

Package ‘WAASB’

October 5, 2018

Type Package

Title Weighted Average of Absolute Scores from SVD of BLUP-Interaction Effects

Version 1.0.1

Author Tiago Olivoto

Date 2018-07-05

Maintainer Tiago Olivoto <tiagoolivoto@gmail.com>

Description This package compute stability analysis based on AMMI and BLUP methods. Several procedures, including prediction for AMMI family models and cross-validation for AMMI and BLUP models are available. Tools for personalized graphics is also provided.

Depends R (>= 3.5.0)

License GPL (>=3)

Encoding UTF-8

LazyData true

Imports agricolae, dplyr, plyr, data.table, dendextend, heatmaply, gplots, ggExtra, grid, ggplot2, ggrepel, lme4, broom, gtools, magrittr

RoxygenNote 6.0.1

Suggests knitr, rmarkdown, testthat, xlsx, readxl, rJava, ReporteRs

URL <https://github.com/TiagoOlivoto/WAASB>

VignetteBuilder knitr

RemoteType github

RemoteHost https://api.github.com

RemoteRepo WAASB

RemoteUsername TiagoOlivoto

RemoteRef master

RemoteSha 45436552502c28a02b3192b990ea5bc35236c41f

GithubRepo WAASB

GithubUsername TiagoOlivoto

GithubRef master

GithubSHA1 45436552502c28a02b3192b990ea5bc35236c41f

R topics documented:

INT	2
MEANS	2
plot.blup	3
plot.eigen	4
plot.scores	5
plot.validation.AMMIF	8
plot.WAASBY	9
plot.WAASBYratio	10
predict.AMMI	12
predict.AMMImean	13
summary.WAASB	13
theme_waasb	14
validation.AMMI	14
validation.AMMIF	16
validation.blup	17
WAAS.AMMI	18
WAASB	19
WAASBdata	21
WAASBYratio	21
WAASratio.AMMI	22
Index	25

INT	<i>Dataset with genotype-by-environment interaction effects</i>
-----	---

Description

This dataset contains a matrix with interaction effects of 10 genotypes cultivated in 5 environments.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

MEANS	<i>A dataset with means of 10 genotypes cultivated in 5 environments</i>
-------	--

Description

This dataset contains the means for grain yield of 10 genotypes cultivated in 5 environments. The interaction effects for this data is found in [INT](#)

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

plot.blup

*Plot the estimated BLUPs of genotypes***Description**

Plot the predicted BLUP of the genotypes.

Usage

```
## S3 method for class 'blup'
plot(x, prob = 0.95, export = FALSE, file.type = "pdf",
      file.name = NULL, theme = theme_waasb(), width = 6, height = 6,
      size.err.bar = 0.5, size.shape = 3.5, height.err.bar = 0.3,
      x.lim = NULL, x.breaks = waiver(), leg.pos = c(0.9, 0.1),
      col.shape = c("blue", "red"), y.lab = "Genotypes",
      x.lab = "Predicted Grain Yield", resolution = 300, ...)
```

Arguments

x	The WAASB object
prob	The probability to be considered in confidence interval prediction.
export	Export (or not) the plot. Default is TRUE.
file.type	If export = TRUE, define the type of file to be exported. Default is pdf, Graphic can also be exported in *.tiff format by declaring file.type = "tiff".
file.name	The name of the file for exportation, default is NULL, i.e. the files are automatically named.
theme	The graphical theme of the plot. Default is 'theme = theme_waasb()'. Please, see '?WAASB::theme_waasb'. An own theme can be applied using the arguments: 'theme = theme_waasb() + theme(some stuff here)'. For more details, please, see '?ggplot2::theme'
width	The width "inch" of the plot. Default is 6.
height	The height "inch" of the plot. Default is 6.
size.err.bar	The size of the error bar for the plot. Default is 0.5.
size.shape	The size of the shape (both for genotypes and environments). Default is 3.5.
height.err.bar	The height for error bar. Default is 0.3.
x.lim	The range of x-axis. Default is NULL (maximum and minimum values of the data set). New arguments can be inserted as x.lim = c(x.min, x.max).
x.breaks	The breaks to be plotted in the x-axis. Default is automatic breaks. New arguments can be inserted as x.breaks = c(breaks)
leg.pos	A vector of length 2 to configure, respectively, the horizontal and vertical, position of the legend in the plot. Default is c(0.9, 0.1) (bottom-right).
col.shape	A vector of length 2 that contains the color of shapes for genotypes above and below of the mean, respectively. Default is c("blue", "red").
x.lab	The label of the x-axis in the plot. Default is "Predicted Grain Yield".
y.lab	The label of the y-axis in the plot. Default is "Genotypes".
resolution	The resolution of the plot. Parameter valid if file.type = "tiff" is used. Default is 300 (300 dpi)
...	Other arguments of the function

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#), [plot.WAASBY](#)

Examples

```
library(WAASB)
model = WAASB(WAASBdata, resp = "GY", weight.response = 50, weight.WAAS = 50)
plot.blup(model)
```

plot.eigen

Plot the eigenvalues

Description

Plot the eigenvalues for from singular value decomposition of BLUP interaction effects matrix.

