

# Package ‘paltran’

January 2, 2012

**Type** Package

**Title** WA, WA-PLS, MW for paleolimnology

**Version** 1.3-0

**Date** 2008-04-29

**Author** Sven Adler

**Maintainer** Sven Adler <sven.adler2@uni-rostock.de>

**Depends** vegan, MASS, mgcv

**Description** This package contains functions for paleolimnology -wa-regression (see also package analogue by G. Simpson!), wa-pls and the mowing-window approach. The function palplot allows a first plot of the data including a trend analysis. The mtf function is just a first test version.

**License** GPL (>= 2)

**URL** <http://www.biologie.uni-rostock.de/abt/botanik/AG-Phykologie/index-engl.htm>

**Repository** CRAN

**Date/Publication** 2010-02-06 14:32:35

## R topics documented:

paltran-package . . . . .	2
age.dud . . . . .	3
dud.df . . . . .	4
mtf . . . . .	4
mw . . . . .	6
palplot . . . . .	9
rand.test . . . . .	11
train_env.MV . . . . .	12
train_set.MV . . . . .	12
wa . . . . .	13
wapls . . . . .	15

---

paltran-package	<i>WA, WA-PLS and MW for Paleolimnology</i>
-----------------	---

---

### Description

This package contains functions for paleolimnology - WA-regression (see also package analogue!), WA-PLS and MW regression. The function palplot allows a first plot of the data including a trend analysis. Additionally a transfer function based on proportional odds (POM) is added, but this function is just a first test version based on class data.

### Details

Package:	paltran
Type:	Package
Version:	1.3-0
Date:	2010-02-06
License:	GPL (>= 2)
LazyLoad:	yes

### Author(s)

Sven Adler <sven.adler2@uni-rostock.de>

### References

ter Braak, C.J.F. & van Dam, H. 1989. Inferring pH from diatoms: a comparison of old and new calibration methods. *Hydrobiologia* 178:209-23. ter Braak, C.J.F. & Juggins, S. 1993. Weighted averaging partial least squares regression WA-PLS: an improved method for reconstructing environmental variables from species assemblages. *Hydrobiologia* 269:485-502. Huebener, T., Dressler, M., Schwarz, A., Langner, K., Adler, S. 2008. Dynamic adjustment of training sets ('moving windows' reconstruction) by using transfer functions in paleolimnology - a new approach, *J. o. Paleolimnology* 40, 79-95

### See Also

package analogue by G. Simpson and J. Oksanen

### Examples

```
data(age.dud)
data(dud.df)
data(train_set.MV)
```

```
data(train_env.MV)
try<-wa(train_set.MV,train_env.MV,dud.df, val="boot",run=10)
# run should be set higher 100-1000

spec<-c(23,24,217,218,223,228)
c<-c(rep(4,6),2,3)
t<-c(rep("h",6),"l","p")
x<-c(rep("abundance",6),"n","mug/l")
tr<-c(rep(0,7),1)
palplot(dud.df[,spec],try[[12]],10^try[[13]],trend=tr,
cex.axis=0.8,p.col=c,ptype=t,p.main=c(names(dud.df[,spec]),
"N2","inf. log(TP)"),dis=0.2,p.xlab=x,age=age.dud[,1])

palplot(try[[14]],ptype="l",error=1.96*try[[15]],
error.ptype="p",error.plot=TRUE,p.col="blue",age=age.dud[,1])
```

---

age.dud	<i>age of sediment samples from a sediment core taken of lake Dudinghausen</i>
---------	--

---

## Description

age of the sediment samples within the data set dud.df

## Usage

```
data(age.dud)
```

## References

Dressler, M., Selig, U., Doerfler, W., Adler, S., Schubert, H., Huebener, T. 2006. Environmental changes and the migration period in Europe by the example of Lake Dudinghausen northern Germany. Quat. Res. 66:25-37.

## Examples

```
data(age.dud)
```

---

 dud.df

*Diatom samples of a sediment core - Lake Dudinghausen*


---

### Description

Lake Dudinghausen is a recent eutrophic lake. The lake morphology of Lake Dudinghausen (DUD) (53.54.30 N, 12.12.40 E) is characterized by a nearly homogeneously shaped basin. The lake is almost completely surrounded in agricultural used watershed, and only a small part of the northern shore is forested. Large areas of cattail are located at the western and eastern edges. Species codes are following the EDDI code <http://craticula.ncl.ac.uk/Eddi/jsp/index.jsp>

### Usage

```
data(dud.df)
```

### Format

A data frame with 133 observations enfolding 309 taxa.

