

Package ‘modernVA’

July 23, 2025

Type Package

Title An Implementation of Two Modern Education-Based Value-Added Models

Version 0.1.3

Maintainer Garritt L. Page <page@stat.byu.edu>

Description Provides functions that fit two modern education-based value-added models.

One of these models is the quantile value-added model. This model permits estimating a school's value-added based on specific quantiles of the post-test distribution. Estimating value-added based on quantiles of the post-test distribution provides a more complete picture of an education institution's contribution to learning for students of all abilities. See Page, G.L.; San Martín, E.; Orellana, J.; Gonzalez, J. (2017) <doi:10.1111/rssa.12195> for more details. The second model is a temporally dependent value-added model. This model takes into account the temporal dependence that may exist in school performance between two cohorts in one of two ways. The first is by modeling school random effects with a non-stationary AR(1) process. The second is by modeling school effects based on previous cohort's post-test performance. In addition to more efficiently estimating value-added, this model permits making statements about the persistence of a schools effectiveness. The standard value-added model is also an option.

License GPL

Encoding UTF-8

RoxygenNote 7.1.0

NeedsCompilation yes

Author Garritt L. Page [aut, cre, cph],
S. McKay Curtis [ctb, cph],
Radford M. Neal [ctb, cph]

Repository CRAN

Date/Publication 2023-11-23 20:20:02 UTC

Contents

qVA	2
tdVA	5
Index	8

qVA	<i>R wrapper that accesses C code to fit quantile value-added model</i>
-----	---

Description

qVA is the main function used to fit hierarchical model that produces quantile value-added estimates.

Usage

```
qVA(y,
    xmat,
    school,
    tau=0.5,
    draws=1100, burn=100, thin=1,
    priorVal=c(100^2, 1, 1, 1, 1, 0, 100^2),
    verbose=FALSE)
```

Arguments

y	numeric vector response variable. Must be in long format.
xmat	N x p matrix of covariates (column of 1's must NOT be included) where N is total number of observations.
school	vector indicating to which school each student belongs. School labels must be contiguous and start with 1.
tau	quantile specification. The median is used as a default (tau=0.5)
priorVal	vector of prior distribution parameter values. s2b - prior variance for beta, default is 100^2. a1 - prior shape for lambda, default is 1. b1 - prior rate for lambda, default is 1. a2 - prior shape for sigma2, default is 1. b2 - prior rate for sigma2, default is 1. ma - mean for a, default is 0. s2a - variance for a, default is 100^2
draws	total number of MCMC iterates to be collected. default is 1100
burn	number of total MCMC iterates discarded as burn-in. default is 100
thin	number by which the MCMC chain is thinned. default is 1. Note that the number of MCMC iterates provided is (draws - burn)/thin.
verbose	Logical indicating if MCMC progress and other data summaries should be printed to screen

Value

This function returns a list containing MCMC iterates that correspond to model parameters and school-specific quantile value-added estimates. In order to provide more detail, in what follows let

"T" - be the number of MCMC iterates collected (draws - burn)/thin,

"M" - be the number of schools,

"N" - be the total number of observations.

"p" - be the number of covariates

The output list contains the following

beta - an matrix of dimension (T, p) containing MCMC iterates associated with quantile regression covariate estimates.

alpha - an matrix of dimension (T, M) containing MCMC iterates associated with school-specific random effects.

v - matrix of dimension (T, N) containing MCMC iterates of auxiliary variable.

a - matrix of dimension (T, 1) containing MCMC iterates of mean of the school-specific random effects (alpha).

sig2a - a matrix of dimension (T, 1) containing MCMC iterates of variance of the school-specific random effects (alpha).

lambda - a matrix of dimension (T, M) containing MCMC iterates associated with the lambda parameter of asymmetric laplace distribution.

cVA - a matrix of dimension (T, M) containing MCMC iterates associated with conditional value-added for each school.

mVA - a matrix of dimension (T, M) containing MCMC iterates associated with the marginal value-added for each school.

qVA - a matrix of dimension (T, M) containing MCMC iterates associated with each schools quantile value-added.

