

Package ‘GPvam’

September 18, 2026

Type Package

Title Maximum Likelihood Estimation of Multiple Membership Mixed Models Used in Value-Added Modeling

Version 3.3-0

Date 2026-09-18

Description An EM algorithm, Karl et al. (2013) <[doi:10.1016/j.csda.2012.10.004](https://doi.org/10.1016/j.csda.2012.10.004)>, estimates generalized, reduced generalized, variable, complete, and zero persistence models by maximum likelihood or restricted maximum likelihood (the default). These are multiple-membership linear mixed models with teachers modeled as ``G-side" effects and students modeled with either ``G-side" or ``R-side" effects; see Mariano et al. (2010) <[doi:10.3102/1076998609346967](https://doi.org/10.3102/1076998609346967)>.

Depends R (>= 3.2.0), Matrix

Imports numDeriv, rlang, Rcpp (>= 0.11.2), graphics, grDevices, methods, stats, utils, ggplot2, patchwork, MASS

LinkingTo Rcpp, RcppArmadillo

ByteCompile yes

NeedsCompilation yes

LazyData yes

License GPL-2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 8.0.0

Encoding UTF-8

Author Andrew Karl [cre, aut] (ORCID: <<https://orcid.org/0000-0002-5933-8706>>),
Yan Yang [aut],
Sharon Lohr [aut]

Maintainer Andrew Karl <akar1@asu.edu>

Repository CRAN

Date/Publication 2026-09-18 17:52:04 UTC

Contents

GPvam-package	2
bias.test.custom	3
GPvam	5
GPvam-methods	9
GPvam.benchmark	10
plot.GPvam	11
print.GPvam	12
summary.GPvam	13
vam_data	14
Index	16

GPvam-package	<i>Maximum Likelihood Estimation of Multiple Membership Mixed Models</i>
---------------	--

Description

An EM algorithm fits multiple-membership linear mixed models used in educational value-added assessment. Teachers are modeled with random effects; dependence among a student’s observations is modeled through either the residual covariance or a random student effect. Restricted maximum likelihood (REML) is used by default; set REML = FALSE in `GPvam` for maximum likelihood (ML).

Details

Use `GPvam` to fit generalized, reduced generalized, variable, complete, or zero persistence models. `summary.GPvam` summarizes a fitted model and `plot.GPvam` draws diagnostic plots. `bias.test.custom` provides a permutation diagnostic for bias induced by dependence between random effects and their model matrix.

The package includes simulated example data in `vam_data` and historical runtime and memory measurements in `GPvam.benchmark`.

Author(s)

Andrew Karl <akar1@asu.edu>, Yan Yang and Sharon Lohr.

References

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

Karl, A., Yang, Y. and Lohr, S. (2013). 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. doi:10.1016/j.csda.2012.10.004.

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

Karl, A., Yang, Y. and Lohr, S. (2013). 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. doi:10.3102/1076998613494819.

Karl, A. and Zimmerman, D. (2021). A diagnostic for bias in linear mixed model estimators induced by dependence between the random effects and the corresponding model matrix. *Journal of Statistical Planning and Inference* **211**, 107–118. doi:10.1016/j.jspi.2020.06.004.

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

Mariano, L., McCaffrey, D. and Lockwood, J. (2010). A Model for Teacher Effects From Longitudinal Data Without Assuming Vertical Scaling. *Journal of Educational and Behavioral Statistics* **35**, 253–279. doi:10.3102/1076998609346967.

McCaffrey, D. and Lockwood, J. (2011). Missing Data in Value-Added Modeling of Teacher Effects. *Annals of Applied Statistics* **5**, 773–797.

bias.test.custom

Assess Potential Fixed-Effect Bias with a Permutation Diagnostic

Description

Use fitted random effects to diagnose potential bias in fixed-effect estimates or their contrasts, following Karl and Zimmerman (2021). The diagnostic addresses dependence between random effects and the corresponding model matrix.