Usage

```
## S3 method for class 'eigen'
plot(x, export = FALSE, theme = theme_waasb(),
     file.type = "pdf", file.name = NULL, width = 6, height = 6,
     size.shape = 3.5, size.line = 1, col.shape = "blue",
     col.line = "blue", y.lab = "Eigenvalue",
     x.lab = "Number of multiplicative terms", resolution = 300, ...)
```

Arguments

x	The WAASB object
export	Export (or not) the plot. Default is TRUE.
theme	The graphical theme of the plot. Default is 'theme = theme_waasb()'. Please, see '?WAASB::theme_waasb'. An own theme can be applied using the arguments: 'theme = theme_waasb() + theme(some stuff here)'. For more details, please, see '?ggplot2::theme'
file.type	If export = TRUE, define the type of file to be exported. Default is pdf, Graphic can also be exported in *.tiff format by declaring file.type = "tiff".
file.name	The name of the file for exportation, default is NULL, i.e. the files are automatically named.
width	The width "inch" of the plot. Default is 6.
height	The height "inch" of the plot. Default is 6.
size.shape	The size of the shape. Default is 3.5.
size.line	The size of the line. Default is 1.
col.shape	The color for shape. Default is c("blue").

<code>col.line</code>	The color for line. Default is <code>c("blue")</code> .
<code>y.lab</code>	The label of the y-axis in the plot. Default is "Eigenvalue".
<code>x.lab</code>	The label of the x-axis in the plot. Default is "Number of multiplicative terms".
<code>resolution</code>	The resolution of the plot. Parameter valid if <code>file.type = "tiff"</code> is used. Default is 300 (300 dpi)
<code>...</code>	Other arguments of the function

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#), [plot.WAASBY](#)

Examples

```
library(WAASB)
model = WAASB(WAASBdata, resp = "GY", weight.response = 50, weight.WAAS = 50)
plot.eigen(model)
```

plot.scores

Plot scores in different graphical interpretations

Description

Plot scores of genotypes and environments in different graphics. 1 = PC1 x PC2, 2 = GY x PC1, 3 = GY x WAASB, and 4 = Nominal yield x EPCA1.

Usage

```
## S3 method for class 'scores'
plot(x, type, file.type = "pdf", export = FALSE,
     file.name = NULL, theme = theme_waasb(), width = 8, height = 7,
     x.lim = NULL, x.breaks = waiver(), x.lab = NULL, y.lab = NULL,
     y.lim = NULL, y.breaks = waiver(), shape.gen = 21, shape.env = 23,
     size.shape = 3.5, size.tex = 3.5, size.line = 0.5, size.segm.line = 0.5,
     leg.lab = c("Gen", "Env"), line.type = "dashed", line.alpha = 0.9,
     col.line = "gray", col.gen = "gray75", col.env = "red", col.alpha = 0.9,
     col.segm.gen = "transparent", col.segm.env = "grey50", resolution = 300,
     ...)
```


<code>col.line</code>	The color of the line that indicate the means in the biplot. Default is "gray"
<code>col.gen</code>	The shape color for genotypes. Default is "gray75". Other values can be attributed. For example, "transparent", will make a plot with only an outline around the shape area.
<code>col.env</code>	The shape color for environments. Default is "red". The same usability than "col.gen".
<code>col.alpha</code>	The alpha value for the color. Default is 0.9. Values must be between 0 (full transparency) to 1 (full color).
<code>col.segm.gen</code>	The color of segment for genotypes. Default is "transparent". Parameter valid for <code>type = 1</code> and <code>type = 2</code> graphics. This segment start in the origin of the biplot and end in the scores values.
<code>col.segm.env</code>	The color of segment for environments. Default is "gray50". The same usability than "col.segm.gen"
<code>resolution</code>	The resolution of the plot. Parameter valid if <code>file.type = "tiff"</code> is used. Default is 300 (300 dpi)
<code>...</code>	Other arguments of the function

Details

The plots type 1 and 2 have the same interpretation than those used in traditional-usage AMMI analysis (well know as AMMI2 and AMMI1, respectively). In the plot type 3, the scores of both genotypes and environments are plotted considering the response variable and the WAASB (stability index that considers all significant principal component axis of traditional AMMI models or all principal component axis estimated with BLUP-interaction effects. Plot type 4 may be used to better understand the well known "which-won-where" pattern, facilitating the recommendation of appropriate genotypes targeted for specific environments, thus allowing the exploitation of narrow adaptations.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.WAASBY](#)

