

# Package: RealVAMS (via r-universe)

September 5, 2024

**Type** Package

**Title** Multivariate VAM Fitting

**Version** 0.4-6

**Date** 2024-04-05

**Description** Fits a multivariate value-added model (VAM), see Broatch, Green, and Karl (2018) <[doi:10.32614/RJ-2018-033](https://doi.org/10.32614/RJ-2018-033)> and Broatch and Lohr (2012) <[doi:10.3102/1076998610396900](https://doi.org/10.3102/1076998610396900)>, with normally distributed test scores and a binary outcome indicator. A pseudo-likelihood approach, Wolfinger (1993) <[doi:10.1080/00949659308811554](https://doi.org/10.1080/00949659308811554)>, is used for the estimation of this joint generalized linear mixed model. The inner loop of the pseudo-likelihood routine (estimation of a linear mixed model) occurs in the framework of the EM algorithm presented by Karl, Yang, and Lohr (2013) <[DOI:10.1016/j.csda.2012.10.004](https://doi.org/10.1016/j.csda.2012.10.004)>. This material is based upon work supported by the National Science Foundation under grants DRL-1336027 and DRL-1336265.

**License** GPL-2

**Depends** R (>= 3.0.0), Matrix

**Imports** numDeriv, Rcpp (>= 0.11.2), methods, stats, utils, grDevices, graphics

**LazyData** yes

**ByteCompile** yes

**NeedsCompilation** yes

**LinkingTo** Rcpp, RcppArmadillo

**Author** Andrew Karl [cre, aut] (<<https://orcid.org/0000-0002-5933-8706>>), Jennifer Broatch [aut], Jennifer Green [aut]

**Maintainer** Andrew Karl <[akar1@asu.edu](mailto:akar1@asu.edu)>

**Date/Publication** 2024-04-05 09:13:06 UTC

**Repository** <https://akar146556.r-universe.dev>

**RemoteUrl** <https://github.com/cran/RealVAMS>

**RemoteRef** HEAD

**RemoteSha** 2bc4b0db2619ef6f07b0b328e9070b82f7f0fd9c

## Contents

RealVAMS-package . . . . .	2
example.outcome.data . . . . .	3
example.score.data . . . . .	5
plot . . . . .	6
print . . . . .	8
RealVAMS . . . . .	8
REML_Rm . . . . .	13
R_mstep2 . . . . .	14
summary . . . . .	15
vp_cp . . . . .	16

**Index** **17**

---

RealVAMS-package	<i>Multivariate VAM Fitting</i>
------------------	---------------------------------

---

## Description

Fits a multivariate value-added model (VAM), see Broatch, Green, and Karl (2018) <doi:10.32614/RJ-2018-033>, and Broatch and Lohr (2012) <doi:10.3102/1076998610396900>, with normally distributed test scores and a binary outcome indicator. A pseudo-likelihood approach, Wolfinger and O'Connell (1993) <doi:10.1080/00949659308811554>, is used for the estimation of this joint generalized linear mixed model. The inner loop of the pseudo-likelihood routine (estimation of a linear mixed model) occurs in the framework of the EM algorithm presented by Karl, Yang, and Lohr (2013) <DOI:10.1016/j.csda.2012.10.004>. This material is based upon work supported by the National Science Foundation under grants DRL-1336027 and DRL-1336265.

## Details

Package:	RealVAMS
Type:	Package
Version:	0.4-6
Date:	2023-04-05
License:	GPL-2

**Author(s)**

Authors: Andrew T. Karl, Jennifer Broatch, and Jennifer Green  
 Maintainer: Andrew Karl <akarl@asu.edu>

