to be plotted.
<code>env</code>	environmental variable name or level of a nominal environmental variable to be plotted

Details

In the one E vector - one T vector case, you can choose your own labels for `trait` and `env`. In the E or T matrix or data frame case, names should refer to names of variables or labels of factors. If in error, a list of names is returned. The arguments `trait` and `env` can be numbers referring to the column numbers in `x$E` and `x$T`; these numbers do not reflect the original E and T if any variable in them is a factor (nominal).

The code uses the libraries `ggplot2` and `dplyr`.

Value

A list of two printable and modifiable objects of class `ggplot2`. The first plots the main effects, the second the association. The y-coordinate of the points in the first plot is the logarithm of the site total (left) and the species total (right). The y-coordinate of the points in the second plot is weighted trait mean (CWM, left) and weighted environmental mean (SNC, right). The long-dash line is at the weighted mean of the y-coordinate of the points in plots. The dotted line is at the unweighted mean of the trait (left) and of the environmental variable (right). If, in the second-left plot, the dashed line is above the dotted line, the trait main effect is positive (check in first plot, right diagram). If, in the second-right plot, the dashed line is above the dotted line, the environmental main effect is positive (check in first plot, left diagram). In this sense, the second plot shows not only the association (interaction) but also the sign of the main effects, shown more explicitly in the first plot.

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[CWMSNC_regressions.](#)

Examples

```
# get Aravo data set -----
data("aravo", package = "ade4")
Y <- aravo$spe
SLA <- aravo$traits$SLA
Snow <- aravo$env$Snow
nrepet <- 19 # change to e.g. 499 or 999
result <- CWMSNC_regressions(Snow, Y, SLA, weighing = "N2", nrepet = nrepet)
names(result)
result$p_values
summary(result)
plot(result)

Snow <- aravo$env$Snow
Spread <- log(aravo$traits$Spread)
result <- CWMSNC_regressions(Snow, Y, Spread, weighing = "N2", nrepet = nrepet)
result$p_values
summary(result)
# in the one E vector - one T vector case, you can choose your own labels
plot(result, trait = "log Spread", env = "Snow")

# untransformed analysis of all pairs. See TutorialWA_Aravo_Multi.r
#                                     for transformations that appear useful
# data frames E,L,T
result <- CWMSNC_regressions(aravo$env, aravo$spe, aravo$traits, weighing = "N2", nrepet = nrepet)
#result$p_values # contains all pairwise p-values
#               for site-based, species-based and max-based permutations
#result$wFC     # contains all pairwise weighted fourth-corner correlations
# (site-based, species-based, and signed min-based)
summary(result, type = "max", p_value_adjust_method = "fdr", significance_level = 0.05)
# in the E or T matrix or data frame case,
# names should refer to names of variables or labels of factors
# The next statement generates the available valid names.
(nam.list <- plot(result, trait = "", env = "" ))
plot(result, trait = "Spread", env = "Snow" )
## All plots
# for (trait in nam.list$trait.names){
#   for (env in nam.list$env.names){
#     print(plot(result, trait = trait, env = env ))
#   }
# }
```

`plot_MLM3`*Plotting the interaction of a MLM3 model*

Description

`plot_MLM3` plots the trait-environment interaction of a fitted MLM3 object.

Usage

```
plot_MLM3(  
  MLM3,  
  verbose = FALSE,  
  title = paste("site and species effects (fixed + random) against env and trait")  
)
```

Arguments

<code>MLM3</code>	the fitted MLM3 object, created by <code>glmer</code> (lme4) or <code>glmmTMB</code> .
<code>verbose</code>	logical; default FALSE. If TRUE, a summary of the MLM3 object is printed.
<code>title</code>	character text to adapt the main title.

Details

The code uses libraries `ggplot2` and `dplyr`. The trait and environment variable names in the MLM3 model must be 'trait' and 'env' as in the example (as they are hard coded)

Value

A list of printable and modifiable objects of class `ggplot2`.

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[expand4glmm](#).