Q - a matrix of dimension (T, N) containing MCMC iterates associated with the marginal quantile valued-added regression value for each student (i.e., averaging over school).

References

Page, Garritt L.; San Martín, Ernesto; Orellana, Javiera; Gonzalez, Jorge. (2017) "Exploring Complete School Effectiveness via Quantile Value-Added" Journal of the Royal Statistical Society: Series A 180(1) 315-340

Examples

```
# Example with synthetic data
tau <- 0.75

m <- 4 # number of schools
n <- 25 # number of students
N <- m*n
p <- 1 # number of covariates
betaT <- 0.5
```

```

alphaT <- seq(-10,10, length=m)

# Generate from the asymmetric Laplace
# using a mixture of a Normal and an Exponential
lambdaT <- 0.1;
xi <- rexp(N, 1/lambdaT)
epsilon <- (sqrt((lambdaT*2*xi)/(tau*(1-tau)))*rnorm(N,0,1) +
            (1-2*tau)/(tau*(1-tau))*xi)
epsilon <- rnorm(N,0,1)
alphavec <- rep(alphaT, each=n)

x <- rnorm(N,250,1)
y <- x*betaT + alphavec + epsilon

X <- cbind(x)

school <- rep(1:m, each=n)

fitQ3 <- qVA(y=y, xmat=X, school=school, tau=0.75, verbose=FALSE)

# quantile value-added estimates with 95% credible intervals for each school
qVA.est <- apply(fitQ3$qVA,2,mean)
qVA.int <- apply(fitQ3$qVA,2,function(x) quantile(x, c(0.025, 0.975)))

beta <- fitQ3$beta
alpha <- fitQ3$alpha

mVA <- fitQ3$mVA
cVA <- fitQ3$cVA
Q <- fitQ3$Q

# Plot results.
plot(x,y, col=rep(c("red","blue","green","orange"), each=n), pch=19)

# Plot Q3 quantile regression line for each school
lines(X[school==1,],
      (X[school==1,]*mean(beta) + apply(alpha,2,mean)[1], col='red', lwd=3)
lines(X[school==2,],
      (X[school==2,]*mean(beta) + apply(alpha,2,mean)[2], col='blue', lwd=3)
lines(X[school==3,],
      (X[school==3,]*mean(beta) + apply(alpha,2,mean)[3], col='green', lwd=3)
lines(X[school==4,],
      (X[school==4,]*mean(beta) + apply(alpha,2,mean)[4], col='orange', lwd=3)

# Plot the marginal VA for each school
points(tapply(X, school,mean), apply(mVA,2,mean),
       col=c("red","blue","green","orange"), pch=4, cex=2, lwd=2)

# Plot the conditional VA for each school
points(tapply(X, school,mean), apply(cVA,2,mean),

```

```
col=c("red","blue","green","orange"),pch=10,cex=2, lwd=2)

# Plot the "global" Q3 quantile regression line.
points(X, apply(Q,2,mean), type='l', lwd=2)
```

tdVA	<i>R wrapper that accesses C code to fit temporal dependent value-added models for two cohorts</i>
------	--

Description

tdVA is the main function used to fit the temporally dependent value-added model for two cohorts

Usage

```
tdVA(y1, xmat1,
     y2, xmat2,
     school1, school2,
     groupID=NULL,
     model=0,
     priors=c(0, 100^2, 1, 1, 1, 1, 0, 100^2, -1, 1, 0, 100^2, 0, 100^2),
     var.global=TRUE,
     MHsd=c(0.2),
     nchains=1,
     draws=50000, burn=40000, thin=10,
     verbose=FALSE)
```

Arguments

y1	numeric vector (response variable) of length N1 for cohort 1. Must be in long format.
y2	numeric vector (response variable) of Length N2 for cohort 2. Must be in long format.
xmat1	N1 x p matrix of covariates for cohort 1 (column of 1's must NOT be included).
xmat2	N2 x p matrix of covariates for cohort 2 (column of 1's must NOT be included).
school1	numeric vector indicating to which school each student belongs for cohort 1. These labels must be contiguous labels and start with 1
school2	numeric vector indicating to which school each student belongs for cohort 2. These labels must be contiguous labels and start with 1
groupID	Optional vector that identifies to which group a school belongs. If NULL there is no grouping