**References**

- Broatch, J. and Lohr, S. (2012) <DOI:10.3102/1076998610396900> Multidimensional Assessment of Value Added by Teachers to Real-World Outcomes. *Journal of Educational and Behavioral Statistics* **37**, 256–277.
- Broatch, J., Green, J., Karl, A. (2018) <DOI:10.32614/RJ-2018-033> RealVAMS: An R Package for Fitting a Multivariate Value-added Model (VAM). *The R Journal* 10/1, 22–30.
- Karl, A., Yang, Y. and Lohr, S. (2013) <DOI:10.1016/j.csda.2012.10.004> Efficient Maximum Likelihood Estimation of Multiple Membership Linear Mixed Models, with an Application to Educational Value-Added Assessments. *Computational Statistics & Data Analysis* **59**, 13–27.
- Karl, A., Yang, Y. and Lohr, S. (2013) <DOI:10.3102/1076998613494819> A Correlated Random Effects Model for Nonignorable Missing Data in Value-Added Assessment of Teacher Effects. *Journal of Educational and Behavioral Statistics* **38**, 577–603.
- Karl, A., Yang, Y. and Lohr, S. (2014) <DOI:10.1016/j.csda.2013.11.019> Computation of Maximum Likelihood Estimates for Multiresponse Generalized Linear Mixed Models with Non-nested, Correlated Random Effects. *Computational Statistics & Data Analysis* **73**, 146–162.
- Lockwood, J., McCaffrey, D., Mariano, L., Setodji, C. (2007) <DOI:10.3102/1076998606298039> Bayesian Methods for Scalable Multivariate Value-Added Assessment. *Journal of Educational and Behavioral Statistics* **32**, 125–150.
- Wolfinger, R. (1993) <DOI:10.1080/00949659308811554> Generalized linear mixed models a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* **48** 233–243.

**Examples**

```
data(example.score.data)
data(example.outcome.data)
#The next line exists to show that the function can run and that the package
#installed correctly. This is a CRAN requirement to ensure that the package
#works in future version of R
RealVAMS(example.score.data,example.outcome.data,max.PQL.it=1,max.iter.EM=2,
var.parm.hessian=FALSE)

res<-RealVAMS(example.score.data,example.outcome.data)
```

---

example.outcome.data    *Simulated Data*

---

**Description**

A simulated data set used to illustrate the functionality of the package. This data set represents binary outcome measurements on 625 students (with one missing).

**Usage**

```
data(example.outcome.data)
```

**Format**

A data frame with 624 observations. The data set contains the following 2 variables.

`r` a numeric vector composed of 0's and 1's representing a binary outcome measured on students.

`student` a numeric vector

**Details**

The data set may be reproduced with the following code.

```
set.seed(0)
library(MASS)
#number of years: fixed at 3 for this code block
years<-3
#teacher in each year
teachers<-25
#students in each class
students<-25
alpha<-.5
eta.stu.j <-
mvrnorm(n=teachers*students,mu=c(0,0),Sigma=cbind(c(5,.2),c(.2,.1)))
eta.stu<-eta.stu.j[,1]
eta.stu.r<- eta.stu.j[,2]
z1<-rep(1:teachers,each=students)
z2<-sample(rep(1:teachers,each=students))
z3<-sample(rep(1:teachers,each=students))
cont_var1<-rnorm(students*teachers,0,.5)
cont_var2<-rnorm(students*teachers,0,.5)
cont_var3<-rnorm(students*teachers,0,.5)
gam <- mvrnorm(n=teachers*years,mu=c(0,0),Sigma=cbind(c(5,.6),c(.6,.6)))
eps1<- rnorm(students*teachers,0,sqrt(5))
eps2<- rnorm(students*teachers,0,sqrt(5))
eps3<- rnorm(students*teachers,0,sqrt(5))
gam1<-gam[seq(1,teachers),1]
gam2<-gam[seq((teachers+1),(2*teachers)),1]
gam3<-gam[seq((2*teachers+1),(3*teachers)),1]
gam1.r<-gam[seq(1,teachers),2]
gam2.r<-gam[seq((teachers+1),(2*teachers)),2]
gam3.r<-gam[seq((2*teachers+1),(3*teachers)),2]
y1<-50+ eta.stu+gam1[z1]+cont_var1+eps1
y2<-eta.stu+gam1[z1]*alpha+gam2[z2]+cont_var2+eps2
y3<-100+eta.stu+gam1[z1]*alpha+gam2[z2]*alpha+gam3[z3]+cont_var3+eps3
r1<-rbinom(students*teachers,1,
           pnorm(.1+eta.stu.r+gam1.r[z1]+gam2.r[z2]+gam3.r[z3]))
student<-1:(students*teachers)
```

```
teacher<-c(z1,z2,z3)
cont_var<-c(cont_var1,cont_var2,cont_var3)
year<-c(rep(1:3,each=students*teachers))
y<-c(y1,y2,y3)
vam_data2<-as.data.frame(cbind(student,teacher,year,y,cont_var))
vam_data2<-vam_data2[order(vam_data2$student,vam_data2$year),]
example.score.data<-vam_data2
vam_data2.r<-as.data.frame(cbind(student,r=r1))
example.outcome.data<-vam_data2.r
```