Examples

```
library(WAASB)
model = WAASB(WAASBdata, resp = "GY", weight.response = 50, weight.WAAS = 50)
plot.scores(model, 1, file.type = "tiff", width = 8, height = 7, col.gen = "green")
# PC1 x PC2.tiff

plot.scores(model, 2, width = 8, height = 7, col.gen = "green", shape.gen = 23)
# GY x PC1.pdf

plot.scores(model, 3, col.gen = "blue", col.env = "red",
  theme = theme_waasb() + theme(axis.title = element_text(size = 14)))
# GY x WAAS.pdf

plot.scores(model, 4)
# no file exported
```

plot.validation.AMMIF *Plot the RMSE of all AMMI-family tested models*

Description

Plot the Root Means Square Error of all AMMI-family tested models

Usage

```
## S3 method for class 'validation.AMMIF'
plot(x, violin = TRUE, export = FALSE,
     file.type = "pdf", file.name = NULL, theme = theme_waasb(),
     width = 6, height = 6, resolution = 300, col.violin = "gray90",
     col.boxplot = "gray70", width.boxplot = 0.2, x.lim = NULL,
     x.breaks = waiver(), ...)
```

Arguments

x	A validation.AMMIF object
violin	Define if a violine plot is used with boxplot. Default is "TRUE"
export	Export (or not) the plot. Default is T.
file.type	The type of file to be exported. Default is pdf, Graphic can also be exported in *.tiff format by declaring file.type = "tiff".
file.name	The name of the file for exportation, default is NULL, i.e. the files are automatically named.
theme	The graphical theme of the plot. Default is 'theme = theme_waasb()'. Please, see '?WAASB::theme_waasb'. An own theme can be applied using the arguments: 'theme = theme_waasb() + theme(some stuff here)'. For more details, please, see '?ggplot2::theme'
width	The width "inch" of the plot. Default is 6.
height	The height "inch" of the plot. Default is 6.
resolution	The resolution of the plot. Parameter valid if file.type = "tiff" is used. Default is 300 (300 dpi)
col.violin	Parameter valid if violin = T. Define the color of the violin plot. Default is "gray90.
col.boxplot	Define the color for boxplot. Default is "gray70".
width.boxplot	The width of boxplots. Default is 0.2.
x.lim	The range of x-axis. Default is NULL (maximum and minimum values of the data set). New arguments can be inserted as x.lim = c(x.min, x.max).
x.breaks	The breaks to be plotted in the x-axis. Default is automatic breaks. New arguments can be inserted as x.breaks = c(breaks)
...	Other arguments of the function

Details

Five statistics are shown in this type of plot. The lower and upper hinges correspond to the first and third quartiles (the 25th and 75th percentiles). The upper whisker extends from the hinge to the largest value no further than $1.5 * \text{IQR}$ from the hinge (where IQR is the inter-quartile range). The lower whisker extends from the hinge to the smallest value at most $1.5 * \text{IQR}$ of the hinge. Data beyond the end of the whiskers are considered outlying points.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#), [plot.WAASBY](#)

Examples

```
library(WAASB)
validation = validation.AMMIF(WAASBdata, resp = "GY", nboot = 100, nrepval = 2)

#A plot without a violin plot exported to a *.pdf file
plot.validation.AMMIF(validation, violin = F, export = T)
```

plot.WAASBY	<i>Plot WAASBY values for genotype ranking</i>
-------------	--

Description

Plot heat maps with genotype ranking in two ways.

Usage

```
## S3 method for class 'WAASBY'
plot(x, export = F, file.type = "pdf", file.name = NULL,
     theme = theme_waasb(), width = 6, height = 6, size.shape = 3.5,
     col.shape = c("blue", "red"), y.lab = "Genotypes", x.breaks = waiver(),
     resolution = 300, ...)
```

Arguments

x	The WAASBY object
export	Export (or not) the plot. Default is T.
file.type	The type of file to be exported. Default is pdf, Graphic can also be exported in *.tiff format by declaring file.type = "tiff".
file.name	The name of the file for exportation, default is NULL, i.e. the files are automatically named.
theme	The graphical theme of the plot. Default is 'theme = theme_waasb()'. Please, see '?WAASB::theme_waasb'. An own theme can be applied using the arguments: 'theme = theme_waasb() + theme(some stuff here)'. For more details, please, see '?ggplot2::theme'

width	The width "inch" of the plot. Default is 8.
height	The height "inch" of the plot. Default is 7.
size.shape	The size of the shape in the plot. Default is 3.5.
col.shape	A vector of length 2 that contains the color of shapes for genotypes above and below of the mean, respectively. Default is c("blue", "red").
y.lab	The label of the y-axis in the plot. Default is "Genotypes".
x.breaks	The breaks to be plotted in the x-axis. Default is automatic breaks. New arguments can be inserted as x.breaks = c(breaks)
resolution	The resolution of the plot. Parameter valid if file.type = "tiff" is used. Default is 300 (300 dpi)
...	Other arguments of the function

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#)

Examples

```
library(WAASB)
# plotting the WAASBY values based on blup procedure
#model = WAASBYratio(WAASBdata, resp = "GY", increment = 5, saveWAASY = 60)
#plot.WAASBY(model)

# plotting the WAASY value based on traditional AMMI model
#model2 = WAASratio.AMMI(WAASBdata, resp = "GY", increment = 5, saveWAASY = 60)
#plot.WAASBY(model2)
```

plot.WAASBYratio	<i>Plot heat maps with genotype ranking</i>
------------------	---

Description

Plot heat maps with genotype ranking in two ways.