### References

Dressler, M., Selig, U., Doerfler, W., Adler, S., Schubert, H., Huebener, T. 2006. Environmental changes and the migration period in Europe by the example of Lake Dudinghausen northern Germany. *Quat. Res.* 66:25-37.

### Examples

```
data(dud.df)
```

---

 mtf

*multinomial transfer function for Palaeolimnology*


---

### Description

Here the species abundances were assigned to different abundance classes, for each class a species response curve is generated using GLM. These response curves are used to infer the environmental parameter of interest (s. Details)

### Usage

```
mtf(..., d.plot = TRUE, scale = TRUE, datatype = TRUE,
n= 4, set.zero = FALSE, not.av = c("zero", "lower", "max"),
val = "loo", class=c(1,5,10,30,60), out=TRUE)
```

**Arguments**

...	x,y,z: required: species training set (x) as matrix and related environmental parameter (y). optional: core data (z) - species data from a sediment core
d.plot	if TRUE diagnostic plots are given at the end of the analysis
scale	should the data scaled up to 100 percent? (Default)
datatype	What kind of data input? TRUE for percentage data, otherwise no transformation to class data will be done
n	minimum number of species occurrence within on class
set.zero	If a species occurs within on class less than n time the values will be set as zero, if set.zero = TRUE. If set.zero = FALSE, the values will be allocate to the next lower class
not.av	If a species occurs within a abundance class that is not given in the training set, three possibilities are available. "zero" means that this species will be excluded, "lower" stands for to use the next lower class density. If "max" is chosen, the cumulative species response curve of all classes will be calculate and use
val	validation method: So fare only Leave-one-out is performed
class	what classes should be used? Default is 0,1,5,10,30,60
out	should the result be shown on the console?

**Details**

The relative abundances of a taxon within a data set is transformed to abundance classes (class 1:  $0 < x \leq 1$ , class 2:  $1 < x \leq 5$ , class 3:  $5 < x \leq 10$ , class 4  $10 < x \leq 30$ , class 5  $30 < x \leq 60$ , class 6: 60-100 percent abundance, number of classes and class borders can be changed). Occurs a taxa in a sediment core sample within the class 1, it might give better reconstructions, when the optimum for this specific species abundance class to a related environmental factor in the training set is used instead of the optimum of the hole response. This optimum can than be used to infer past environmental parameters. In general: Not the overall species optimum is used to infer environmental parameter, but several different optima were estimate related to different abundance classes using GLM with binomial distribution. The inferred value for the core sample is than calculated following the ML-Method: multiplying the class density functions of the species abundance class values from the core. The first version of this function (paltran 1.0) uses mainly the function polr from the package MASS (Venables, W.N. Ripley, B.D. (2002). Instead, each class is modelled using GLM. If a species occurs only rarely ( $< n$ ) in the training set tow options are possible. First, it can removed (set.zero=TRUE), secondly the occurrence can add to the next lower class. On one hand this is a misclassification, on the other hand, the data are not completely lost, and might improve the model. If a species class from a core sample is not represent in the training set, the same options can be chosen. The species can be removed (not.av="zero"), or the next lower density function can be used. Thirdly the overall density function (product of all single class density functions) - assuming a symmetric species distribution - can be chosen (not.av="max"). As validation method until now only Leave One Out is available, as the algorithm is still very slow. With improving computing time, other methods like bootstrap might be incorporated.