Examples

```
## Not run:  
data("Revisit")  
formula.MLM3 <- y ~ env*trait + (1 + env|species) + (1 + trait| site)  
MLM3 <- glmmTMB(formula.MLM3, family = betabinomial, data=Revisit)  
plot_MLM3(MLM3)  
  
## End(Not run)
```

Revisit

Revisit data

Description

The data is from the revisit to the low-elevation, non-serpentine sites of Robert Whittaker's historic plant community study sites in the Siskiyou Mountains of Southwest Oregon (Damschen, Harrison & Grace 2010). Briefly, the data consists of the abundance of 75 species in 52 sites, calculated from the number of 100 quadrat corners per site that each species intersected (Miller et al. 2018, 2019). The environmental variable and functional trait are standardized versions of the original topographic moisture gradient (TMG) and the leaf carbon-to-nitrogen ratio (C:N), respectively (Miller et al. 2019). The dataframe `Revisit` has been derived from the `whittaker_revisit_data.csv` file from Miller et al. (2018), who describe the variables as follows

- site study site ID (integer, one study plot per site)
- species name of species
- trait functional trait: leaf tissue carbon to nitrogen ratio
- env environmental variable: site position on the topographic moisture gradient (higher numbers are topographically drier sites)
- value plant abundance (see Description for details)
- y matrix with two columns with, as successes and failures, value and 100 - value, respectively.

References

- Damschen, E.I., Harrison, S. & Grace, J.B. (2010) Climate change effects on an endemic-rich edaphic flora: resurveying Robert H. Whittaker's Siskiyou sites (Oregon, USA). *Ecology*, 91, 3609-3619. <https://doi.org/10.1890/09-1057.1>
- Miller JED, Damschen EI, Ives AR (2018) Data from: Functional traits and community composition: a comparison among community-weighted means, weighted correlations, and multilevel models. Dryad Digital Repository. <https://doi.org/10.5061/dryad.7gj0s3b>
- Miller JED, Damschen EI, Ives AR (2019) Functional traits and community composition: a comparison among community-weighted means, weighted correlations, and multilevel models. *Methods in Ecology and Evolution*. <https://doi.org/10.1111/2041-210x.13119>
- ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

summary.CWMSNCr	<i>Summary of a weighed averaging-based analysis of trait-environment association</i>
-----------------	---

Description

summary.CWMSNCr summarizes the results of [CWMSNC_regressions](#). For a single quantitative trait and a single quantitative environmental variable three types of correlations and p-values are given (sites, species, min/max). For nominal and multiple trait and environment data, one set of such correlations and p-values is printed (by default, the max p-values and signed minimum fourth-corner correlations) with an associated heatmap of the significant associations. The p-values can be adjusted for multiple testing.

Usage

```
## S3 method for class 'CWMSNCr'
summary(
  object,
  ...,
  digits = 3,
  type = "max",
  p_value_adjust_method = "fdr",
  significance_level = 0.05,
  silent = FALSE
)
```

Arguments

object	an object of class CWMSNCr, created by CWMSNC_regressions .
...	other optional arguments
digits	number of digits to print
type	"sites", "species" or "max" (default, which combines sites and species). Used only for multiple or nominal trait and environmental data
p_value_adjust_method	method for adjustment of the p-values for multiple comparison. "fdr" (default), "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", or "none". Used only for multiple or nominal trait and environmental data
significance_level	threshold for plotting a heatmap of the signed p-values
silent	no printed output when TRUE (default FALSE)

Details

p_value_adjust_method uses [p.adjust](#) on P-values of all pxq trait-environment combinations of the site-level and the species-level tests, which, for type == 'max' are subsequently combined by taking the maximum of the adjusted site P-value of the CWM regression and adjusted species

P-value for the SNC regression for each combination. This adjustment method is the same as used in [fourthcorner](#) and is less conservative than applying `p.adjust` on the max P-values directly.

Value

For quantitative single trait-single environmental variable data: a numeric matrix with trait-environment correlations (first row) and p-values (second row) for sites, species and their min/max combination (columns). Correlations in the min/max column are signed minima of sites and species, being 0 if the correlations for sites and species differ in sign. p-values in the min/max column are the maximum of the p-values for sites and species.

For nominal or multiple trait/environment data: a list of

<code>p_val_adj</code>	the adjusted p-values
<code>FCcorrelations</code>	the (weighted) fourth corner correlations
<code>heatmap</code>	a gglot object containing the basis heatmap on the basis of argument <code>significance_level</code>

References

ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)

See Also

[plot.CWMSNCr](#).