Usage

```
## S3 method for class 'WAASBYratio'
plot(x, type, export = FALSE, file.type = "pdf",
     file.name = NULL, width = 6, height = 5, size.lab = 1,
     margins = c(5, 4), y.lab = NULL, x.lab = NULL,
     key.lab = "Genotype ranking", resolution = 300, ...)
```

Arguments

x	The WAASYratio object
type	1 = Heat map Ranks: this graphic shows the genotype ranking considering the WAAS estimated with different numbers of Principal Components; 2 = Heat map WAASY-GY ratio: this graphic shows the genotype ranking considering the different combinations in the WAAS/GY ratio.
export	Export (or not) the plot. Default is TRUE.
file.type	If export = TRUE define the type of file to be exported. Default is pdf, Graphic can also be exported in *.tiff format by declaring file.type = "tiff".
file.name	The name of the file for exportation, default is NULL, i.e. the files are automatically named.
width	The width "inch" of the plot. Default is 8.
height	The height "inch" of the plot. Default is 7.
size.lab	The label size of the plot. It is suggested attribute 1
margins	Numeric vector of length 2 containing the margins for column and row names, respectively. Default is c(5, 4).
y.lab	The label of y axis. Default is "Genotypes".
x.lab	The label of x axis. Default is "Number of axes".
key.lab	The label of color key. Default is "Genotype ranking".
resolution	Valid parameter if file.type = "tiff". Define the resolution of the plot. Default is "300".
...	Other arguments of the function

Details

The first type of heatmap shows the genotype ranking depending on the number of principal component axis used for estimating the WAASB index. An euclidian distance-based dendrogram is used for grouping the genotype ranking for both genotypes and principal component axis. The second type of heatmap shows the genotype ranking depending on the WAASB/GY ratio. The ranks obtained with a ratio of 100/0 considers exclusively the stability for the genotype ranking. On the other hand, a ratio of 0/100 considers exclusively the productivity for the genotype ranking. Four clusters are estimated (1) unproductive and unstable genotypes; (2) productive, but unstable genotypes; (3) stable, but unproductive genotypes; and (4), productive and stable genotypes.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#)

Examples

```
library(WAASB)
model = WAASBYratio(WAASBdata, resp = "GY")

# plot the heatmap considering the ranks in different number of PC axis.
plot.WAASBYratio(model, type = 1)
```

```
# plot the heatmap considering the ranks in WAASY/GY ratio and export to a *.tiff file
plot.WAASBYratio(model, type = 2, export = TRUE, file.type = "tiff")
```

predict.AMMI	<i>Predict means in AMMI analysis</i>
--------------	---------------------------------------

Description

Predict the means of i -genotypes in j -environment considering a specific number of axis.

Usage

```
## S3 method for class 'AMMI'
predict(object, resp, naxis, ...)
```

Arguments

object	The data set. Must be in the following order. Environment, Genotype, Block/Rep, and response variable(s).
resp	The response variable, for example <code>resp = "RG"</code> .
naxis	The the number of axis to be use in the prediction.
...	Additional parameter for the function

Details

This function is used to predict the response variable of a two-way table (for examples the yielding of the i -th genotype in the j -th environment) based on AMMI model. This prediction is based on the number of multiplicative terms used. If `naxis = 0`, only the main effects (AMMI0) are used. In this case, the predicted mean will be the predicted value from OLS estimation. If `naxis = 1` the AMMI1 (with one multiplicative term) is used for predicting the response variable. If `naxis = min(gen-1; env-1)`, the AMMIF is fitted and the predicted value will be the cell mean, i.e. the mean of R -replicates of the i -th genotype in the j -th environment. The number of axis to be used must be carefully chosen. Precures based on Postdictive sucess (such as Gollob's d.f.) or Predictive sucess (such as cross-validation) should be used to do this. This package provide both. [WAAS.AMMI](#) function compute traditional AMMI analysis showing the number of significant axis. On the other hand, [validation.AMMIF](#) function provide a cross-validation, estimating the RMSE of all AMMI-family models, based on resampling procedure.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[predict.AMMImean](#)

Examples

```
library(WAASB)
predict = predict.AMMI(WAASBdata, resp = "GY", naxis = 3)
```

predict.AMMImean	<i>Predict means in AMMI analysis with data from single observation</i>
------------------	---

Description

Predict the means of i-genotypes in j-environment considering a specific number of axis.