Usage

```
bias.test.custom(
  result,
  k_vectors = NULL,
  n_perms = 1e+05,
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

result	A fitted GPvam object containing X, Z, vinv, eta.hat, G, num. teach, and persistence. G-side student models also require their student count and student-side meta-data.
k_vectors	A numeric contrast vector, or a list of such vectors. Each vector must have one finite coefficient per column of X. If NULL, test each fixed effect separately using the standard basis vectors.
n_perms	Positive integer number of random permutations for each contrast. The default is 1e5; larger values improve Monte Carlo precision at a greater computational cost.

plot	Logical; display the permutation histograms and their combined plot. Histogram objects are returned even when FALSE.
verbose	Logical; print progress and the results table.

Details

For a contrast k , the observed diagnostic is $\nu' \hat{\eta}$, with

$$\nu' = k'(X'V^{-1}X)^{-1}X'V^{-1}Z.$$

Here X and Z are the fixed- and random-effects design matrices, V is the fitted marginal covariance, and $\hat{\eta}$ is the vector of predicted random effects.

Teacher effects are permuted among teachers from the same originating year. For generalized and reduced generalized persistence models, each teacher's whole vector of current and future effects moves together, preserving the within-teacher covariance structure. Random student intercepts in G-side models are permuted separately among students. This block permutation specializes the diagnostic to GPvam's covariance structure; it does not implement the rank-based sampler in Karl and Zimmerman (2021).

The reported two-sided Monte Carlo p-value is $(1 + B)/(1 + n_perms)$, where B counts permutations whose absolute diagnostic is at least as large as the observed value. Including the observed arrangement avoids zero p-values from a finite random sample of permutations (Phipson and Smyth, 2010). Because covariance parameters and random effects are estimated, these are approximate fitted-model diagnostics, not exact finite-sample tests of fixed-effect bias.

Permutations use R's current random-number generator. Call `set.seed()` before the diagnostic to reproduce the results.

Value

A list with:

permutation_results	A data frame with columns <code>Fixed_Effect</code> (effect or contrast name), <code>Nu_Prime_Eta</code> (observed diagnostic), and <code>Permutation_P_Value</code> (two-sided Monte Carlo p-value).
plot_list	A list of <code>ggplot2</code> histogram objects.

References

- Karl, A. and Zimmerman, D. (2021). A diagnostic for bias in linear mixed model estimators induced by dependence between the random effects and the corresponding model matrix. *Journal of Statistical Planning and Inference* **211**, 107–118. doi:[10.1016/j.jspi.2020.06.004](https://doi.org/10.1016/j.jspi.2020.06.004).
- Phipson, B. and Smyth, G. K. (2010). Permutation P-values Should Never Be Zero: Calculating Exact P-values When Permutations Are Randomly Drawn. *Statistical Applications in Genetics and Molecular Biology* **9**(1), Article 39. doi:[10.2202/15446115.1585](https://doi.org/10.2202/15446115.1585).

See Also

[GPvam](#)

Examples

```
data(vam_data)
result <- GPvam(vam_data, persistence = "CP",
               fixed_effects = ~ as.factor(year) + cont_var + 0,
               verbose = FALSE)
set.seed(2026)
bias <- bias.test.custom(result, k_vectors = c(0, 0, 0, 1),
                        n_perms = 999, plot = FALSE, verbose = FALSE)
bias$permutation_results
```

GPvam

Fit a Generalized or Variable Persistence Model

Description

Fit a multiple-membership linear mixed model by an EM algorithm. Teacher effects may persist into future years, and dependence among a student's observations is represented by a residual covariance matrix or a random student effect. See Karl, Yang and Lohr (2013) and Mariano et al. (2010).

Usage

