Package ‘REndo’

November 8, 2017

Type Package
Title Fitting Linear Models with Endogenous Regressors using Latent Instrumental Variables
Version 1.3
Date 2017-11-08
Author Raluca Gui, Markus Meierer, Rene Algesheimer
Maintainer Raluca Gui <raluca.gui@business.uzh.ch>
This version:
- includes an omitted variable test in the multilevel estimation. It is reported in the summary() function of the multilevelIV() function.
- resolves the error "Error in listIDs[, 1] : incorrect number of dimensions" when using the multilevelIV() function.
- a new simulated dataset is provided, dataMultilevelIV, on which to exemplify the multilevelIV() function.
Imports optimx, mvtnorm, AER, e1071, stats, corpcor, lme4, gmm, lmtest, plyr, sandwich, compiler, data.table, Matrix
Depends methods
License GPL-3
RoxygenNote 6.0.1


**R topics documented:**

- `boots` .................................................. 2
- `copulaCorrection` ...................................... 3
- `copulaPStar` ........................................... 5
- `dataCopC1` ............................................. 6
- `dataCopC2` ............................................. 6
- `dataCopDis` ........................................... 7
- `dataHetIV` ............................................. 8
- `dataHigherMoments` ................................... 8
- `dataLatentIV` ......................................... 9
- `dataMultilevelIV` ..................................... 9
- `hetErrorsIV` .......................................... 10
- `higherMomentsIV` ...................................... 12
- `internalIV` ........................................... 14
- `latentIV` ............................................... 15
- `multilevelIV` .......................................... 17

**Description**

Performs bootstrapping to obtain the standard errors of the estimates of the model with one continuous endogenous regressor estimated via maximum likelihood using the `copulaCorrection` function.

**Usage**

```r
boots(bot, formula, endoVar, param, intercept = NULL, data)
```

**Arguments**

- `bot` number of bootstrap replicates.
- `formula` the model formula, e.g. `y ~ X1 + X2 + P`.
- `endoVar` a string with the name of the endogenous variable/s, in quotation marks.
- `param` initial values for the parameters to be optimized over. See `copulaCorrection` for more details.
intercept an optional parameter. The model is estimated by default with intercept. If no intercept is desired in model estimation, intercept should be given the value "FALSE", otherwise the value "TRUE".

data a data frame or matrix containing the variables of the model.

Details

The function could be used only when there is a single endogenous regressor and method one is selected in copulaCorrection. of the copulaCorrection function is used for estimation.

Value

Returns the standard errors of the estimates of the model using the copula method 1 described in Park and Gupta (2012). See Details section of copulaCorrection.

See Also

copulaCorrection

copulaCorrection  Fitting Linear Models Endogeneous Regressors using Gaussian Copula

description

Fits linear models with continuous or discrete endogeneous regressors using Gaussian copulas, method presented in Park and Gupta (2012). This is a statistical technique to address the endogeneity problem, where no external instrumental variables are needed. The important assumption of the model is that the endogeneous variables should NOT be normally distributed.

Usage

copulaCorrection(formula, endoVar, param, type, method, intercept, data)

Arguments

formula the model formula, e.g. $y \sim X_1 + X_2 + X_3$.
endoVar a string with the name/s of the endogenous variables.
param the vector of initial values for the parameters of the model to be supplied to the optimization algorithm. The parameters to be estimated are $\theta = \{b, a, \rho, \sigma\}$, where $b$ are the parameters of the exogenous variables, $a$ is the parameter of the endogenous variable, $\rho$ is the parameter for the correlation between the error and the endogenous regressor, while $\sigma$ is the standard deviation of the structural error.
type the type of the endogenous regressor/s. It can take two values, "continuous" or "discrete".
method

the method used for estimating the model. It can take two values, "1" or "2",
where "1" is the ML approach described in Park and Gupta (2012), and "2" is
the equivalent OLS approach described in the same paper. "1" can be applied
when there is just a single, continuous endogenous variable. With one discrete or
more than one continuous endogenous regressors, the second method is applied
by default.

intercept

optional parameter. The model is estimated by default with intercept. If no
intercept is desired in the model estimation, intercept should be given the value
"FALSE".

data

data frame or matrix containing the variables of the model.

