Package: RealVAMS (via r-universe)

October 18, 2026

Type Package

Title Multivariate VAM Fitting

Version 0.5.0

Date 2026-09-18

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 (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.

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

Suggests testthat (>= 3.2.0), roxygen2

Config/testthat/edition 3

RoxygenNote 8.0.0

Encoding UTF-8

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

Maintainer Andrew Karl <akarl@asu.edu>

Repository https://cran.r-universe.dev

Date/Publication 2026-09-18 17:32:04 UTC

RemoteUrl https://github.com/cran/RealVAMS

RemoteRef HEAD

RemoteSha 29a736a3ed6f9920a3672add2482637497642142

Contents

RealVAMS-package 2
example.outcome.data 3
example.score.data 5
plot.RealVAMS 6
print.RealVAMS 8
RealVAMS 8
summary.RealVAMS 15

Index **16**

`RealVAMS-package` *Multivariate VAM Fitting*

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

See `RealVAMS` for model options, return values, and examples.

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.

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

Karl, A. T. (2026). Motivating REML via Prediction-Error Covariances in EM Updates for Linear Mixed Models. arXiv:2602.09247v1. https://arxiv.org/abs/2602.09247v1.

`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.

Usage

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

Format

A data frame with 625 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 following code illustrates the simulation used to generate the data. Exact draws depend on R's random-number generator settings.

```
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 1875 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 following code illustrates the simulation used to generate the data. Exact draws depend on R's random-number generator settings.

```
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,])
```

`plot.RealVAMS` *Plot a RealVAMS fit*

Description

Draw ranked teacher score and outcome effects with pointwise normal intervals, score residual plots by year, and predicted outcome probabilities grouped by the observed binary outcome.

Usage

```
## S3 method for class 'RealVAMS'
plot(x, ..., alpha = 0.1, ask = grDevices::dev.interactive())
```

Arguments

x	An object of class RealVAMS.
...	Named graphical arguments passed to plot and boxplot, such as main, xlab, ylab, or cex. They override the corresponding plot defaults.
alpha	Significance level for the caterpillar plots; a number strictly between zero and one. The default gives 90 percent pointwise intervals.
ask	Logical; ask before starting a new page on an interactive graphics device. The previous setting is restored when the method finishes or fails.

Details

Intervals use the reported prediction standard errors. They are approximate pointwise intervals, not simultaneous intervals for all teachers. Residual plots use the score observations only.

The original caterpillar implementation adapted plotCI from **gplots**, by Bill Venables, Ben Bolker, and Gregory R. Warnes, with contributions by Martin Maechler. The current implementation uses base graphics directly.

Value

The input object, invisibly. Plots are drawn on the current graphics device.

Author(s)

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

See Also

summary.RealVAMS

Examples

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

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

print.RealVAMS *Print a RealVAMS fit or summary*

Description

Print the components and parameter estimates of a fitted object, or the detailed model summary prepared by summary.

Usage

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

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

Arguments

x	An object of class RealVAMS or summary.RealVAMS, as appropriate for the method.
...	Additional arguments passed to the component print methods, for example digits.

Value

The input object, invisibly.

See Also

RealVAMS, summary.RealVAMS

RealVAMS *Multivariate VAM Fitting*

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 * 1e+09,
  var.parm.hessian = TRUE,
  verbose = TRUE,
  independent.responses = FALSE,
  cpp.benchmark = FALSE
)
```