```
GPvam(
  vam_data,
  fixed_effects = formula(~as.factor(year) + 0),
  student.side = "R",
  persistence = "GP",
  max.iter.EM = 1000,
  tol1 = 1e-07,
  hessian = FALSE,
  hes.method = "simple",
  REML = TRUE,
  verbose = TRUE
)
```

Arguments

<code>vam_data</code>	A data frame containing <code>y</code> (numeric student score), <code>student</code> (student identifier), <code>teacher</code> (teacher identifier), and <code>year</code> (numeric year or other ordered time period), together with any fixed-effect covariates. There must be at most one row per student and year. Student identifiers and years must not be missing. Scores and teacher identifiers may be NA; fixed-effect covariates must be complete on rows with observed scores. Unused columns are ignored. See 'Missing data' below.
<code>fixed_effects</code>	A one-sided formula specifying fixed effects. The default estimates a separate mean for each year, or an intercept if there is only one year. Use factors for categorical covariates; for example, <code>~ as.factor(year) + cont_var + 0</code> also

	includes a continuous covariate. Offsets and the reserved variable names <code>mis</code> , <code>pat</code> , and <code>r</code> are not supported in this formula.
<code>student.side</code>	Character string, either "R" for an unstructured within-student residual covariance or "G" for a random student intercept and year-specific residual variances. A G-side model requires repeated observed scores for at least one student.
<code>persistence</code>	Character string selecting "GP", "rGP", "VP", "CP", or "ZP". Only "GP" is supported with <code>student.side = "G"</code> . See 'Persistence models' below.
<code>max.iter.EM</code>	Positive integer giving the maximum number of EM iterations. A fit that reaches this limit without convergence is returned with a warning and <code>converged = FALSE</code> .
<code>tol1</code>	Positive finite convergence tolerance for the relative change in log likelihood. The default is $1e-7$.
<code>hessian</code>	Logical; calculate the observed information for covariance parameters and, for <code>persistence = "VP"</code> , persistence parameters. This supplies their standard errors. Fixed-effect standard errors and teacher prediction-error standard deviations are computed regardless. Covariance/persistence standard errors are NA when the information matrix cannot be computed or is not positive definite.
<code>hes.method</code>	Numerical differentiation method for the information matrix: "simple" or "richardson".
<code>REML</code>	Logical; use restricted maximum likelihood (the default) rather than maximum likelihood. Supported for every available persistence/student-side combination.
<code>verbose</code>	Logical; print model information and iteration progress.

Details

The model is fitted by an EM algorithm based on Karl, Yang and Lohr (2013). Covariance updates use conditional random-effect moments for ML and prediction-error covariance adjustments for REML; see Karl (2026). With incomplete observation patterns, the unstructured residual covariance update uses a Cholesky parameterization to retain positive definiteness. Setting `hessian = TRUE` adds numerical differentiation after fitting and can substantially increase runtime. Historical timings are available in [GPvam.benchmark](#).

Fixed-effect standard errors and teacher prediction-error standard deviations use the inverse mixed-model coefficient matrix with estimated covariance parameters treated as known. They do not include an additional adjustment for uncertainty in those covariance estimates.

Value

An object of class `GPvam`, a list that includes:

<code>loglik</code>	Log likelihood (restricted log likelihood for REML) at the returned fit. Meeting the relative-change convergence criterion does not guarantee a global maximum.
<code>converged</code>	Whether the convergence criterion was met.
<code>iter, loglik_history</code>	Number of evaluated iterations and the corresponding sequence of log likelihoods.
<code>teach.effects</code>	Data frame containing teacher identifiers, originating and effect years, predicted effects (EBLUP), and prediction-error standard deviations (<code>std_error</code>).

<code>parameters</code>	Matrix of fixed-effect, covariance, and applicable persistence estimates with their standard errors. Covariance and persistence standard errors are NA unless <code>hessian = TRUE</code> .
<code>Hessian</code>	Observed information matrix for covariance and persistence parameters when requested, otherwise NA.
<code>R_i</code>	Within-student residual covariance for R-side models; NA for G-side models.
<code>teach.cov</code>	List of teacher covariance blocks, one per originating year.
<code>stu.cov</code>	Student random-effect variance for a G-side model; NA for an R-side model.