<code>model</code>	Integer indicating which value-added model is to be fit 0 - Independent school effects between the two cohorts. 1 - Temporally dependent school effects between two cohorts based on a non-stationary AR(1) process, 2 - Temporally dependent school effects based on previous cohorts post-test performance. 3 - Full model that includes both an AR(1) type correlation and one based on previous cohorts post-test performance.
<code>priors</code>	Vector of prior distribution parameter values. mb - prior mean for beta1 and beta2, default is 0. s2b - prior variance for beta1 and beta2, default is 100^2. at - prior shape for tau22 and tau21, default is 1. bt - prior rate for tau22 and tau21, default is 1. as - prior shape for sigma2, default is 1. bs - prior rate for sigma2, default is 1. mg - prior mean for gamma2, default is 0. (only used if model = 2) s2g - prior variance for gamma2, default is 100^2. (only used if model = 2) lp12 - prior lower bound for phi12, default is -1. (only used if model = 1) up12 - prior upper bound for phi12, default is 1. (only used if model = 1) mp02 - prior mean for phi02, default is 0. s202 - prior variance for phi02, default is 100^2. mp01 - prior mean for phi01, default is 0. s201 - prior variance for phi01, default is 100^2.
<code>var.global</code>	Logical argument. If true, then a model with common sigma21 and sigma22 among schools is fit. If false, then a model with school-specific sigma21i and sigma22i is fit.
<code>MHsd</code>	Tuning parameter associated with M-H step of phi12. Default is 0.2
<code>nchains</code>	number of MCMC chains to run. Default is 1
<code>draws</code>	number of MCMC iterates to be collected. default is 50,000
<code>burn</code>	number of MCMC iterates discarded as burn-in. default is 40,000
<code>thin</code>	number by which the MCMC chain is thinned. default is 10
<code>verbose</code>	Logical indicating if progress of MCMC algorithm should be printed to screen along with other data summaries

Value

This function returns a list that contains MCMC iterates for all the model parameters in addition to the value-added estimates and intervals for each of the two cohorts

Examples

```
# Generate data from model 1 of San Martin et al.
m <- 25 # number of schools
ni <- 20 # number of students per school
```

```
N <- m*ni

# specify parameter values to generate data
beta1 <- 0.6
beta2 <- 0.75;
sig21 <- 100;
sig22 <- 100;
tau2 <- 100
phi02 <- 0;
phi12 <- 0.75;
phi01 <- 0

X1 <- rnorm(N, 0, sqrt(200))
X2 <- rnorm(N, 0, sqrt(200))

alpha1 <- rnorm(m, phi01, sqrt(tau2))
alpha2 <- rnorm(m, phi02 + phi12*alpha1, sqrt(tau2*(1-phi12^2)))

Y1 <- rep(alpha1, each=ni) + X1*beta1 + rnorm(N, 0, sqrt(sig21))
Y2 <- rep(alpha2, each=ni) + X2*beta2 + rnorm(N, 0, sqrt(sig22))

# Create school vector indicating to which school each observation belongs
school1 <- rep(1:m, each=ni)
school2 <- rep(1:m, each=ni)

# design matrix only one covariate and no intercept
X1i <- cbind(X1)
X2i <- cbind(X2)

fit <- tdVA(y1=Y1,xmat1=X1i,y2=Y2,xmat2=X2i,
            school1=school1,school2=school2, groupID=NULL, model=2,
            var.global=TRUE, nchains=1)

# Value-added estimates of cohort 1 and 2 with 95% credible intervals. See paper for details
cbind(fit$VA1.estimate,t(fit$VA1.intervals))
cbind(fit$VA2.estimate,t(fit$VA2.intervals))
```

Index

qVA, [2](#)

tdVA, [5](#)