## Examples

```
data(example.outcome.data)
print(example.outcome.data[1,])
```

---

example.score.data      *Simulated Data*

---

## Description

A simulated data set used to illustrate the functionality of the package. The data are simulated according to the VP model.

## Usage

```
data(example.score.data)
```

## Format

A data frame with 1874 observations on 625 students over 3 years, with 25 teachers in each year. The data set contains the following 5 variables.

y a numeric vector representing the student score  
student a numeric vector  
year a numeric vector  
teacher a numeric vector  
cont\_var a numeric vector representing a continuous covariate

## Details

The data set may be reproduced with the following code.

```
set.seed(0)
library(MASS)
#number of years: fixed at 3 for this code block
years<-3
#teacher in each year
```

```

teachers<-25
#students in each class
students<-25
alpha<- .5
eta.stu.j <-
mvrnorm(n=teachers*students,mu=c(0,0),Sigma=cbind(c(5,.2),c(.2,.1)))
eta.stu<-eta.stu.j[,1]
eta.stu.r<- eta.stu.j[,2]
z1<-rep(1:teachers,each=students)
z2<-sample(rep(1:teachers,each=students))
z3<-sample(rep(1:teachers,each=students))
cont_var1<-rnorm(students*teachers,0,.5)
cont_var2<-rnorm(students*teachers,0,.5)
cont_var3<-rnorm(students*teachers,0,.5)
gam <- mvrnorm(n=teachers*years,mu=c(0,0),Sigma=cbind(c(5,.6),c(.6,.6)))
eps1<- rnorm(students*teachers,0,sqrt(5))
eps2<- rnorm(students*teachers,0,sqrt(5))
eps3<- rnorm(students*teachers,0,sqrt(5))
gam1<-gam[seq(1,teachers),1]
gam2<-gam[seq((teachers+1),(2*teachers)),1]
gam3<-gam[seq((2*teachers+1),(3*teachers)),1]
gam1.r<-gam[seq(1,teachers),2]
gam2.r<-gam[seq((teachers+1),(2*teachers)),2]
gam3.r<-gam[seq((2*teachers+1),(3*teachers)),2]
y1<-50+ eta.stu+gam1[z1]+cont_var1+eps1
y2<-eta.stu+gam1[z1]*alpha+gam2[z2]+cont_var2+eps2
y3<-100+eta.stu+gam1[z1]*alpha+gam2[z2]*alpha+gam3[z3]+cont_var3+eps3
r1<-rbinom(students*teachers,1,
           pnorm(.1+eta.stu.r+gam1.r[z1]+gam2.r[z2]+gam3.r[z3]))
student<-1:(students*teachers)
teacher<-c(z1,z2,z3)
cont_var<-c(cont_var1,cont_var2,cont_var3)
year<-c(rep(1:3,each=students*teachers))
y<-c(y1,y2,y3)
vam_data2<-as.data.frame(cbind(student,teacher,year,y,cont_var))
vam_data2<-vam_data2[order(vam_data2$student,vam_data2$year),]
example.score.data<-vam_data2
vam_data2.r<-as.data.frame(cbind(student,r=r1))
example.outcome.data<-vam_data2.r