Usage

```
## S3 method for class 'AMMImean'
predict(object, resp, naxis, ...)
```

Arguments

object	The data set. Must be in the following order. Environment, Genotype, and response variable(s).
resp	The response variable, for example resp = "RG".
naxis	The the number of axis to be use in the prediction. If NAXIS = 0, the predicted mean is the predicted value of OLS estimation. If NAXIS = "min(gen-1;env-1)", the predicted value will be the cell mean, i.e. the mean of R-replicates of the i-th genotype in the j-th environment
...	Additional parameter for the function

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[predict.AMMI](#)

Examples

```
library(WAASB)
predmean = predict.AMMImean(MEANS, resp = "Resp", naxis = 3)
```

summary.WAASB	<i>Summary a WAASB object</i>
---------------	-------------------------------

Description

Summary the WAASB object in two ways. By default, the results are shown in the R console. The results can also be exported to the directory.

Usage

```
## S3 method for class 'WAASB'
summary(object, export = FALSE, file.name = NULL, digits = 4, ...)
```

Arguments

<code>object</code>	The WAASB object
<code>export</code>	A logical argument. If TRUE T, a *.txt file is exported to the working directory
<code>file.name</code>	The name of the file if <code>export = TRUE</code>
<code>digits</code>	The significant digits to be shown.
<code>...</code>	Other arguments of the function

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

Examples

```
library(WAASB)
model = WAASB(WAASBdata, resp = "GY", prob = 0.95, weight.response = 50, weight.WAAS = 50)
summary.WAASB(model, export = T, file.name = "my results")
```

<code>theme_waasb</code>	<i>A personalized theme for the WAASB ggplot2-based graphics</i>
--------------------------	--

Description

The function `theme_waasb()` is a personalized theme for the WAASB ggplot2-based graphics. An own theme can be applied in all graphics by using the following arguments in the `plot. functions` of the package: `theme = theme_waasb() + theme(some stuff here...)`. See `?ggplot2::theme` for more details.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

<code>validation.AMMI</code>	<i>Cross-validation for estimation of AMMI model</i>
------------------------------	--

Description

Cross-validation for estimation of AMMI models

Usage

```
validation.AMMI(data, resp, design = "RCBD", nboot, nrepval, naxis, progbar = TRUE)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s)
resp	The response variable, for example <code>resp = "RG"</code> .
design	The experimental design to be considered. Default is RCBD (Randomized complete Block Design). For Completely Randomized Designs inform <code>design = "CRD"</code> .
nboot	The number of resamples to be used in the cross-validation
nrepval	The number of replicates (r) from total number of replicates (R) to be used in the modeling dataset. Only one replicate is used as validating data each step, so, Nrepval must be equal R-1
naxis	The number of axis to be considered for estimation of GE effects.
progbar	A logical argument to define if a progress bar is shown. Default is TRUE.

Details

For each iteration, the original dataset is split into two datasets: modeling and validating data. The dataset "modeling" has all combinations (genotype x environment) with the number of replications informed in `nrepval`. The dataset "validating" has one replication. The splitting of the dataset into modeling and validating data depends on the design informed. For Completely Randomized Block Design (default), completely blocks are selected within environments. The remained block serves validation data. If `design = "RCD"` is informed, completely randomly samples are made for each genotype-by-environment combination. The estimated values (depending on NAXIS informed) are compared with the "validating" data. the Root Means Square error is computed. At the end of boots, a list is returned with the following values.

Value

RMSE	A vector with Nboot-estimates of the root mean squared error estimated with the difference between predicted and validating data.
RSMEmean	The mean of RMSE estimates.
Estimated	A data frame that contain the values (predicted, observed, validation) of the last loop.
Modeling	The dataset used as modeling data in the last loop.
Testing	The dataset used as testing data in the last loop.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[validation.AMMIF](#)

Examples

```
library(WAASB)
validation = validation.AMMI(WAASBdata, resp = "GY", nboot = 100, nrepval = 2, naxis = 4)
```

validation.AMMIF

*Cross-validation for estimation of all AMMI-family models***Description**

Cross-validation for estimation of all AMMI-family models

Usage

```
validation.AMMIF(data, resp, design = "RCBD", nboot, nrepval, progbars = TRUE)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s)
resp	The response variable, for example resp = "RG".
design	The experimental design to be considered. Default is RCBD (Randomized complete Block Design). For Completely Randomized Designs inform design = "CRD".
nboot	The number of resamples to be used in the cross-validation
nrepval	The number of replicates (r) from total number of replicates (R) to be used in the modeling dataset. Only one replicate is used as validating data each step, so, Nrepval must be equal R-1
progbars	A logical argument to define if a progress bar is shown. Default is TRUE.

Details

This function provides a complete cross-validation of replicate-based data using AMMI-family models. Automatically the first validation is carried out considering the AMMIF (all possible axis used). Considering this model, the original dataset is split into two datasets: modeling and validating data. The dataset "modeling" has all combinations (genotype x environment) with the number of replications informed in nrepval. The dataset "validating" has one replication. The splitting of the dataset into modeling and validating data depends on the design informed. For Completely Randomized Block Design (default), completely blocks are selected within environments. The remained block serves validation data. If design = "RCD" is informed, completely randomly samples are made for each genotype-by-environment combination. The estimated values (depending on naxis informed) are compared with the "validating" data. the Root Means Square error is computed. At the end of boots, a list is returned with the following values.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[validation.AMMI](#)

Examples

```
library(WAASB)
validation = validation.AMMIF(WAASBdata, resp = "GY", nboot = 100, nrepval = 2)
```

validation.blup	<i>Cross-validation for blup prediction</i>
-----------------	---

Description

Cross-validation for blup prediction.