Examples

```
# get Aravo data set -----
data("aravo", package = "ade4")
Y <- aravo$spe
SLA <- aravo$traits$SLA
Snow <- aravo$env$Snow
nrepet <- 19 # change to e.g. 499 or 999
result <- CWMSNC_regressions(Snow, Y, SLA, weighing = "N2", nrepet = nrepet)
names(result)
result$p_values
summary(result)
plot(result)

Snow <- aravo$env$Snow
Spread <- log(aravo$traits$Spread)
result <- CWMSNC_regressions(Snow, Y, Spread, weighing = "N2", nrepet = nrepet)
result$p_values
summary(result)
# in the one E vector - one T vector case, you can choose your own labels
plot(result, trait = "log Spread", env = "Snow")

# untransformed analysis of all pairs. See TutorialWA_Aravo_Multi.r
#                                     for transformations that appear useful
# data frames E,L,T
```

```

result <- CWMSNC_regressions(aravo$env, aravo$spe, aravo$traits, weighing = "N2", nrepet = nrepet)
#result$p_values # contains all pairwise p-values
#               # for site-based, species-based and max-based permutations
#result$wFC     # contains all pairwise weighted fourth-corner correlations
# (site-based, species-based, and signed min-based)
summary(result, type = "max", p_value_adjust_method = "fdr", significance_level = 0.05)
# in the E or T matrix or data frame case,
# names should refer to names of variables or labels of factors
# The next statement generates the available valid names.
(nam.list <- plot(result, trait = "", env = "" ))
plot(result, trait = "Spread", env = "Snow" )
## All plots
# for (trait in nam.list$trait.names){
#   for (env in nam.list$env.names){
#     print(plot(result, trait = trait, env = env ))
#   }
# }

```

WA_p_max

Max test for the joint association between multiple traits and environmental variables (incl. nominal variables) using weighted averaging methods (CWM/SNC regression)

Description

For the test on multi-trait multi-environment association, two test statistics are available, one which accounts for correlations among the traits and among the environmental variables (`score_test == TRUE`) and one which does not. The former is related to double constrained correspondence analysis (ter Braak et al. 2018) and the latter to RLQ (Dray et al. 2014). created using `make_obj_for_traitenv`. If E is a TE_obj, the arguments L and T are ignored.

Usage

```

WA_p_max(
  E,
  L = 0,
  T = 0,
  nrepet = 19,
  weighing = "N2",
  type = "joint",
  score_test = TRUE,
  cutoff = 0,
  fast = TRUE
)

```

Arguments

E	a data frame, numeric n-vector, factor or n-by-p matrix, containing the values of the environmental variables in n sites, or a TE_obj object containing the trait, environment and abundance data,
L	a n by m matrix or data frame of abundance values of m species in n sites
T	a data frame, numeric m-vector, factor or m-by-q matrix, containing the values of the trait(s) for m species
nrepet	number of permutations
weighing	type of weights used in the regression of CWM on E and SNC on T; "unw", "FC" or "N2" (default)
type	'joint' testing of the association between a set of traits and a set environmental variables (default) or 'onebyone' 1-1 tests for each pair of trait and environmental variable
score_test	logical. TRUE for the score test statistic, which accounts for correlations (default) and FALSE without accounting for correlations among traits and among environmental variables
cutoff	minimal number of occurrences of species
fast	logical, whether the fast version is used or the slow version (default TRUE). For multi-trait-environment data, fast is set to TRUE.

Details

Nominal traits and environmental variables (factors) are expanded into dummy variables if E and/or T are data frames. Set E or T to a data frame with a single factor to obtain test of association with the factor. For multi-trait, multi-environment data and score_test==TRUE, the trait and environment data are internally transformed to normalized singular vectors. If score_test==FALSE, the variables are standardized internally to weighted mean 0 and variance 1. The permutation tests is carried out by the max test. This test takes the maximum of the p-values of the CWM-E regression and the SNC-T regression. Both p-values are obtained by permutation. In the former, the rows of E are permuted, in the latter the rows of T are permuted. The slow version recomputes the CWM and SNC and the associated regression from scratch for each permutation and is available only with single numeric trait and environmental variable. The fast version precomputes the CWM and SNC and computes the test statistic by a matrix product in which the permuted traits or environmental variables are rescaled to weighted mean 0 and variance 1 using the precomputed weights (determined by the weighing option).