```

### Examples

```

data(example.score.data)
print(example.score.data[1,])

```

**Description**

Plot teacher effects and residuals. The caterpillar plots use a modified version of the plotCI function from R package gplots. According to that package, "Original version [of plotCI] by Bill Venables wvenable@attunga.stats.adelaide.edu.au posted to r-help on Sep. 20, 1997. Enhanced version posted to r-help by Ben Bolker ben@zoo.ufl.edu on Apr. 16, 2001. This version was modified and extended by Gregory R. Warnes greg@warnes.net. Additional changes suggested by Martin Maechler maechler@stat.math.ethz.ch integrated on July 29, 2004."

**Usage**

```
## S3 method for class 'RealVAMS'  
plot(x, ..., alpha)
```

**Arguments**

x	an object of class RealVAMS
...	other arguments
alpha	the significance level for the caterpillar plots

**Value**

Requires user to click window or press "enter" to progress through plots. Returns caterpillar plots (via the package gplots) and residual plots.

**Author(s)**

Andrew Karl <akar1@asu.edu>, Jennifer Broatch, Jennifer Green, Other authors as listed above for the caterpillar plots.

**See Also**

[summary.RealVAMS](#)

**Examples**

```
data(example.score.data)  
data(example.outcome.data)  
  
res<-RealVAMS(example.score.data,example.outcome.data)  
plot(res)
```

---

print	<i>Print</i>
-------	--------------

---

### Description

Prints names of elements in RealVAMS object.

### Usage

```
## S3 method for class 'RealVAMS'
print(x, ...)
```

### Arguments

x	object of class RealVAMS
...	other arguments to be passed to summary

---

RealVAMS	<i>Multivariate VAM Fitting</i>
----------	---------------------------------

---

### Description

Fits a multivariate value-added model (VAM), see Broatch, Green, and Karl (2018) <doi:10.32614/RJ-2018-033>, and Broatch and Lohr (2012) <doi:10.3102/1076998610396900>, with normally distributed test scores and a binary outcome indicator. A pseudo-likelihood approach, Wolfinger and O'Connell (1993) <doi:10.1080/00949659308811554>, is used for the estimation of this joint generalized linear mixed model. The inner loop of the pseudo-likelihood routine (estimation of a linear mixed model) occurs in the framework of the EM algorithm presented by Karl, Yang, and Lohr (2013) <DOI:10.1016/j.csda.2012.10.004>. This material is based upon work supported by the National Science Foundation under grants DRL-1336027 and DRL-1336265.

### Usage

```
RealVAMS(score.data, outcome.data, persistence = "CP", school.effects = FALSE,
REML = TRUE, score.fixed.effects = formula(~as.factor(year) + 0),
outcome.fixed.effects = formula(~1), max.iter.EM = 10,
outcome.family = binomial(link = "probit"), tol1 = 1e-07, max.PQL.it = 30,
pconv = .Machine$double.eps*1e9, var.parm.hessian = TRUE, verbose = TRUE,
independent.responses = FALSE, cpp.benchmark=FALSE)
```