<code>mresid, cresid, sresid</code>	Marginal residuals, conditional residuals, and conditional residuals scaled using the residual precision matrix.
<code>y, yhat, yhat.m, yhat.s</code>	Observed responses, conditional fitted values, marginal fitted values, and scaled conditional fitted values, in the internal fitting order.
<code>X, Z, G, R.inv, vinv</code>	Fixed- and random-effects design matrices, random-effects covariance, residual precision, and marginal precision.
<code>ybetas, eta.hat</code>	Unrounded fixed-effect estimates and random-effect predictions.
<code>num.obs, num.student, num.year, num.teach</code>	Observation, student, year, and per-year teacher counts.
<code>persistence, student.side, REML</code>	The fitted model and estimation options.
<code>key</code>	Mapping between original year values and internal year indices.
<code>observation</code>	Input-row indices in the internal fitting order.
<code>call, fixed_effects</code>	Original model call and fixed-effects formula.

Use the standard [GPvam-methods](#) to extract full-precision fixed-effect estimates and covariances, fitted values, residuals, likelihood, and observation count. The fitted-value and residual methods return observations in input order with missing responses omitted.

Persistence models

The generalized persistence model ("GP") assigns each teacher correlated current-year and future-year effects. The reduced model ("rGP") combines a teacher's future-year effects into one shared effect. The variable persistence model ("VP") makes a teacher's future-year effects multiples of the current-year effect and estimates those multipliers. Complete ("CP") and zero ("ZP") persistence fix the multipliers at one and zero, respectively.

With `student.side = "R"`, an unstructured covariance matrix models within-student dependence. With `student.side = "G"`, a random student intercept represents that dependence and the remaining errors are independent, with a separate variance for each year. The latter formulation assumes that scores across years use a common scale and can have substantial memory requirements for large numbers of random effects.

Missing data

A missing score is omitted from the response used for fitting. A known teacher assignment on that row is retained in the student's teaching history and can contribute persistent effects to later observed scores. A missing teacher identifier represents an unknown assignment, not a teacher named "NA"; no teacher effect is attached to that assignment.

Note

Each teacher-year is a separate unit. If the same teacher identifier appears in different years, the corresponding effects are modeled independently. Internally, a year suffix is added to each known teacher identifier, and the original year labels are retained in the output. Before evaluating `fixed_effects`, years are recoded to consecutive numeric indices in increasing order. Thus `~ year` specifies a linear trend across those indices. To use the original numerical spacing, copy the original years into a separate covariate and use that covariate in the formula.

The fixed-effects design must have full column rank. Use `~ as.factor(year) * cont_var + 0` to include year-specific slopes; see [formula](#) for formula syntax.

Author(s)

Andrew Karl <akar1@asu.edu>, Yan Yang and Sharon Lohr.

References

- Karl, A. T. (2026). Motivating REML via Prediction-Error Covariances in EM Updates for Linear Mixed Models. arXiv:2602.09247v1. <https://arxiv.org/abs/2602.09247>.
- Karl, A., Yang, Y. and Lohr, S. (2013). 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. doi:10.1016/j.csda.2012.10.004.
- Karl, A., Yang, Y. and Lohr, S. (2014). Computation of Maximum Likelihood Estimates for Multiresponse Generalized Linear Mixed Models with Non-nested, Correlated Random Effects. *Computational Statistics & Data Analysis* **73**, 146–162.
- Karl, A., Yang, Y. and Lohr, S. (2013). 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. doi:10.3102/1076998613494819.
- Karl, A. and Zimmerman, D. (2021). A diagnostic for bias in linear mixed model estimators induced by dependence between the random effects and the corresponding model matrix. *Journal of Statistical Planning and Inference* **211**, 107–118. doi:10.1016/j.jspi.2020.06.004.
- Lockwood, J., McCaffrey, D., Mariano, L. and Setodji, C. (2007). Bayesian Methods for Scalable Multivariate Value-Added Assessment. *Journal of Educational and Behavioral Statistics* **32**, 125–150.
- Mariano, L., McCaffrey, D. and Lockwood, J. (2010). A Model for Teacher Effects From Longitudinal Data Without Assuming Vertical Scaling. *Journal of Educational and Behavioral Statistics* **35**, 253–279. doi:10.3102/1076998609346967.
- McCaffrey, D. and Lockwood, J. (2011). Missing Data in Value-Added Modeling of Teacher Effects. *Annals of Applied Statistics* **5**, 773–797.

See Also

[plot.GPvam](#), [summary.GPvam](#), [bias.test.custom](#), [vam_data](#)

Examples