**Value**

inferred train.set  
inferred environmental parameter for the training set

performance performance of the pom-regression

spec distribution  
species distribution curves for the single classes

inferred train.set (loo)  
inferred environmental parameter for the training set using leave one out as cross validation method

reconstruction\_core.samples  
reconstructed environmental parameter for the samples of the core

mean (reconstruction\_core.samples) (loo)  
reconstructed environmental parameter for the samples of the core using leave one out

sd (reconstruction\_core.samples) (loo)  
standard deviation of the reconstructed environmental parameter for the samples of the core using "loo"

### Author(s)

Sven Adler

### References

in preperation

### Examples

```
data(train_set.MV)
data(train_env.MV)
data(dud.df)
try<-mtf(train_set.MV,train_env.MV,dud.df[1:3,],not.av="max",val=FALSE)

#using "loo" takes several minutes for computing
```

### Description

The moving window method identifies for each single fossil diatom sample an optimal training set using DCA (CA, CCA) or simple distance measurement in combination with WA or WA-PLS error statistik. Downweighting of rare taxa can be chosen as non significant taxa can be excluded

**Usage**

```
mw(train_set, train_env, core_data, method = c("wapls", "wa", "mtf"),
  comp = 4, val = c("boot", "loo", "10-cross"), run = 10,
  mwsize = c(20, 40, 60), dim = c(2, 3, 4),
  mw.type = c("dca", "ca", "sample", "cca"), dist.m = "euclidean",
  rmsep.incl = TRUE, env.trans = FALSE, spec.trans = FALSE,
  rplot = TRUE, drop.non.sig = FALSE, min.occ = 1, scale = FALSE,
  dw = TRUE, selection = c("rand.test", "pred.error"))
```

**Arguments**

train_set	required: matrix or data frame including species of the complete training set. rows = samples, columns = species, row and column names are required
train_env	required: environmental variable belongs to the training set
core_data	required: species data from a core, those taxa that are not in the training set will be omitted. Minimum number is two samples.
method	type "wa" for weighted averaging regression or "wapls" for weighted averaging-partial least square regression. Which type of transfer function should be used to infer the environmental variable to the core samples, default is "wapls"
comp	if wapls is used, how many components should be extract? Default is 4.
val	validation method for the transfer function, on of "loo" for leave-on-out, "boot" for bootstrap, or "10-cross" for 10-fold cross validation, default is boot
run	if "boot" or "10-cross" was chosen: how many cycles should be done? Should be low when running a new data set the first time, high values 1000 and more results in a large computing time
mwsize	vector of window size: how many nearest neighbours should be included? Default is 20,40,60.
dim	how many dimensions should be used when the nearest neighbours are calculate using the sample scores of DCA ore CA, default is 2.
mw.type	type "dca" for DCA, "ca" for CA or "sample" for simple distance measurement. When choosing "dca" or "ca" the core samples are plotted in the training set samples using predict.cca or predict.decorana (package vegan) and than the nearest training set samples to each single core sample are analysed. Using "sample" the ditances of the samples are analysed using the original species data instead of sample scores. Chosing "cca" a CCA is done and the scores of the first axis are used to analyse the nearest neighbours
dist.m	how to analyse the distance of the sample scores between training samples and core samples? All distances that are incorporated in vegdist (package vegan) are possible to use.
rmsep.incl	should the RMSEP be include in model selection or only R2.cross, mean(error).cross and max(error).cross
env.trans	should the environmental parameter bee transformed? "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.
spec.trans	should the species data bee transformed? "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.

<code>rplot</code>	should a plot during the analysis be shown? Is set to be FALSE if <code>mw.type</code> equals "sample" or "cca"
<code>drop.non.sig</code>	should a taxon that have non significant response to the environmental variable within the mw-training set be deleted? The calculation, if there is a significant relation between a taxa and the environmental variable of interest, is undertaken using a generalized additive model (GAM) and the package <code>mgcv</code> . As a GAM only works if a taxon occurred several times, only those taxa will be included that occurred more than 5 times ( $k=3$ ). If the <code>mwsiz</code> is too small, it can happen, that no taxa have a significant response and the function stops
<code>min.occ</code>	minimum occurrence: all taxa with less than <code>min.occ</code> will be deleted from the training set
<code>scale</code>	should the data scaled up to 100 percent? (Default = FALSE)
<code>dw</code>	should rare taxa be downweighted? (see function <code>downweight</code> in the <code>vegan</code> package by J. Oksanen)
<code>selection</code>	should the best window size be chosen based on the error of the transfer function (Huebener et al. 2008) or should be the randomisation test by van der Voet (1994) be used?