Details

The maximum likelihood estimation is performed by the "BFGS" algorithm. When there are two
endogenous regressors, there is no need for initial parameters since the method applied is by default
the augmented OLS, which can be specified by using method two - "method="2"".

Value

Depending on the method and the type of the variables, it returns the optimal values of the pa-
rameters and their standard errors. When the method one is used, the standard errors returned are
obtained bootstrapping over 10 samples. If more bootstrapping samples are desired, the standard
errors can be obtained using the boots function from the same package. The following are being
returned and can be saved:

coefficients  the estimated coefficients.
standard errors  the corresponding estimated coefficients standard errors.
fitted.values  the fitted values.
residuals  the estimated residuals.
logLik  the estimated log likelihood value in the case of method 1.
AIC  Akaike Information Criterion in the case of method 1.
BIC  Bayesian Information Criterion in the case of method 1.

Author(s)

The implementation of the model by Raluca Gui based on the paper of Park and Gupta (2012).

References

Park, S. and Gupta, S., (2012), 'Handling Endogeneous Regressors by Joint Estimation Using Cop-
ulas’, Marketing Science, 31(4), 567-86.

See Also

higherMomentsIV
Examples

```r
# load dataset dataCopC1, where P is endogenous, continuous and not normally distributed
data(dataCopC1)
## Not run:
c1 <- copulaCorrection(formula = y ~ X1 + X2 + P, endoVar = "P", type = "continuous", method = "1", intercept=TRUE, data = dataCopC1)
summary(c1)

## End(Not run)

# an alternative model can be obtained using "method ="2"".
c12 <- copulaCorrection(formula = y ~ X1 + X2 + P, endoVar = "P", type = "continuous", method = "2", intercept=TRUE, data = dataCopC1)
summary(c12)

# with 2 endogenous regressors no initial parameters are needed, the default is the augmented OLS.
data(dataCopC2)
c2 <- copulaCorrection(formula = y ~ X1 + X2 + P1 + P2, endoVar = c("P1","P2"), type = "continuous", method = "2", intercept=TRUE, data = dataCopC2)
summary(c2)

# load dataset with 1 discrete endogeneous variable.
# having more than 1 discrete endogenous regressor is also possible
data(dataCopDis)
c3 <- copulaCorrection(formula = y ~ X1 + X2 + P, endoVar = "P", type = "discrete", intercept=TRUE, data = dataCopDis)
summary(c3)
```

---

**copulaPStar**

*Inverse-Normal Distribution of the Empirical Distribution Function*

### Description

Computes the empirical distribution function of a variable and the inverse-normal distribution of ECDF.

### Usage

```r
copulaPStar(P)
```

### Arguments

- **P**
  - the variable for which the inverse-normal distribution of its empirical distribution function is needed.

### Value

Returns the inverse-normal distribution of the empirical distribution function of variable P.
See Also
copulaCorrection

dataCopC1  Simulated Dataset

Description
A dataset with two exogenous regressors, \(X_1, X_2\), and one endogenous, continuous regressor, \(P\), having a T-distribution with 3 degrees of freedom. An intercept and a dependent variable, \(y\), are also included. The true parameter values for the coefficients are: \(b_0 = 2, \ b_1 = 1.5, \ b_2 = -3\) and the coefficient of the endogenous regressor is set to \(a_1 = -1\).