Usage

```
validation.blup(data, resp, nboot, nrepval, progbar = TRUE)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s).
resp	The response variable, for example <code>resp = "RG"</code> .
nboot	The number of resamples to be used in the cross-validation
nrepval	The number of replicates (r) from total number of replicates (R) to be used in the modeling dataset. Only one replicate is used as validating data each step, so, Nrepval must be equal R-1
progbar	A logical argument to define if a progress bar is shown. Default is TRUE.

Details

This function provides a cross-validation of replicate-based data using mixed models. By default, complete blocks are randomly selecte each environment. In each interaction, the original dataset is split into two datasets: modeling and validating data. The dataset "modeling" has all combinations (genotype x environment) with the number of replications informed in nrepval. The dataset "validating" has one replication. The estimated values are compared with the "validating" data. the Root Means Square error is computed. At the end of boots, a list is returned.

Value

RMSE	The root mean squared error estimated with the difference between predicted and validating data.
------	--

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

See Also

[plot.scores](#), [plot.WAASBY](#)

Examples

```
library(WAASB)
validation = validation.blup(WAASBdata, resp = "GY", nboot = 100, nrepval = 2)
```

WAAS.AMMI

*Weighted Average of Absolute Scores for AMMI analysis***Description**

Compute the Weighted Average of Absolute Scores for AMMI analysis.

Usage

```
WAAS.AMMI(data, resp, p.valuePC = 0.05, naxis = NULL,
           weight.response = 50, weight.WAAS = 50)
```

Arguments

<code>data</code>	The data frame. Data must be in the following order: Environment, Genotype, Block, Response variable.
<code>resp</code>	The response variable, for example <code>resp = "RG"</code> .
<code>p.valuePC</code>	The p-value for considering the PC significant. The number of significant Principal Components to be used for calculating the WAAS will be chosen based on this probability.
<code>naxis</code>	The number of axis to be used for computing the WAAS value. Default it NULL. If informed some values, the number of axis informed is used for computing the WAAS.
<code>weight.response</code>	The weight for the response variable.
<code>weight.WAAS</code>	The weight for the WAAS (stability).

Details

This function compute the weighted average of absolute scores considering the significant principal component axis. The number of axis to be used depends on the probability level informed in `p.valuePC`. For example if `p.valuePC = 0.05`, all axis that are significant considering this probability level are used. The number of axis can be also informed by declaring `naxis = x`. This comand ignores the `p.valuePC` comand.

Value

<code>individual</code>	A within-environments ANOVA considering a fixed-effect model.
<code>model</code>	A data frame with the response variable, the scores of all Principal Components, the estimates of Weighted Average of Absolute Scores, and WAASY (the index that consider the weights for stability and productivity in the genotype ranking).
<code>MeansGxE</code>	The means of genotypes in the environments, with observed, predicted and residual values.
<code>PCA</code>	Principal Component Analysis.
<code>anova</code>	Joint analysis of variance for the main effects and Principal Component analysis of the interaction effect.

Details A list summarizing the results. The following information are showed. WgtResponse, the weight for the response variable in estimating WAASB, WgtWAAS the weight for stability, Ngen the number of genotypes, Nenv the number of environments, OVmean the overall mean, Min the minimum observed (returning the genotype and environment), Max the maximum observed, Max the maximum observed, MinENV the environment with the lower mean, MaxENV the environment with the larger mean observed, MinGEN the genotype with the lower mean, MaxGEN the genotype with the larger.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

Examples

```
# considering p-value <= 0.05 to compute the WAAS
library(WAASB)
model = WAAS.AMMI(WAASBdata, resp = "GY", weight.response = 40, weight.WAAS = 60)

# considering declaring the number of axis to be used for computing WAAS
library(WAASB)
model = WAAS.AMMI(WAASBdata, resp = "GY", naxis = 3, weight.response = 40, weight.WAAS = 60)
```

WAASB

Weighted Average of Absolute Scores for BLUP-based PCA analysis

Description

Compute the Weighted Average of Absolute Scores for PCA analysis based on the GE-interaction matrix.

Usage

```
WAASB(data, resp, random = "gen", prob = 0.95, weight.response = 50, weight.WAAS = 50)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s)
resp	The response variable, for example resp = "RG".
random	The effect to be considered as random. Default is genotype "gen". For considering both genotypes and environments as random inform random = "all".
prob	The probability for estimating confidence interval for BLUP's prediction.
weight.response	The weight for the response variable. Defatul is 50
weight.WAAS	The weight for the WAAS (stability). Default is 50

Details

This function compute the weighted average of absolute scores considering all principal component axis from Singular Value Decomposition of BLUP-interaction effects matrix. As the matrix of genotypic effects (free from error and interaction effects) is used, all multiplicative terms can be used without a noise-adding effect. The main advantage of this procedure in relation to the WAAS.AMMI function is that, by using mixed model-based analysis, random effects can be included in the model. In addition, unbalanced datasets can also be modeled.