```
data(vam_data)

result <- GPvam(vam_data, persistence = "CP",
               fixed_effects = ~ as.factor(year) + cont_var + 0,
               verbose = FALSE)
summary(result)
plot(result, ask = FALSE)
```

GPvam-methods

Extract components of a fitted GPvam model

Description

Standard model methods expose the unrounded estimates used by the fitting algorithm. The displayed parameters table is rounded for presentation.

Usage

```
## S3 method for class 'GPvam'
coef(object, ...)

## S3 method for class 'GPvam'
vcov(object, ...)

## S3 method for class 'GPvam'
fitted(object, type = c("conditional", "marginal"), ...)

## S3 method for class 'GPvam'
residuals(object, type = c("conditional", "marginal", "scaled"), ...)

## S3 method for class 'GPvam'
nobs(object, ...)

## S3 method for class 'GPvam'
logLik(object, ...)
```

Arguments

<code>object</code>	A fitted GPvam object.
<code>...</code>	Further arguments, currently unused.

`type` For fitted, "conditional" includes the predicted random effects and "marginal" uses only the fixed effects. For residuals, these choices subtract the corresponding fitted values; "scaled" returns the conditional residuals transformed by a Cholesky factor of the inverse residual covariance.

Details

`logLik` returns the likelihood criterion actually fitted (ML or REML). It does not refit or convert between likelihood types. The parameter-count convention agrees with common R mixed-model packages. Compare REML likelihoods or information criteria only for models with the same fixed-effect design and the same observations; ML and REML criteria are not directly comparable.

These methods apply to objects fitted by GPvam 3.3-0 or later, which retain full-precision matrices and an input-row map for all model settings.

Value

`coef` returns a named fixed-effect coefficient vector; `vcov` returns its estimated covariance matrix. `fitted` and `residuals` return named numeric vectors in the order of the input rows with observed responses. Rows with missing responses are omitted. `nobs` returns the number of observed responses. `logLik` returns an object of class `logLik`, with degrees of freedom equal to the number of fixed-effect and covariance/persistence parameters.

Examples

```
data(vam_data)
fit <- GPvam(vam_data, persistence = "CP", verbose = FALSE)
coef(fit)
vcov(fit)
head(fitted(fit))
head(residuals(fit, type = "marginal"))
logLik(fit)
nobs(fit)
```

GPvam.benchmark

Historical Runtime and Memory Benchmarks

Description

Runtime and peak memory measurements for different persistence models and simulated data sets. The data sets vary in their numbers of years, teachers per year, and students per teacher. These are historical measurements, not predictions for a current computer or package version.

Usage