Usage
data("dataCopC1")

Format
A data frame with 2500 observations on the following 5 variables.
- \(y\) a numeric vector representing the dependent variable.
- \(I\) a numeric vector representing the intercept.
- \(X_1\) a numeric vector, normally distributed and exogenous.
- \(X_2\) a numeric vector, normally distributed and exogenous.
- \(P\) a numeric vector, continuous and endogenous having T-distribution with 3 degrees of freedom.

dataCopC2  Simulated Dataset

Description
A dataset with two exogenous, normally distributed regressors, \(X_1\) and \(X_2\), two endogenous, continuous regressors, \(P_1\) and \(P_2\), having a T-distribution with 3 and 5 degrees of freedom respectively, with a correlation of 0.25. The correlation between \(P_1\) and the error was set at 0.33, while between \(P_2\) and the error, at 0.15. The dataset contains an intercept and the dependent variable, \(y\). The true parameter value for the model: \(y = b_0 + b_1 \cdot X_1 + b_2 \cdot X_2 + a_1 \cdot P_1 + a_2 \cdot P_2 + \text{eps}\), are: \(b_0 = 2, \ b_1 = 1.5, \ b_2 = -3, \ a_1 = -1, \ a_2 = 0.8\).

Usage
data("dataCopC2")
**dataCopDis**

**Format**

A data frame with 2500 observations on the following 6 variables.

- **y** a numeric vector representing the dependent variable.
- **I** a numeric vector representing the intercept.
- **X1** a numeric vector, normally distributed and exogenous.
- **X2** a numeric vector, normally distributed and exogenous.
- **P1** a numeric vector, continuous and endogenous having T-distribution with 3 degrees of freedom.
- **P2** a numeric vector, continuous and endogenous having T-distribution with 5 degrees of freedom.

**dataCopDis**

**Simulated Dataset**

**Description**

A dataset with an intercept, two exogenous regressors and one endogenous, discrete variable, used for exemplifying the use of `copulaCorrection` function. The true parameter values are: \( b_0 = 2, b_1 = 1.5, b_2 = -3 \), and the coefficient of the endogenous variable is set to \( a_1 = -1 \). The correlation between the endogenous regressor \( P \) and the error term is 0.33. \( P \) has a Poisson distribution with \( \lambda = 5 \).

**Usage**

```r
data("dataCopDis")
```

**Format**

A data frame with 2500 observations on the following 5 variables.

- **y** a numeric vector representing the dependent variable.
- **I** a numeric vector representing the intercept.
- **X1** a numeric vector, normally distributed and exogenous.
- **X2** a numeric vector, normally distributed and exogenous.
- **P** a numeric vector, discrete and endogenous.
### Simulated Dataset

**Description**

A dataset simulated to exemplify the use of the `hetErrorsIV()` function.

**Usage**

```r
data("dataHetIV")
```

**Format**

A data frame with 2500 observations on the following 4 variables.

- `y`: a numeric vector representing the dependent variable.
- `X1`: a numeric vector representing a normally distributed exogenous variable.
- `X2`: a numeric vector representing a normally distributed exogenous variable.
- `P`: a numeric vector representing a normally distributed endogenous variable.

### Simulated Dataset

**Description**

A dataset enclosing a dependent variable, `y`, two exogenous regressors, `X1` and `X2` and one endogenous variable, `P`. The endogenous regressor has to have a non-normal distribution for identification. The model is:

\[
y = b0 + b1 \times X1 + b2 \times X2 + a1 \times P + \text{epsilon}
\]

True parameter values are `b0 = 2`, `b1 = 1.5`, `b2 = -3`, `a1 = -1`.

**Usage**

```r
data("dataHigherMoments")
```

**Format**

A data frame with 2500 observations on the following 4 variables.

- `y`: a numeric vector representing the dependent variable.
- `X1`: a numeric vector, normally distributed and exogenous.
- `X2`: a numeric vector, normally distributed and exogenous.
- `P`: a numeric vector, representing an endogenous regressor.

See Also

[higherMomentsIV](#)
**dataLatentIV**

**Simulated Dataset**

**Description**

A dataset with one endogenous, discrete regressor used for exemplifying the use of the Latent Instrumental Variable function `latentIV`.