**Arguments**

<code>score.data</code>	a data frame that contains at least a column "y" containing the student scores, a column "student" containing unique student ID's, a column "teacher" containing the teacher ID's, and a column "year" which contains the year (or semester, etc.) of the time period. The "y" and "year" variables needs to be numeric. If other variables are to be included as fixed effects in the score model, they should also be included in <code>score.data</code> . See 'Note' for further discussion.
<code>outcome.data</code>	a data frame that contains at least a column "r" containing the binary student outcomes (coded 0/1), and a column "student" containing unique student ID's. The student ID's should match those in <code>score.data</code> . If other variables are to be included as fixed effects in the outcome model, they should also be included in <code>outcome.data</code> .
<code>persistence</code>	a character object. Choices are "CP" or "VP", for complete and variable persistence of the teacher score effects, respectively. The teacher outcome effects are modeled with complete persistence, regardless of the selection here.
<code>school.effects</code>	logical. If TRUE, correlated random school-level effects are fitted in the score and outcome response models. For both responses, the school effects are fit with zero-persistence (a student's score in each year is associated with the current school attended, and their outcome is associated with the last school the student attended). The school ID should be included as a column <code>schoolID</code> in the <code>score.data</code> data frame.
<code>REML</code>	logical. If TRUE, the pseudo-response is fit using REML. If FALSE, ML is used.
<code>score.fixed.effects</code>	an object of class formula describing the structure of the fixed effects for the student scores. Categorical variables should be wrapped in an <code>as.factor</code> statement.
<code>outcome.fixed.effects</code>	an object of class formula describing the structure of the fixed effects for the student outcomes. Categorical variables should be wrapped in an <code>as.factor</code> statement.
<code>max.iter.EM</code>	numeric. The maximum number of EM iterations during each pseudo-likelihood iteration
<code>outcome.family</code>	an object of class family describing the assumed distribution of the response. <code>binomial</code> is required, but any link function may be used.
<code>tol1</code>	numeric. Convergence tolerance for EM algorithm during each interior pseudo-likelihood iteration. The convergence criterion is specified under 'Details'.
<code>max.PQL.it</code>	numeric. Maximum number of outer pseudo-likelihood iterations.
<code>pconv</code>	numeric. Convergence criterion for outer pseudo-likelihood iterations. Compare to the <code>PCONV</code> option of SAS PROC GLIMMIX.
<code>var.parm.hessian</code>	logical. If TRUE, the Hessian of the parameters in the error and random effects covariance matrices is calculated, providing standard errors for those parameters. Setting this option to FALSE will reduce the run time of the program: only standard errors for the fixed effects will be returned.
<code>verbose</code>	logical. If TRUE, model information will be printed at each iteration.

`independent.responses`

logical. If TRUE, this option will model the responses independently by fixing the covariances in G at 0 as well as the covariances in the last row/column of R. The resulting estimates are the same as those that would be obtained by modelling the test scores in package GPvam (with REML=FALSE) and modelling the binary responses in SAS GLIMMIX (with link=probit) RealVAMS has been validated against these programs.

`cpp.benchmark`

logical. If TRUE, this option will perform the calculations shown in equation (16) of Karl, Yang, Lohr (2013) using both R and the embedded C++ code to demonstrate the time savings of using C++. A summary table is printed at the end.

## Details

\*The persistence option determines the type of persistence effects that are modeled. The variable persistence model ("VP") assumes that teacher effects in future years are multiples of their effect in the current year (Lockwood et al. 2007). The multipliers in the VP model are called persistence parameters, and are estimated. By contrast, the complete persistence ("CP") model fixes the persistence parameters at 1 and 0 (Lockwood et al. 2007).

\*Convergence is declared for each interior iteration when  $(l_k - l_{k-1})/l_k < \text{tol1}$ , where  $l_k$  is the log-likelihood at iteration  $k$ .

\*The model is linearized using a pseudo-likelihood approach (Wolfinger 1993) and the resulting multiple membership linear mixed model is estimated via an EM algorithm (Karl et al. 2012).