### Details

Using `mw`, for each sample 3 WA-PLS runs (default) are calculated using 100 bootstrap runs for each. This takes time. The reconstruction for a whole sediment (80-100 samples) core can take several minutes. Please try first with a small test set or with a low value for `run` (see examples), before running the whole reconstruction! At least the number of components (default is 4) can be reduced to make the function faster. Data must be organised in the same way as running `cca` or `dca` in package `vegan`. This approach needs large training sets, like the combined TP data set from EDDI. If as selection `rand.test` is chosen the randomisation test of van der Voet (1994) will be used as following: Let  $k$  be the size of the smaller window and  $k+n$  the size of the larger window. For the transfer function based on  $k$  samples the sample specific error will be estimated (bootstrap or LOO) and additionally the prediction error will be estimated for the  $n$  samples. These  $k+n$  prediction errors will then be compared with the sample specific error of the transfer function based on  $k+n$  samples using the randomisation test as described in the appendix of Van der Voet (1994). If the error of the transfer function based on  $k$  samples for the  $k+n$  samples is significantly lower or just equal than the error of the transfer function based on  $k+n$  samples, the smaller window size will be used to infer the environmental parameter for the given test sample. If the prediction error is higher for the transfer function based on  $k$  samples, the larger MW size will be used to infer the environmental parameter for the given test sample.

### Value

<code>sample.performance</code>	gives all information for each core sample, which window size was used and the performance of the related transfer function
<code>reconstruction</code>	reconstructed values for the core sample
<code>mean(reconstruction).val</code>	mean values for the reconstruction for the core sample using bootstrap or 10-fold cross validation

```
sd(reconstruction).val
      standard deviation of the reconstructed values for the core sample using boot-
      strap or 10-fold cross validation
```

### Author(s)

Sven Adler

### References

Huebener, T., Dressler, M., Schwarz, A., Langner, K., Adler, S. 2008. Dynamic adjustment of training sets ('moving windows' reconstruction) by using transfer functions in paleolimnology - a new approach, *J. o. Paleolimnology* 40: 79-95

### See Also

wa, wapls, package analogue (G. Simpson and J Oksanen) and package vegan (J. Oksanen)

### Examples

```
data(dud.df)
data(train_set.MV)
data(train_env.MV)
test<-dud.df[1:3,]
fit<-mw(train_set.MV, train_env.MV, test, mwsizer = c(40, 60), val="boot", run=5, comp=3)
names(fit)

fit<-mw(train_set.MV, train_env.MV, test, mwsizer = c(40, 60, 80)
, comp=3, method="wa", val="loo",)

fit<-mw(train_set.MV, train_env.MV, test, mwsizer = c(40, 60), run=5,
mw.type="sample", dist.m="bray", dw=TRUE)
```

---

palplot

*plotting a simple plot for paleolimnology*

---

### Description

Before exporting the results of wa, wapls or mw functions to C2 or TILIA simple stratigraphic plots can be done with palplot including several variables to check the results. For each variable the colour and the plot style can be chosen, error bars and trend lines can be added.

### Usage

```
palplot(..., age = NA, error.plot = FALSE, error = 0, ptype = "b",
error.ptype = "b", p.col = "black",
p.lty = 1, p.xlab = NULL, cex.main = 1,
cex.axis = 1, dis = 0.15, p.main = NULL,
y.lab = "sample", trend = 0, span = 0.1)
```

**Arguments**

...	required: matrixes, vectors or data frames of species data and or reconstructed environmental parameters
age	if available: a vector including the age of the samples
error.plot	should error bars be included?
error	vector or matrix included the error
p.type	how should the variables be plotted? String or vector: "h" for bars, "p" for points, "l" for line, "b" for both line and points
error.p.type	how should the errors be plotted? String or vector: "b" for error bars, "p" for shaded regions
p.col	string or vector: colour of the plots
p.lty	if "l" is chosen for p.type, the line type can be chosen. string or vector
p.xlab	string or vector: labels of the x axis of each variable
cex.main	numeric or vector: size of the main of the plots for each variable
cex.axis	numeric or vector: size of x axis labels
dis	distance between the single plots
p.main	string or vector: main of the plots for each variable
y.lab	string: label of the y axis
trend	should a trend line be plotted? String or vector
span	what span should be used for the trend lines