**Usage**

```r
data("dataLatentIV")
```

**Format**

A data frame with 2500 observations on the following 3 variables.

- `y` a numeric vector representing the dependent variable.
- `p` a numeric vector representing a discrete and endogenous regressor.
- `z` a numeric vector representing the discrete, latent IV used to build `p`.

**Details**

The dataset was modeled according to the following equations:

\[
P = g0 \star Z + nu
\]

\[
y = b0 + a1 \star P + epsilon
\]

where \( g0 = 2, b0 = 3 \) and \( a1 = -1 \).

**See Also**

`internalIV`

---

**dataMultilevelIV**

**Simulated Dataset**

**Description**

A dataset simulated to exemplify the use of the `multilevelIV()` function.

**Usage**

```r
data("dataMultilevelIV")
```
hetErrorsIV

Fitting Linear Models with Endogenous Regressors using Heteroskedastic Covariance Restrictions

Description
This function estimates the model parameters and associated standard errors for a linear regression model with one or more endogenous regressors. Identification is achieved through heteroscedastic covariance restrictions within the triangular system as proposed in Lewbel(2012). The function hetErrorsIV builds on the lewbel function form the ivlewbel package. Changes have been made only to the printing and the summary of the function, as well as the name.

Usage
hetErrorsIV(formula, data, clustervar = NULL, robust = TRUE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>formula</td>
<td>the model formula, e.g. y ~ X1 + X2 + P.</td>
</tr>
<tr>
<td>data</td>
<td>the data frame containing the dataset. This argument is mandatory.</td>
</tr>
<tr>
<td>clustervar</td>
<td>a character value naming the cluster on which to adjust the standard errors and test statistics.</td>
</tr>
<tr>
<td>robust</td>
<td>if TRUE the function reports standard errors and test statistics that have been corrected for the presence heteroscedasticity using White’s method.</td>
</tr>
</tbody>
</table>
Details
The formula follows a four-part specification. The following formula is an example:
\[ y_2 \sim y_1 | x_1 + x_2 + x_3 | x_1 + x_2 | z_1 \]. Here, \( y_2 \) is the dependent variable and \( y_1 \) is the endogenous regressor. The code \( x_1 + x_2 + x_3 \) represents the exogenous regressors whereas the third part \( x_1 + x_2 \) specifies the exogenous heteroscedastic variables from which the instruments are derived. The final part \( z_1 \) is optional, allowing the user to include traditional instrumental variables. If both robust=TRUE and clustervar=TRUE, the function overrides the robust command and computes clustered standard errors and test statistics adjusted to account for clustering. The function also computes partial F-statistics that indicate potentially weak identification. In cases where there is more than one endogenous regressor the Angrist-Pischke (2009) method for multivariate first-stage F-statistics is employed.

Value
Returns an object of class `hetREndo`, with the following components:

- `coefficients`: a coefficient matrix with columns containing the estimates, associated standard errors, test statistics and p-values.
- `call`: the matched call.
- `obs`: the number of observations.
- `jtest`: J-test for overidentifying restrictions.
- `ftest`: Partial F-test statistics for weak IV detection.

Author(s)
The implementation of the model formula by based on the paper of Lewbel (2012).

References


See Also
`lewbel`, `higherMomentsIV`, `latentIV`, `copulaCorrection`, `ivreg`.

Examples
```r
data(dataHetIV)
resultsHetIV <- hetErrorsIV(y ~ P | X1 + X2 | X1 + X2, data = dataHetIV)
summary(resultsHetIV)
```
higherMomentsIV

**Fitting Linear Models with Endogenous Regressors using Lewbel’s Higher Moments Approach**

**Description**
Fits linear models with one endogenous regressor using internal instruments built using the approach described in Lewbel A. (1997). This is a statistical technique to address the endogeneity problem where no external instrumental variables are needed. The implementation allows the incorporation of external instruments if available. An important assumption for identification is that the endogenous variable has a skewed distribution.