Value

<code>individual</code>	A within-environments ANOVA considering a fixed-effect model.
<code>model</code>	A data frame with the response variable, the scores of all Principal Components, the estimates of Weighted Average of Absolute Scores, and WAASY (the index that consider the weights for stability and productivity in the genotype ranking).
<code>BLUPgen</code>	The estimated BLUPS for genotypes.
<code>BLUPgge</code>	The estimated BLUPS of all genotypes in all environments "BLUPij".
<code>PCA</code>	The results of Principal Component Analysis with eigenvalues and explained variance of BLUP-interaction matrix.
<code>MeansGxE</code>	The phenotypic means of genotypes in the environments, with observed, predicted and OLS residual prediction.
<code>Details</code>	A list summarizing the results. The following information are showed. <code>WgtResponse</code> , the weight for the response variable in estimating WAASB, <code>WgtWAAS</code> the weight for stability, <code>Ngen</code> the number of genotypes, <code>Nenv</code> the number of environments, <code>OVmean</code> the overall mean, <code>Min</code> the minimum observed (returning the genotype and environment), <code>Max</code> the maximum observed, <code>MinENV</code> the environment with the lower mean, <code>MaxENV</code> the environment with the larger mean observed, <code>MinGEN</code> the genotype with the lower mean, <code>MaxGEN</code> the genotype with the larger.
<code>REML</code>	A list with the following data frames: <code>fixed</code> (estimates for fixed effects); <code>random</code> (the variance components for random effects); <code>statistics</code> (some statistics such as <code>sigma</code> , <code>logLik</code> , <code>AIC</code> , <code>BIC</code> , <code>deviance</code> ...)
<code>ESTIMATES</code>	A list with the following values: <code>GEV</code> the genotype-by-environment variance (and percentage of phenotypic variance); <code>GV</code> the genotypic variance (and percentage of phenotypic variance); <code>EV</code> the environmental variance; <code>RV</code> the residual variance (and percentage of phenotypic variance); <code>FV</code> the phenotypic variance; <code>h2g</code> the heritability of the trait; <code>GEr2</code> the coefficient of determination of the interaction effects; <code>h2mg</code> the heritability of the mean; <code>AccuGen</code> the selective accuracy; <code>rge</code> the genotype-environment correlation; <code>CVg</code> the genotypic coefficient of variation; <code>CVr</code> the residual coefficient of variation; <code>CVratio</code> the ratio between genotypic and residual coefficient of variation.
<code>LRT</code>	The Likelihood Ratio Test for the random effects of the model.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

Examples

```
# solely genotypes as random effects and equal weights for both response variable and stability
#library(WAASB)
```

```
#model = WAASB(WAASBdata, resp = "GY", prob = 0.95)

# both genotypes and environments as random effects and a higher weight for stability
#library(WAASB)
#model = WAASB(WAASBdata, resp = "GY", weight.response = 70, weight.WAAS = 30)
```

WAASBdata

*A replicate-based data of 10 genotypes cultivated in 11 environments***Description**

This dataset contains the observed grain yield of 10 genotypes cultivated in 11 environments. The experimental design was a RCBD with 3 replicates (blocks). This data provides examples for several functions of the WAAS package.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

WAASBYratio

*Weighted Average of Absolute Scores from SVD of BLUP-interaction effects matrix in different scenarios of WAASB/GY ratio***Description**

Compute the Weighted Average of Absolute Scores for AMMI analysis in different combinations of weights for stability and productivity.