**Author(s)**

Sven Adler

**See Also**

package analogue (G. Simpson) or package palaeo (S. Juggins)

**Examples**

```

data(age.dud)
data(dud.df)
data(train_set.MV)
data(train_env.MV)
try<-wa(train_set.MV, train_env.MV, dud.df, val="boot", run=10)

spec<-c(23,24,217,218,223,228)
c<-c(rep(4,6),2,3)
t<-c(rep("h",6),"l","p")
x<-c(rep("abundance",6),"n","mug/l")
tr<-c(rep(0,7),1)
palplot(dud.df[,spec], try[[12]], 10^try[[13]], trend=tr,
cex.axis=0.8, p.col=c, p.type=t, p.main=c(names(dud.df[,spec]),
"N2", "inf. log(TP)"), dis=0.2, p.xlab=x)

```

```
palplot(try[[14]], ptype="l", error=1.96*try[[15]],  
error.ptype="p", error.plot=TRUE, p.col="blue", age=age.dud[,1])
```

---

rand.test	<i>randomisation test</i>
-----------	---------------------------

---

## Description

This function computes a randomisation test to compare two different models

## Usage

```
rand.test(error1, error2, ran.nb = 1000, seed = 1)
```

## Arguments

error1	error of model 1
error2	error of model 2
ran.nb	number of runs
seed	set the value of the function set.seed)

## Value

MSEP1	mean of error of model 1
MSEP2	mean of error of model 1
p	if <0.05 both models are of significant different quality

## Author(s)

Sven Adler

## References

van der Voet, H. (1994) Comparing the predictive accuracy of models using a simple randomisation test. *Chemometrics and Intelligent Laboratory Systems* 28: 165-180.

## Examples

```
data(train_set.MV)  
data(train_env.MV)  
try<-waps(train_set.MV, train_env.MV, val="boot")  
e1<-try[[5]][,2]-train_env.MV  
e2<-try[[5]][,3]-train_env.MV  
rand.test(e1, e2)
```

---

`train_env.MV`*North German diatom training set - Total Phosphorous*

---

**Description**

For each lake, water samples were collected three times a year (spring, summer and autumn) at a water depth of 0.5 m and analysed for 19 environmental parameters using standard methods. All lakes, excepting nine shallow lakes are dimictic and have surface areas >50 ha. The here given Total Phosphorous (TP) values are the mean from spring, summer and autumn TP. The data are log10 transformed.

**Usage**

```
data(train_env.MV)
```

**Format**

A data frame with 84 observations

**Author(s)**

Sven Adler and Thomas Huebener

**Source**

[www.biologie.uni-rostock.de/abt/botanik/AG-Phykologie/index-engl.htm](http://www.biologie.uni-rostock.de/abt/botanik/AG-Phykologie/index-engl.htm)

**References**

in preperation

**Examples**

```
data(train_env.MV)
```

---

`train_set.MV`*North German diatom training set*

---

**Description**

The Mecklenburg-Vorpommern data set consists of 84 lakes located in the NE German lowlands (Mecklenburg-Western Pomerania, Mecklenburg-Vorpommern (MV)). The lakes are typical kettle-type, formed in glacial till of Weichselian origin, and are well-buffered, hard water and carbonate-rich lakes. Intensification of agricultural in the lake catchments during recent decades has resulted in more or less eutrophic conditions in the most of the lakes. Species codes are following the EDDI code <http://craticula.ncl.ac.uk/Eddi/jsp/index.jsp>

**Usage**

```
data(train_set.MV)
```

**Author(s)**

Sven Adler and Thomas Huebener

**Source**

[www.biologie.uni-rostock.de/abt/botanik/AG-Phykologie/index-engl.htm](http://www.biologie.uni-rostock.de/abt/botanik/AG-Phykologie/index-engl.htm)

**Examples**

```
data(train_set.MV)
```

---

wa	<i>weighted averaging (WA) regression for paleolimnology</i>
----	--

---

**Description**

This function computes with a given training set and a given environmental parameter a weighted averaging transfer function as used in paleolimnology. For the calculation of the model predicting error 10 fold cross validation, bootstrap or Leave-on-out can be chosen. Inverse or classical deshrinking are supported.