**Usage**
```
higherMomentsIV(formula, endoVar, G = NULL, IIV = c("g", "gp", "gy", "yp", "p2", "y2"), EIV = NULL, data)
```

**Arguments**
- `formula` - the model formula, e.g. `y ~ x1 + x2 + P`.
- `endoVar` - a string with the name/s of the endogenous variable/s.
- `G` - the functional form of G. It can take four values, `x2`, `x3`, ln(x) or `1/x`. The last two forms are conditional on the values of the exogenous variables: greater than 1 or different from 0 respectively.
- `IIV` - stands for "internal instrumental variable". It can take six values: `g`, `gp`, `gy`, `yp`, `p2` or `y2`. Tells the function which internal instruments to be constructed from the data. See "Details" for further explanations.
- `EIV` - stands for "external instrumental variable". It is an optional argument that lets the user specify any external variable(s) to be used as instrument(s).
- `data` - a matrix or data frame containing the variables of the model.

**Details**
Consider the model below:

\[
Y_t = \beta_0 + \gamma'X_t + \alpha P_t + \epsilon_t \quad (1)
\]

\[
P_t = Z_t + \nu_t \quad (2)
\]

The observed data consist of \(Y_t, X_t\) and \(P_t\), while \(Z_t, \epsilon_t,\) and \(\nu_t\) are unobserved. The endogeneity problem arises from the correlation of \(P_t\) with the structural error, \(\epsilon_t\), since \(E(\epsilon\nu) \neq 0\). The requirement for the structural and measurement error is to have mean zero, but no restriction is imposed on their distribution.

Let \(\bar{S}\) be the sample mean of a variable \(S_t\) and \(G_t = G(X_t)\) for any given function \(G\) that has finite third own and cross moments. Lewbel(1997) proves that the following instruments can be constructed and used with 2SLS to obtain consistent estimates:

\[
q_{1t} = (G_t - \bar{G}) \quad (3a)
\]
$q_{2t} = (G_t - \bar{G})(P_t - \bar{P})$ (3b)
$q_{3t} = (G_t - \bar{G})(Y_t - \bar{Y})$ (3c)
$q_{4t} = (Y_t - \bar{Y})(P_t - \bar{P})$ (3d)
$q_{5t} = (P_t - \bar{P})^2$ (3e)
$q_{6t} = (Y_t - \bar{Y})^2$ (3f)

Instruments in equations 3e and 3f can be used only when the measurement and the structural errors are symmetrically distributed. Otherwise, the use of the instruments does not require any distributional assumptions for the errors. Given that the regressors $G(X) = X$ are included as instruments, $G(X)$ should not be linear in $X$ in equation 3a.

Let small letter denote deviation from the sample mean: $s_i = S_i - \bar{S}$. Then, using as instruments the variables presented in equations 3 together with 1 and $X_t$, the two-stage-least-squares estimation will provide consistent estimates for the parameters in equation 1 under the assumptions exposed in Lewbel(1997).

**Value**

Returns an object of class `ivreg`, with the following components:

- **coefficients**: parameters estimates.
- **residuals**: a vector of residuals.
- **fitted.values**: a vector of predicted means.
- **n**: number of observations.
- **df.residual**: residual degrees of freedom for the fitted model.
- **cov.unscaled**: unscaled covariance matrix for coefficients.
- **sigma**: residual standard error.
- **call**: the original function call.
- **formula**: the model formula.
- **terms**: a list with elements "regressors" and "instruments" containing the terms objects for the respective components.
- **levels**: levels of the categorical regressors.
- **contrasts**: the contrasts used for categorical regressors.
- **x**: a list with elements "regressors", "instruments", "projected", containing the model matrices from the respective components. "projected" is the matrix of regressors projected on the image of the instruments.

**Author(s)**

The implementation of the model formula by Raluca Gui based on the paper of Lewbel (1997).