Arguments

`score.data`	A data frame containing numeric scores in `y`, student identifiers in `student`, teacher identifiers in `teacher`, and numeric time periods in `year`. Include any variables used by `score.fixed.effects`. Each student may have at most one row per year. Scores and teacher identifiers may be missing. Each year must have at least two nonconstant observed scores and at least one known teacher among students with observed scores. See 'Note' for missing-data handling and the interpretation of teacher effects.
`outcome.data`	A data frame containing numeric or integer binary outcomes (0 or 1) in `r` and student identifiers in `student`. Each student may have at most one outcome row, and all outcome students must appear in `score.data`. Outcomes may be missing. Both 0 and 1 must be observed among students with recorded scores. Include any variables used by `outcome.fixed.effects`.
`persistence`	A character string, either `"CP"` for complete persistence or `"VP"` for variable persistence of teacher score effects. Teacher outcome effects have complete persistence under either choice.
`school.effects`	Logical. If `TRUE`, fit correlated random school effects for scores and outcomes, using the `schoolID` column of `score.data`. School effects have zero persistence: a score is associated with the current school and the outcome with the student's last supplied school.
`REML`	Logical. If `TRUE`, fit each linearized pseudo-response model by restricted maximum likelihood (REML); otherwise use maximum likelihood (ML). This choice applies to the inner linear mixed model.
`score.fixed.effects`	A one-sided formula for the score fixed effects. Wrap categorical variables in `as.factor()` when they should be treated as factors. For a single score year,

the default formula (including when supplied explicitly) becomes `~1`. Offsets are not supported.

`outcome.fixed.effects`
: A one-sided formula for the outcome fixed effects. Wrap categorical variables in `as.factor()` when they should be treated as factors. Offsets are not supported.

`max.iter.EM`
: Positive integer. Maximum number of EM iterations in each pseudo-likelihood iteration.

`outcome.family`
: A `binomial` family object, including the desired link, such as `binomial(link = "probit")` or `binomial(link = "logit")`.

`tol1`
: Finite nonnegative convergence tolerance for the inner EM algorithm. See 'Details'.

`max.PQL.it`
: Positive integer. Maximum number of outer pseudo-likelihood iterations.

`pconv`
: Finite nonnegative convergence tolerance for the outer pseudo-likelihood iterations. See 'Details'.

`var.parm.hessian`
: Logical. If `TRUE`, calculate the Hessian for the free covariance and persistence parameters to obtain their standard errors. Fixed parameters have no Hessian-based standard errors. Setting this to `FALSE` reduces run time; fixed-effect standard errors and random-effect prediction standard errors are still returned.

`verbose`
: Logical. If `TRUE`, print model information and progress during fitting. Set to `FALSE` for silent fitting.

`independent.responses`
: Logical. If `TRUE`, set all score-outcome covariances in the teacher, school (if fitted), and residual covariance matrices to zero. The score and outcome models then have independent random components. This option does not change the selected persistence, ML/REML method, link function, Hessian, or benchmarking settings.

`cpp.benchmark`
: Logical. If `TRUE`, calculate the residual covariance sufficient statistics in both R and C++, checking agreement and recording timing. Timing is printed when `verbose = TRUE`. This uses the ML or REML formulation selected by `REML`.

Details

The `persistence` option controls persistence of teacher score effects. Under variable persistence (`"VP"`), a teacher's effect in each subsequent year is an estimated multiple of the effect in the current year (Lockwood et al., 2007). Under complete persistence (`"CP"`), current and subsequent effects have multiplier 1, and effects before the teacher's year have multiplier 0.

The model is linearized by the pseudo-likelihood approach of Wolfinger and O'Connell (1993). Each resulting multiple-membership linear mixed model is fitted by an EM algorithm based on Karl, Yang, and Lohr (2013). ML and REML differ in the covariance adjustments used in these updates.

The inner EM stopping criterion compares the absolute likelihood change with the magnitude of the current likelihood, using `tol1`. The outer stopping criterion is $2 \max_j \frac{|\theta_j^{new} - \theta_j^{old}|}{|\theta_j^{new}| + |\theta_j^{old}| + 10^{-6}} < pconv$, where the parameters include the fixed effects, free covariances, and persistence parameters. Reaching an iteration limit does not establish convergence; inspect the returned `convergence` component.

The reported likelihood is for a linearized pseudo-response model. Likelihood-ratio tests are not valid for comparing these PQL fits.