Usage

```
WAASBYratio(data, resp, increment = 5, saveWAASY = 50, progbar = TRUE)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s)
resp	The response variable, for example <code>resp = "RG"</code> .
increment	The range of the increment for WAASB/GY ratio. Default is 5. The function computes the WAASBY values starting with a weight of 100 for stability and 0 for response variable. With the default, the first scenario will be a WAASB/GY ratio = 100/0. In the next scenario, the WAASBY values are computed based on a WAASB/GY ratio = 95/5.
saveWAASY	Automatically save the WAASY values when the weight for WAAS (stability) in the WAAS/GY ratio is "saveWAASY". Default is 100. The value of "saveWAASY" must be multiple of "Increment". If this assumption is not valid, an error will occur.
progbar	A logical argument to define if a progress bar is shown. Default is TRUE.

Details

This function considers both stability (weighted average of absolute scores based on SVD of BLUP-interaction effects matrix) and productivity for genotype ranking. This function provide the option of attributing weights for stability and productive in genotype ranking. This is important depending on the goal of a selection strategy. For example, if a goal of a breeding program is to select a genotype with high yielding (independently on the stability performance), that genotype with the first rank in an $WAASB/GY = 0/100$ ratio should be selected. The reciprocal is true. Aiming at selecting a high-stable genotype (independently on the productivity), that genotype with the first rank in an $WAASB/GY = 100/0$ ratio should be selected. By default, the increment on the $WAASB/GY$ ratio is equal to 5. In other words, twenty one different combinations are computed. Each combination, the genotypes are ranked regarding the WAASY value.

Value

anova	Joint analysis of variance for the main effects and Principal Component analysis of the interaction effect.
PC	Principal Component Analysis.
MeansGxE	The means of genotypes in the environments, with observed, predicted and residual values.
WAAS	A data frame with the response variable, the scores of all Principal Components, the estimates of Weighted Average of Absolute Scores, and WAASY (the index that consider the weights for stability and productivity in the genotype ranking).
WAASY	The values of the WAASY estimated when the weight for the stability in the loop match with argument "saveWAASY".
WAASY.values	All the values of WAASY estimated in the different scenarios of WAAS/GY weighting ratio.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

Examples

```
# Default, with increment of 5 and saving the WAASY values when weight is 50
#library(WAASB)
#model= WAASBYratio(WAASBdata, resp = "GY")

# Incrementing 2-by-2 and saving the WAASY values when weight is 70
#library(WAASB)
#model= WAASBYratio(WAASBdata, resp = "GY", increment = 2, saveWAASY = 70)
```

WAASratio.AMMI	<i>Weighted Average of Absolute Scores for AMMI analysis in different scenarios of WAAS/GY ratio</i>
----------------	--

Description

Compute the Weighted Average of Absolute Scores for AMMI analysis in different combinations of weights for stability and productivity.

Usage

```
WAASratio.AMMI(data, resp, p.valuePC = 0.05, increment = 5, saveWAASY = 50,
  progar = TRUE)
```

Arguments

data	The dataset. The input must be in the following order: Environment, Genotype, replication/block and response variable(s)
resp	The response variable, for example <code>resp = "RG"</code> .
p.valuePC	The p-value for considering the PC significant. Default is 0.05. The number of significant Principal Components to be used for calculating the WAASB will be chosen based on this probability.
increment	The range of the increment for WAAS/GY ratio. Default is 5. The function compute the WAASY values starting with a weight o 100 for stability and 0 for response variable. With the default, the first scenario will be a WAAS/GY ratio = 100/0. In the next scenario, the WAASY values are computed based on a WAAS/GY ratio = 95/5.
saveWAASY	Automatically save the WAASY values when the wheight for WAAS (stability) in the WAAS/GY ratio is "saveWAASY". Default is 100. The value of "save-WAASY" must be multiple of "Increment". If this assumption is not valid, an error will be occur.
progar	A logical argument to define if a progress bar is shown. Default is TRUE.

Details

This function is very similar to the `WAASBYratio`. The main difference is that here, the `WAASBY` values are computed considering a traditional AMMI model.

Value

anova	Joint analysis of variance for the main effects and Principal Component analysis of the interaction effect.
PC	Principal Component Analysis.
MeansGxE	The means of genotypes in the environments, with observed, predicted and residual values.
WAAS	A data frame with the response variable, the scores of all Principal Components, the estimates of Weighted Average of Absolute Scores, and WAASY (the index that consider the weights for stability and productivity in the genotype ranking.
WAASY	The values of the WAASY estimated when the wheight for the stability in the loop match with argument "saveWAASY".
WAASY.values	All the values of WAASY estimated in the different scenarios of WAAS/GY weighting ratio.

Author(s)

Tiago Olivoto <tiagoolivoto@gmail.com>

Examples

```
# Default, with increment of 5 and saving the WAASY values when weight is 50
#library(WAASB)
#model= WAASratio.AMMI(WAASBdata, resp = "GY")

# Incrementing 2-by-2 and saving the WAASY values when weight is 70
#library(WAASB)
#model= WAASratio.AMMI(WAASBdata, resp = "GY", increment = 2, saveWAASY = 70)
```


Index

INT, [2](#), [2](#)

MEANS, [2](#)

plot.blup, [3](#)

plot.eigen, [4](#)

plot.scores, [4](#), [5](#), [5](#), [9–11](#), [17](#)

plot.validation.AMMIF, [8](#)

plot.WAASBY, [4](#), [5](#), [7](#), [9](#), [9](#), [17](#)

plot.WAASBYratio, [10](#)

predict.AMMI, [12](#), [13](#)

predict.AMMImean, [12](#), [13](#)

summary.WAASB, [13](#)

theme_waasb, [14](#)

validation.AMMI, [14](#), [16](#)

validation.AMMIF, [12](#), [15](#), [16](#)

validation.blup, [17](#)

WAAS.AMMI, [12](#), [18](#)

WAASB, [19](#)

WAASBdata, [21](#)

WAASBYratio, [21](#)

WAASratio.AMMI, [22](#)