**Usage**

```
wa(..., d.plot = TRUE, env.trans = FALSE, spec.trans = FALSE,
diagno = TRUE, val = c("none", "n.cross", "loo", "boot"),
run = 10, scale = FALSE, seed = 1, out = TRUE,
desh.meth = c("class", "inverse"),
drop.non.sig = FALSE, min.occ = 1, nfold = 10)
```

**Arguments**

...	required x,y: a matrix or data frame of the species training set (x) and a vector or data frame of the related environmental parameter (y). optional: core samples (z) - vector or data frame of species data from a sediment core.
d.plot	TRUE/FALSE: if TRUE diagnostic plots are given at the end of the analysis.
env.trans	Should the environmental parameter be transformed? Type "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.
spec.trans	Should the species data be transformed? "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.
diagno	should N2, number of non zero values be calculated for the training set and test set? Default is TRUE

val	validation method: one of "boot" (bootstrap), "loo" (Leave-on-out), or "n.cross" (n-fold cross validation)
run	if "boot" or "n.cross" were chosen: number of cycles to run
scale	should the data scaled up to 100 percent? (Default is FALSE)
seed	set the seed for the random generator (using boot or 10-cross), default = 1
out	should the results printed on the console?
desh.meth	what kind of deshrinking method should be used "class"(classical deshrinking), or "inverse" (inverse deshrinking), default is "inverse"
drop.non.sig	should a taxon that have non significant response to the environmental variable be deleted? The calculation, if there is a significant relation between a taxa and the environmental variable of interest, is undertaken using a generalized additive model (GAM) and the package mgcv. As a GAM only works if a taxon occurred several times, only those taxa will be included that occurred more than 5 times (k=3).
min.occ	minimum occurrence: all taxa with less than min.occ will be deleted from the training set
nfold	what type of n-fold cross validation should be used?

### Value

species in train.set	Number of non zero species in each sample of the training set
N2 train.set	Hill's N2 of each sample of the training set
species.optima	wa-optima of each species
inferred train.set	inferred environmental parameter for the training set
performance	performance of the wa-regression
species in core.samples	Number of none zero species in each sample of the core data set
n species core.samples in train.set	How many species in the core samples are represented in the training set
N2 in core.samples	Hill's N2 of each sample of the core data
reconstruction_core.samples	reconstructed environmental parameter for the samples of the core
inferred train.set.val	mean inferred environmental parameter for the training set using cross validation
mean(reconstruction_core.samples).val	reconstructed environmental parameter for the samples of the core using "boot" or "loo"
sd(reconstruction_core.samples).val	standard deviation of the reconstructed environmental parameter for the samples of the core using "boot" or "loo"
reconstruction_core.samples.val	reconstructed environmental parameter for the samples of the core for each run of "boot" or "loo"

**Author(s)**

Sven Adler

**References**

ter Braak, C.J.F. & van Dam, H. 1989. Inferring pH from diatoms: a comparison of old and new calibration methods. *Hydrobiologia* 178:209-23.

**See Also**

package analogue

**Examples**

```
data(train_set.MV)
data(train_env.MV)
data(dud.df)
try<-wa(train_set.MV,train_env.MV)
try<-wa(train_set.MV,train_env.MV,desh.meth="class")
names(try)
try<-wa(train_set.MV,train_env.MV,dud.df,val="boot",run=10)
```

---

wapls	<i>weighted averaging - partial least square (WA-PLS) regression for paleoecology</i>
-------	---

---

**Description**

This function computes with a given training set and environmental parameter a weighted averaging - partial least square (WA-PLS) transfer function as used in paleolimnology. For the calculation of the model predicting error 10 fold cross validation, bootstrap, or Leave-on-out can be chosen.