## Value

RealVAMS returns an object of class RealVAMS

<code>loglik</code>	the maximized log-likelihood at convergence of the EM algorithm. Warning: Likelihood-ratio tests are not valid with results from a PQL estimation routine.
<code>teach.effects</code>	a data frame containing the predicted teacher effects and standard errors
<code>school.effects</code>	if <code>school.effects=TRUE</code> , a data frame containing the predicted school effects and standard errors. Otherwise, NULL.
<code>parameters</code>	a matrix of estimated model parameters and standard errors. Wald p-values and 95% confidence intervals are provided for convenience, but these calculations assume infinite degrees of freedom. If working with a "small" data set, re-calculate these using the appropriate degrees of freedom.
<code>Hessian</code>	the Hessian of the variance parameters. <code>var.parm.hessian</code> must be set to TRUE.
<code>R_i</code>	a matrix containing the error covariance matrix of a student's test scores and outcome (assuming a complete response vector with no missing test scores or outcome indicator). The bottom-right component corresponds to the variance of the binary response, and is fixed at 1.
<code>teach.cov</code>	a list containing the unique blocks of the covariance matrix of teacher effects (the G matrix).
<code>mresid</code>	a vector of the raw marginal residuals. Can be reproduced with <code>y.combined-X%%fixed.effects</code> .

<code>cresid</code>	a vector of the raw conditional residuals. Can be reproduced with <code>y.combined-X%%fixed.effects-Z%%eblup[,2]</code> .
<code>y.combined</code>	a vector of the pseudo-responses from the final PQL iteration, with score and outcome responses interleaved (see the notes below). The test scores will be the same as those given as an input, but the 0/1 responses for the binary distribution will be the pseudo-responses. The vector <code>y.response.type</code> indicates which response each component corresponds to. For components corresponding to the binary response, the original response can be obtained from <code>joined.table\$y.combined.original</code> .
<code>y.combined.hat</code>	a vector of the predicted values. This vector can be reconstructed with other variables returned in this object as <code>X%%fixed.effects+Z%%eblup[,2]</code> . The values corresponding to the binary outcome responses can be converted to probabilities (of a 1 response) via the inverse link function <code>outcome.family\$linkinv()</code> . The covariance matrix for this vector of predictions is <code>Z%%G%%t(Z)+R.full</code> .
<code>y.response.type</code>	a vector indicating the type of response in each component of <code>y.combined</code>
<code>y.year</code>	a vector indicating the year in which each component of <code>y.combined</code> was recorded
<code>num.obs</code>	total number of observations (test scores and binary responses)
<code>num.student</code>	total number of students included in the data
<code>num.year</code>	number of years over which test scores were modeled
<code>num.teach</code>	a vector listing the number of teachers in each year
<code>persistence</code>	a character vector indicating the persistence structure (VP or CP) used to model the teacher test-score effects
<code>persistence_parameters</code>	a matrix of the persistence parameters. The (i,j)-th component gives the persistence parameter for year-j teachers on year-i scores.
<code>X</code>	the fixed effects design matrix of the interleaved score and outcome responses
<code>Z</code>	the random effects design matrix of the interleaved score and outcome responses
<code>G</code>	the random effects covariance matrix. This matrix is block diagonal and contains the teacher variance components and, if included, school-level variance components.
<code>C</code>	the joint covariance matrix of the fixed effect estimates and the predicted random effects, that is, for <code>c(fixed.effects,eblup[,1])</code> . See Henderson (1975) for details. This can be used to estimate the variance of an estimable/predictable function.
<code>R</code>	the error covariance matrix, with the variance of the outcome pseudo-responses restricted to 1. See the description for <code>R.full</code> . In most cases, <code>R.full</code> should be used instead of <code>R</code> when analyzing the results of <b>RealVAMS</b> .
<code>R.full</code>	the error covariance matrix, which is formed as the product <code>diag(sqrt.w)%%R%%diag(sqrt.w)</code> . The matrix <code>R</code> assumes a variance of 1 for all of the binomial responses, while <code>R.full</code> includes the variance from the binomial distribution (in Wolfinger (1993), <code>diag(sqrt.w)</code> is called <code>R.mu</code> ).
<code>sqrt.W</code>	vector of weights for the error covariance matrix. See the description for <code>R.full</code> above