Value

A named list, among which,

p_values	five p-values: permutational, site-based and species-based and its maximum, and the parametric ones
obs	values of the test statistic for sites (first row) and species (second row)
sim.row	values of the test statistic for the nrepet data in which the rows of E are permuted
sim.col	values of the test statistic for the nrepet data in which the rows of T are permuted
nrepet	number of permutations

References

- ter Braak (2019) New robust weighted averaging- and model-based methods for assessing trait-environment relationships. *Methods in Ecology and Evolution* (<https://doi.org/10.1111/2041-210X.13278>)
- ter Braak, C.J.F., Šmilauer, P. & Dray, S. (2018) Algorithms and biplots for double constrained correspondence analysis. *Environmental and Ecological Statistics*, 25, 171-197. <https://doi.org/10.1007/s10651-017-0395-x>
- Dray, S., Choler, P., Dolédec, S., Peres-Neto, P.R., Thuiller, W., Pavoine, S. & ter Braak, C.J.F. (2014) Combining the fourth-corner and the RLQ methods for assessing trait responses to environmental variation. *Ecology*, 95, 14-21. <http://dx.doi.org/10.1890/13-0196.1>

Examples

```
# get Aravo data set -----
data("aravo", package = "ade4")
Y <- aravo$spe
# Aravo data set: multi-trait multi-env test -----
summary(aravo$env)
#E <- aravo$env[,-c(5,6)] # without Snow, without ZoogD
E <- aravo$env[,-c(5)] # without Snow, without ZoogD
# Slope is positive and very skew; therefore we logtransform, but there are zeroes
# add minimum non-zero value before taking logs
a <- min(E[E[,"Slope"]>0, "Slope"])
E[,"Slope"] <- log(E[,"Slope"]+ a)
summary(E)
T <- aravo$traits #[, -6 ] # without SLA
# all traits are skew; therefore we logtransform
T[,c(1,3)] <- log(T[,c(1,3)]+ 1) # traits with zeroes
T[,-c(1,3)] <- log(T[,-c(1,3)]) # traits without zeroes
summary(T)

set.seed(1231)
# score_test = TRUE gives the score test which
# takes intra-trait and intra-env correlations into account
#           If weighing == "FC", it give the test used
#           in double constrained correspondence analysis )
# score_test = FALSE ignores such correlations.
#           If weighing == "FC", it give the gives the test used in RLQ

E<- aravo$env[,c(1,6,2)]
T <- aravo$traits[,c(2,6)]
nrepet <- 99 # change to e.g. 499 or 999
cc<- WA_p_max(E, Y, T, nrepet =nrepet, weighing = "N2", type = "onebyone")
cc$p_values

dd<- CWMSNC_regressions(E, Y, T, nrepet =nrepet, weighing = "N2")

aa<- WA_p_max(E, Y, T, nrepet =nrepet, weighing = "N2", score_test = TRUE)
aa$p_values
```

```
# Explicit conversion to dummy variables as performed in WA_p_max -----
# convert any factor into dummy variables
E <- model.matrix(as.formula(paste("~ 0", paste(names(E), collapse= "+"), sep= "+")), data = E)
summary(E)
# convert any factor into dummy variables
T <- model.matrix(as.formula(paste("~ 0", paste(names(T), collapse= "+"), sep= "+")), data = T)
summary(T)
obj <- make_obj_for_traitenv(E,Y, T, cutoff=0)
set.seed(1231)

bb<- WA_p_max(obj, nrepet =nrepet, weighing = "N2", score_test = TRUE)
bb$p_values
```

Index

Bootstrap_test_prmtrc, 2

CWMSNC_regressions, 4, 14, 15, 18

dat4MLM2TE_obj, 7, 12

expand4glimm, 3, 7, 8, 10, 11, 13, 16

fourthcorner, 19

how, 4, 5, 12

make_obj_for_traitenv, 7, 8, 10, 12, 20

MLM3, 11

MLM3_Aravo, 11

MLM3_p_max, 7, 9, 12

p.adjust, 18

plot.CWMSNCr, 5, 14, 19

plot_MLM3, 16

Revisit, 11, 17

summary.CWMSNCr, 5, 18

WA_p_max, 4, 10, 20