**Usage**

```
wapls(..., comp = 4, d.plot = TRUE, plot.comp = "RMSEP", env.trans = FALSE,
spec.trans = FALSE, diagno = TRUE, seed = 1, run = 10,
val = c("none", "10-cross", "loo", "boot"), scale =FALSE,
out = TRUE, drop.non.sig = FALSE, min.occ = 1)
```

**Arguments**

...	required x,y: a matrix or data frame of the species training set (x) and a vector or data frame of the related environmental parameter (y). optional: core samples (z) - vector or data frame of species data from a sediment core.
comp	number of components that will be calculated
d.plot	TRUE/FALSE: if TRUE diagnostic plots are given at the end of the analysis.

plot.comp	if "RMSEP" is chosen, the diagnostic plot for that component is given with the lowest RMESP
env.trans	should the environmental parameter be transformed? "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.
spec.trans	should the species data be transformed? "sqrt" for square root and "log10" for the logarithm to the basis 10 are possible choices, default is FALSE.
diagno	should N2, number of non zero values be calculated for the training set and test set? Default is TRUE
seed	set the seed for the random generator (using boot or 10-cross), default = 1
run	if "boot" or "10-cross" were chosen: number of cycles to run
val	validation method: one of "boot"(bootstrap), "loo"(Leave-on-out), or "10-cross"(10-fold cross validation)
scale	should the data scaled up to 100 percent? (Default is FALSE)
out	should the results printed on the console?
drop.non.sig	should a taxon that have non significant response to the environmental variable be deleted? The calculation, if there is a significant relation between a taxa and the environmental variable of interest, is undertaken using a generalized additive model (GAM) and the package mgcv. As a GAM only works if a taxon occurred several times, only those taxa will be included that occurred more than 5 times (k=3).
min.occ	minimum occurrence: all taxa with less than min.occ will be deleted from the training set

### Details

The 10-fold cross validation is much more slower than the bootstrap or Leave-one-out, because 10 times more wapls-runs must be performed than using e.g. bootstrap (within the same number of runs). The RMSEP of Leave one out is slightly different from C2. In this algorithm before each run of the loop the new training set (each time one sample is taken out) is controlled for zero species and removed (the same procedure as in wa, there C2 does the same). If that row is deleted from the algorithm, the results are equal for LOO. As C2 and R runs with different random numbers, the results of bootstrap and 10 fold cross validation are only equal when using a high number of runs.

### Value

species in train.set	Number of non zero species in each sample of the training set
N2 train.set	Hill's N2 of each sample of the training set
updated opt.	updated optima (see reference)
sample scores	sample scores of the training set
inferred train.set	inferred environmental parameter for the training set
performance	performance of the wa-pls-regression
inferred train.set.val	inferred environmental parameter for the training set using Leave-on-out

`species in core.samples`  
 Number of non zero species in each sample of the core data set  
`n species core.samples in train.set`  
 How many species in the core samples are represented in the training set  
`N2 in core.samples`  
 Hill's N2 of each sample of the core data  
`reconstruction_core.samples`  
 reconstructed environmental parameter for the samples of the core  
`mean(reconstruction_core.samples).val`  
 mean reconstructed environmental parameter for the samples of the core using  
 "boot" or "loo"  
`sd(reconstruction_core.samples).val`  
 standard deviation of the reconstructed environmental parameter for the samples  
 of the core using "boot" or "loo"  
`s1 (boot)` component s1 of the bootstrap  
`s2 (boot)` component s1 of the bootstrap  
`mean(inferred train.set).val`  
 mean inferred environmental variable for the training set using "boot"  
`sd(inferred train.set).val`  
 standard deviation of inferred environmental variable for the training set using  
 "boot"

**Author(s)**

Sven Adler

**References**

ter Braak, C.J.F. & Juggins, S. 1993. Weighted averaging partial least squares regression WA-PLS: an improved method for reconstructing environmental variables from species assemblages. *Hydrobiologia* 269:485-502.

**See Also**

package analogue by G. Simpson

**Examples**

```

data(train_set.MV)
data(train_env.MV)
data(dud.df)
try<-wapls(train_set.MV,train_env.MV,dud.df,val="boot")

```

# Index

## \*Topic **datasets**

age.dud, 3

dud.df, 4

train\_env.MV, 12

train\_set.MV, 12

## \*Topic **hplot**

palplot, 9

## \*Topic **models**

mtf, 4

mw, 6

rand.test, 11

wa, 13

wapls, 15

## \*Topic **package**

paltran-package, 2

age.dud, 3

dud.df, 4

mtf, 4

mw, 6

palplot, 9

paltran (paltran-package), 2

paltran-package, 2

rand.test, 11

train\_env.MV, 12

train\_set.MV, 12

wa, 13

wapls, 15