<code>eblup</code>	a matrix containing the complete random effects vector and associated standard errors. When <code>school.effects=FALSE</code> , this matrix is identical to the one returned by <code>teach.effects</code> . When <code>school.effects=TRUE</code> , this matrix is equal to <code>rbind(teach.effects, school.effects)</code> . Thus, <code>Z%*%eblup[, 2]</code> may be used to return the subject-specific portion of the predictions.
<code>fixed.effects</code>	a vector containing the fixed effect estimates. This is a subset of parameters, and provided for compatibility with <code>X</code> . That is, <code>X%*%fixed.effects</code> will yield the marginal means.
<code>joined.table</code>	a data frame containing the interleaved score and outcome data sets. See the notes below.
<code>outcome.family</code>	returns information about the distribution and link function used for the outcomes.

### Note

The first few iterations of the EM algorithm will take longer than subsequent iterations. This is a result of the hybrid gradient-ascent/Newton-Raphson method used in the M-step for the R matrix in the first two iterations (Karl et al. 2012).

The model assumes that each teacher teaches only one year. If, for example, a teacher teaches in years 1 and 2, his/her first year performance is modeled independently of the second year performance. To keep these effects separate, the program appends "(year i)" to each teacher name, where *i* is the year in which the teacher taught.

To fit the model and allow correlation between test scores and outcomes (at both the student and teacher levels), the score and outcome response vectors are interleaved into a single response vector. For example, if there are three years of test scores modeled with a binary outcome indicator, the binary indicator for a student is inserted immediately after that student's test scores. The `joined.table` that is returned by RealVAMS shows how this was done for a particular data set. Row *i* of `joined.table` corresponds to row *i* of `X`, `Z`, `R.full`, `y.combined`, `y.combined.hat`, and `eblup`.

The `fixed.effects` arguments of RealVAMS utilizes the functionality of R's `formula` class. In the statement `score.fixed.effects=formula(~as.factor(year)+cont_var+0)`, `as.factor(year)` identifies `year` as a categorical variable. `+0` indicates that no intercept is to be fitted, and `+cont_var` indicates that a separate effect is to be fitted for the continuous variable "cont\_var." An interaction between "year" and "cont\_var" could be specified by `~as.factor(year)*cont_var+0`, or equivalently, `~as.factor(year)+cont_var+as.factor(year):cont_var+0`. See [formula](#) for more details.

### Author(s)

Andrew Karl <akar1@asu.edu>, Jennifer Broatch, Jennifer Green

### References

Broatch, J. and Lohr, S. (2012) <DOI:10.3102/1076998610396900> Multidimensional Assessment of Value Added by Teachers to Real-World Outcomes. *Journal of Educational and Behavioral Statistics* **37**, 256–277.

Broatch, J., Green, J., Karl, A. (2018) <DOI:10.32614/RJ-2018-033> RealVAMS: An R Package for Fitting a Multivariate Value-added Model (VAM). *The R Journal* 10/1, 22–30.

Karl, A., Yang, Y. and Lohr, S. (2013) <DOI:10.1016/j.csda.2012.10.004> Efficient Maximum Likelihood Estimation of Multiple Membership Linear Mixed Models, with an Application to Educational Value-Added Assessments. *Computational Statistics & Data Analysis* 59, 13–27.

Karl, A., Yang, Y. and Lohr, S. (2013) <DOI:10.3102/1076998613494819> A Correlated Random Effects Model for Nonignorable Missing Data in Value-Added Assessment of Teacher Effects. *Journal of Educational and Behavioral Statistics* 38, 577–603.

Karl, A., Yang, Y. and Lohr, S. (2014) <DOI:10.1016/j.csda.2013.11.019> Computation of Maximum Likelihood Estimates for Multiresponse Generalized Linear Mixed Models with Non-nested, Correlated Random Effects. *Computational Statistics & Data Analysis* 73, 146–162.

Henderson, C.R. (1975) Best linear unbiased estimation and prediction under a selection model. *Biometrics* 31(2), 423–447.