```
data(GPvam.benchmark)
```

Format

A data frame with 160 rows and nine variables:

Persistence Persistence model and, where indicated, student side.

Years Number of years.

Students.per.Teacher Number of students per teacher.

Teachers.per.Year Number of teachers per year.

Total.Students Total student-year records, equal to $\text{Years} * \text{Students.per.Teacher} * \text{Teachers.per.Year}$; the original column name is retained.

Total.Teachers Total number of teacher-year effects.

Number.of.Random.Effects.in.Model Number of random effects.

PEAK.MEMORY..MB. Peak memory usage in megabytes.

RUNTIME..min. Runtime in minutes.

Details

All benchmarks used `hessian = TRUE`. Numerical Hessian calculation accounted for 20% to 75% of the recorded runtimes. Use `hessian = FALSE` when standard errors of covariance and persistence parameters are not required. The G-side student model can have large memory requirements as the numbers of students and teachers increase.

Examples

```
data(GPvam.benchmark)
head(GPvam.benchmark)
```

plot.GPvam

Plot Teacher Effects and Residuals from a GPvam Model

Description

Draw caterpillar plots of teacher effects with normal-approximation intervals, normal Q-Q plots of conditional residuals, and residuals against fitted values.

Usage

```
## S3 method for class 'GPvam'
plot(x, ..., alpha = 0.1, ask = interactive())
```

Arguments

x	An object of class GPvam.
...	Additional graphical arguments passed to the plotting functions.
alpha	Significance level for the caterpillar intervals, a finite number strictly between zero and one. The default 0.1 produces 90% intervals.
ask	Logical; ask before advancing to the next plot. Defaults to <code>interactive()</code> . The previous graphics-device setting is restored when the method returns.

Details

Caterpillar intervals use the teacher prediction-error standard deviations returned by GPvam, treating estimated covariance parameters as known. They do not include an adjustment for estimation of the covariance parameters. Separate caterpillar plots are drawn for each available originating-year/effect-year pair using the original year labels.

Earlier versions of GPvam used caterpillar code adapted from plotCI in package **gplots**. Its original version was posted to R-help by Bill Venables on September 20, 1997; an enhanced version was posted by Ben Bolker on April 16, 2001. Gregory R. Warnes modified and extended that version, and changes suggested by Martin Maechler were integrated on July 29, 2004.

Value

The input object `x`, invisibly. Plots are drawn on the active graphics device.

Author(s)

Andrew Karl <akarl@asu.edu>, Yan Yang and Sharon Lohr. Caterpillar plotting contributors are listed in ‘Details’.

See Also

[GPvam](#), [summary.GPvam](#)

Examples

```
data(vam_data)
result <- GPvam(vam_data, persistence = "CP", verbose = FALSE)
plot(result, alpha = 0.05, ask = FALSE)
```

print.GPvam

Print a Fitted GPvam Model

Description

Print the fitting iteration count, likelihood, estimated parameters, and teacher effects. The summary print method also displays covariance matrices, information criteria, and residual summaries.

Usage

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

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

Arguments

`x` An object of class GPvam or, for the summary print method, `summary.GPvam`.
`...` Additional arguments; currently unused.

Details

Teacher effects and covariance matrices are formatted using six significant digits for display. The fitted object retains their full numerical precision, and correlations use the unrounded covariances.

Value

The input object `x`, invisibly.

See Also

[summary.GPvam](#), [GPvam](#)

<code>summary.GPvam</code>	<i>Summarize a Fitted GPvam Model</i>
----------------------------	---------------------------------------

Description

Add information criteria to a fitted model for display by its summary print method.

Usage

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

Arguments

`object` An object of class GPvam.
`...` Additional arguments; currently unused.

Details

AIC is $-2\ell + 2k$, where ℓ is the reported likelihood and k counts estimated fixed effects and covariance/persistence parameters. Restricted-likelihood criteria should only compare models with the same fixed-effects design and response data. Do not compare ML and REML criteria directly.

For ML fits, AICc is the conventional adjustment $AIC + 2k(k+1)/(n-k-1)$, using the number of observed scores for n . It is a heuristic for these correlated-data models, not a model-specific finite-sample correction. It is returned as NA for REML fits or when $n \leq k+1$.

Value

An object of class `summary.GPvam`, containing the fitted object's components plus AIC and AICc.

Author(s)

Andrew Karl <akar1@asu.edu>, Yan Yang and Sharon Lohr.

See Also

[GPvam](#), [plot.GPvam](#)

Examples