Value

RealVAMS returns an object of class `RealVAMS`, a list containing:

`loglik`	The scalar log-likelihood of the final fitted linearized pseudo-response model. Under `REML = TRUE`, this is a restricted log-likelihood. It is not the likelihood of the original binary data and must not be used for likelihood-ratio tests.
`teach.effects`	a data frame containing the predicted teacher effects and standard errors
`school.effects`	if `school.effects=TRUE`, a data frame containing the predicted school effects and standard errors. Otherwise, NULL.
`parameters`	A matrix of parameter estimates and standard errors. Wald statistics, p-values, and 95% confidence intervals use a normal approximation and are not small-sample corrections. Variance-component p-values and intervals are not provided; fixed parameters have no standard errors.
`Hessian`	The Hessian for covariance and persistence parameters when `var.parm.hessian = TRUE`; otherwise `NA`. Rows and columns for parameters fixed by the model are `NA`.
`R_i`	The within-student residual covariance matrix before the linearization weights are applied, shown for a complete response pattern. Its final row and column correspond to the standardized outcome residual; the bottom-right component is fixed at 1.
`teach.cov`	a list containing the unique blocks of the covariance matrix of teacher effects (the G matrix).
`mresid`	a vector of the raw marginal residuals. Can be reproduced with `y.combined-X%*%fixed.effects`.
`cresid`	a vector of the raw conditional residuals. Can be reproduced with `y.combined-X%*%fixed.effects-Z%*%eblup[,2]`.
`y.combined`	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 y.response.type indicates which response each component corresponds to. For components corresponding to the binary response, the original response can be obtained from `joined.table$y.combined.original`.
`y.combined.hat`	Conditional fitted values on the score or outcome link scale, `X %*% fixed.effects + Z %*% eblup[, 2]`. Apply `outcome.family$linkinv()` to the outcome components to obtain fitted probabilities. The matrix `Z %*% G %*% t(Z) + R.full` is the marginal covariance of the linearized pseudo-response, not the covariance of these fitted values.
`y.response.type`	a vector indicating the type of response in each component of `y.combined`
`y.year`	a vector indicating the year in which each component of `y.combined` was recorded

`num.obs` total number of observations (test scores and binary responses)

`num.student` total number of students included in the data

`num.year` number of years over which test scores were modeled

`num.teach` a vector listing the number of teachers in each year

`persistence` a character vector indicating the persistence structure (VP or CP) used to model the teacher test-score effects

`persistence_parameters`
a matrix of the persistence parameters. The (i,j)-th component gives the persistence parameter for year-j teachers on year-i scores.

`X` the fixed effects design matrix of the interleaved score and outcome responses

`Z` the random effects design matrix of the interleaved score and outcome responses

`G` the random effects covariance matrix. This matrix is block diagonal and contains the teacher variance components and, if included, school-level variance components.

`C` The inverse mixed-model coefficient matrix, giving the joint fixed-effect estimation and random-effect prediction-error covariance for $(\widehat{\beta} - \beta, \widehat{\eta} - \eta)$ at the fitted covariance parameters. Its order is `c(fixed.effects, eblup[, 2])`. See Henderson (1975). It does not include uncertainty in the estimated covariance parameters.

`R` The assembled residual covariance matrix before the linearization weights are applied. Standardized outcome residuals have variance 1. Use `R.full` for the residual covariance on the pseudo-response scale.

`R.full` The covariance matrix of the linearized residuals, `diag(sqrt.W) %*% R %*% diag(sqrt.W)`. Outcome weights incorporate both the binomial variance and the link derivative; score weights are 1.

`sqrt.W` vector of weights for the error covariance matrix. See the description for R.full above

`eblup` A data frame containing the complete random-effects vector, with columns `effect`, `EBLUP`, and `std_error`. Teacher effects precede any school effects. Thus `Z %*% eblup[, 2]` gives the random-effects contribution to fitted values.