Lockwood, J., McCaffrey, D., Mariano, L., Setodji, C. (2007) <DOI:10.3102/1076998606298039> Bayesian Methods for Scalable Multivariate Value-Added Assessment. *Journal of Educational and Behavioral Statistics* 32, 125–150.

Wolfinger, R. (1993) <DOI:10.1080/00949659308811554> Generalized linear mixed models a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* 48 233–243.

## Examples

```
data(example.score.data)
data(example.outcome.data)
#The next line exists to show that the function can run and that the package
#installed correctly
RealVAMS(example.score.data,example.outcome.data,max.PQL.it=1,max.iter.EM=2,
var.parm.hessian=FALSE)

res<-RealVAMS(example.score.data,example.outcome.data)
```

---

REML\_Rm

*Internal function*

---

## Description

An internal function

## Usage

```
REML_Rm(invsqrtW_,Jyp_,loopsize_, patternlength_,rownumber_,ybetas_,etahat_,
tempmatR_,JXpi_,JXpp_,JXpx_,JXpdim_,JZpi_,JZpp_,JZpx_,JZpdim_,betacov_,C12_)
```

**Arguments**

invsqrtW_	an internal variable
JYp_	an internal variable
loopsize_	an internal variable
patternlength_	an internal variable
rownumber_	an internal variable
ybetas_	an internal variable
etahat_	an internal variable
tempmatR_	an internal variable
JXpi_	an internal variable
JXpp_	an internal variable
JXpx_	an internal variable
JXpdim_	an internal variable
JZpi_	an internal variable
JZpp_	an internal variable
JZpx_	an internal variable
JZpdim_	an internal variable
betacov_	an internal variable
C12_	an internal variable

---

R\_mstep2

*Internal function*


---

**Description**

An internal function

**Usage**

```
R_mstep2(invsqrtW_, JYp_, loopsize_, patternlength_, rownumber_, ybetas_, etahat_,
tempmatR_, JXpi_, JXpp_, JXpx_, JXpdim_, JZpi_, JZpp_, JZpx_, JZpdim_)
```

**Arguments**

invsqrtW_	an internal variable
JYp_	an internal variable
loopsize_	an internal variable
patternlength_	an internal variable
rownumber_	an internal variable
ybetas_	an internal variable

etahat_	an internal variable
tempmatR_	an internal variable
JXpi_	an internal variable
JXpp_	an internal variable
JXpx_	an internal variable
JXpdim_	an internal variable
JZpi_	an internal variable
JZpp_	an internal variable
JZpx_	an internal variable
JZpdim_	an internal variable

---

summary

*Summary*


---

## Description

Prints summary information for object of class RealVAMS

## Usage

```
## S3 method for class 'RealVAMS'
summary(object, ...)
```

## Arguments

object            object of class RealVAMS  
 ...              other arguments to be passed to summary

## Author(s)

Andrew Karl <akar1@asu.edu>, Jennifer Broatch, Jennifer Green

## See Also

[plot.RealVAMS](#)

## Examples

```
## Not run:
res<-RealVAMS(example.score.data,example.outcome.data)
summary(res)

## End(Not run)
```

---

vp\_cp

*Internal function*

---

**Description**

An internal function

**Usage**

```
vp_cp(Z_mat, B.mat, control)
```

**Arguments**

Z_mat	data frame
B.mat	data frame
control	a list



# Index

## \* datasets

example.outcome.data, [3](#)

example.score.data, [5](#)

## \* package

RealVAMS-package, [2](#)

## \* regression

plot, [6](#)

RealVAMS, [8](#)

summary, [15](#)

example.outcome.data, [3](#)

example.score.data, [5](#)

formula, [12](#)

plot, [6](#)

plot.RealVAMS, [15](#)

print, [8](#)

R\_mstep2, [14](#)

RealVAMS, [8](#)

RealVAMS-package, [2](#)

REML\_Rm, [13](#)

summary, [15](#)

summary.RealVAMS, [7](#)

vp\_cp, [16](#)