```
data(vam_data)
result <- GPvam(vam_data, persistence = "CP", verbose = FALSE)
summary(result)
```

vam_data

Simulated Value-Added Data

Description

A data set simulated from a variable persistence model, illustrating a model with perfectly correlated current and future effects for each teacher.

Usage

```
data(vam_data)
```

Format

A data frame with 3,750 observations on 1,250 students over three years, with 50 teachers in each year, and five variables:

student Numeric student identifier.

teacher Numeric teacher identifier, interpreted within year.

year Numeric observation year, from 1 to 3.

y Numeric student score.

cont_var Numeric continuous covariate.

Details

Scores combine a student effect, current and persistent teacher effects, the continuous covariate, and independent errors. The persistence multiplier is 0.4. The student, teacher, and residual standard deviations are each 5; the covariate standard deviation is 4 and its coefficient is 1.

The original simulation used the following code and R's pre-3.6.0 "Rounding" sampler. On R 3.6.0 or later, first select `RNGkind(sample.kind = "Rounding")` to reproduce the supplied data, and restore your preferred RNG settings afterward. With that sampler, the code reproduces the stored data exactly.

```
set.seed(0)
years <- 3
teachers <- 50
students <- 25
alpha <- .4
eta.stu <- rnorm(students * teachers, 0, 5)
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, 4)
cont_var2 <- rnorm(students * teachers, 0, 4)
cont_var3 <- rnorm(students * teachers, 0, 4)
gam1 <- rnorm(teachers, 0, 5)
gam2 <- rnorm(teachers, 0, 5)
gam3 <- rnorm(teachers, 0, 5)
eps1 <- rnorm(students * teachers, 0, 5)
eps2 <- rnorm(students * teachers, 0, 5)
eps3 <- rnorm(students * teachers, 0, 5)
y1 <- eta.stu + gam1[z1] + cont_var1 + eps1
y2 <- eta.stu + gam1[z1] * alpha + gam2[z2] + cont_var2 + eps2
y3 <- eta.stu + gam1[z1] * alpha + gam2[z2] * alpha + gam3[z3] +
  cont_var3 + eps3
student <- 1:(students * teachers)
teacher <- c(z1, z2, z3)
cont_var <- c(cont_var1, cont_var2, cont_var3)
year <- rep(1:3, each = students * teachers)
y <- c(y1, y2, y3)
vam_data <- as.data.frame(cbind(student, teacher, year, y, cont_var))
```

Examples

```
data(vam_data)
head(vam_data)
```

Index

* datasets

GPvam.benchmark, [10](#)

vam_data, [14](#)

* regression

GPvam, [5](#)

plot.GPvam, [11](#)

summary.GPvam, [13](#)

bias.test.custom, [2](#), [3](#), [9](#)

coef.GPvam (GPvam-methods), [9](#)

fitted.GPvam (GPvam-methods), [9](#)

formula, [8](#)

GPvam, [2](#), [4](#), [5](#), [12–14](#)

GPvam-methods, [9](#)

GPvam-package, [2](#)

GPvam.benchmark, [2](#), [6](#), [10](#)

logLik.GPvam (GPvam-methods), [9](#)

nobs.GPvam (GPvam-methods), [9](#)

plot.GPvam, [2](#), [9](#), [11](#), [14](#)

print.GPvam, [12](#)

print.summary.GPvam (print.GPvam), [12](#)

residuals.GPvam (GPvam-methods), [9](#)

summary.GPvam, [2](#), [9](#), [12](#), [13](#), [13](#)

vam_data, [2](#), [9](#), [14](#)

vcov.GPvam (GPvam-methods), [9](#)