`fixed.effects` a vector containing the fixed effect estimates. This is a subset of `parameters`, and provided for compatibility with `X`. That is, `X%*%fixed.effects` will yield the marginal means.

`joined.table` a data frame containing the interleaved score and outcome data sets. See the notes below.

`outcome.family` returns information about the distribution and link function used for the outcomes.

`convergence` A list with logical components `pql` and `em` indicating convergence of the outer pseudo-likelihood loop and final inner EM loop. `pql.iterations` is the number of outer iterations; `em.iterations` gives the number of inner iterations for each outer iteration. `max.parameter.change` is the last maximum relative parameter change, or `NA` before two outer fits have been compared. `em.loglik` is a list of inner Gaussian working-model log-likelihood histories, one vector per

outer iteration. Each vector describes one fixed pseudo-response model; values from different outer iterations correspond to different working models and are not directly comparable. `hessian` reports `"not requested"`, `"available"`, `"not positive definite"`, or `"numerical differentiation failed"` for the optional covariance-parameter uncertainty calculation. Limited-iteration fits are useful for smoke tests but should not be treated as converged estimates.

Note

Missing scores and outcomes are omitted from the response vector. Students without any observed score are excluded. A missing teacher identifier contributes no teacher effect for that student and year. Student identifiers must be nonmissing numeric, character, or factor values and must agree across the two input data frames. Year values must be finite numbers. Internally, years are replaced by chronological indices `1, 2, ...`; model formulas, year labels, and returned persistence indices use this ordering. Covariates used in the fixed-effect formulas must be nonmissing and finite for the corresponding observed responses; unused covariates may contain missing values. When fitting school effects, school identifiers must be supplied and nonmissing for every retained student's score-data row. Invalid inputs produce an error. Each modeled year must contribute teacher effects to both observed scores and observed outcomes; a wholly absent response/year design block is rejected because its variance component is not identifiable.

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. 2013).

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`, and `y.combined.hat`.

The `fixed.effects` arguments of `RealVAMS` use 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 <akarl@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. and O'Connell, M. (1993) <DOI:10.1080/00949659308811554> Generalized linear mixed models a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* **48** 233–243.

Karl, A. T. (2026). Motivating REML via Prediction-Error Covariances in EM Updates for Linear Mixed Models. arXiv:2602.09247v1. https://arxiv.org/abs/2602.09247v1.

Examples

```
data(example.score.data)
data(example.outcome.data)
# A short fit checks installation; it is not a converged analysis.
fit <- RealVAMS(example.score.data, example.outcome.data,
                max.PQL.it = 1, max.iter.EM = 2,
                var.parm.hessian = FALSE, verbose = FALSE)
fit$convergence


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

summary.RealVAMS *Summarize a RealVAMS fit*

Description

Prepare model information for printing, including response counts, covariance matrices, parameter estimates, and score residual summaries. When school effects were fitted, include the school count and shared school score/outcome covariance and correlation matrices. Correlations involving a nonpositive or unavailable variance are shown as NA.

Usage

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

Arguments

object	An object of class RealVAMS.
...	Reserved for additional arguments.

Value

An object of class summary.RealVAMS, containing the fitted object's components and nyear, the number of score years. Printing this object summarizes raw residuals for scores only; binary working residuals are excluded from those summaries.

Author(s)

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

See Also

RealVAMS, plot.RealVAMS

Examples

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

## End(Not run)
```

Index

∗ **datasets**
    example.outcome.data, 3
    example.score.data, 5
∗ **package**
    RealVAMS-package, 2
∗ **regression**
    plot.RealVAMS, 6
    RealVAMS, 8
    summary.RealVAMS, 15

example.outcome.data, 3
example.score.data, 5

formula, *13*

plot.RealVAMS, 6, *15*
print.RealVAMS, 8
print.summary.RealVAMS
        *(*print.RealVAMS*)*, 8

RealVAMS, *2*, 8, *8*, *15*
RealVAMS-package, 2

summary.RealVAMS, *7*, *8*, 15