**References**

**internalIV**  
*Constructs Internal Instrumental Variables From Data*

**Description**

The function can be used to construct additional instruments to be supplied to `higherMomentsIV` as additional instruments in the "EIV" argument.

**Usage**

```r
internalIV(formula, endoVar, G = NULL, IIV = c("g", "gp", "gy", "yp", "p2", "y2"), data)
```

**Arguments**

- `formula`  
  the model formula, e.g. `y ~ X1 + X2 + P`.

- `endoVar`  
  a string with the name/s of the endogenous variable/s.

- `G`  
  the functional form of G. It can take four values, `x2`, `x3`, `lnx` or `1/x`. The last two forms are conditional on the values of the exogenous variables: greater than 0 or different from 0 respectively.

- `IIV`  
  the internal instrumental variable to be constructed. It can take six values, `"g", "gp", "gy", "yp", "p2" or "y2"`. See the "Details" section of `higherMomentsIV` for a description of the internal instruments.

- `data`  
  a matrix or data frame containing the variables in the model.

**Examples**

```r
data(dataHigherMoments)

# call higherMomentsIV with internal instrument yp = (Y - mean(Y))(P - mean(P))
h <- higherMomentsIV(formula = y ~ X1 + X2 + P, endoVar = "P", G = "x2", IIV = "yp",
data = dataHigherMoments)

# build an additional instrument p2 = (P - mean(P))^2 using the internalIV() function
eiv <- internalIV(formula = y ~ X1 + X2 + P, endoVar = "P", G="x2", IIV = "p2",
data = dataHigherMoments)

# use the additional variable as external instrument in higherMomentsIV()
h1 <- higherMomentsIV(formula = y ~ X1 + X2 + P, endoVar = "P", G = "x2", IIV = "yp", EIV=eiv,
data = dataHigherMoments)

summary(h1)

# get the robust standard errors using robust.se() function from package ivpack
# library(ivpack)
# sder <- robust.se(h1)
```

**See Also**

- `internalIV`, `ivreg`, `latentIV`
**latentIV**

**Value**

Returns a vector/matrix constructed from the data which can be used as instrumental variable either in `higherMomentsIV` or in any other function/algorithm making use of instruments.

**References**


**See Also**

`higherMomentsIV`

**Examples**

```r
data(dataHigherMoments)
# build an instrument gp = (G - mean(G))(P - mean(P)) using the internalIV() function
# with G = "x3" meaning G(X) = X^3
eiv <- internalIV(formula = y ~ X1 + X2 + P, endoVar = "P", G = "x3", IIV = "gp",
                  data = dataHigherMoments)
```

---

**latentIV**

*Fitting Linear Models with one Endogenous Regressor using Latent Instrumental Variables*

**Description**

Fits linear models with one endogenous regressor and no additional explanatory variables using the latent instrumental variable approach presented in Ebbes,P., Wedel,M., B"ockenholt, U., and Steerneman, A. G. M. (2005). This is a statistical technique to address the endogeneity problem where no external instrumental variables are needed. The important assumption of the model is that the latent variables are discrete with at least two groups with different means and the structural error is normally distributed.

**Usage**

`latentIV(formula, param = NULL, data)`

**Arguments**

- `formula` an object of type 'formula': a symbolic description of the model to be fitted. Example `var1 ~ var2`, where `var1` is a vector containing the dependent variable, while `var2` is a vector containing the endogenous variable. An intercept is included by default.
param a vector of initial values for the parameters of the model to be supplied to the optimization algorithm. In any model there are eight parameters. The first parameter is the intercept, then the coefficient of the endogenous variable followed by the means of the two groups of the latent IV (they need to be different, otherwise model is not identified), then the next three parameters are for the variance-covariance matrix. The last parameter is the probability of being in group 1. When not provided, initial parameters values are set equal to the OLS coefficients, the two group means are set to be equal to mean(P) and mean(P) + sd(P), the variance-covariance matrix has all elements equal to 1 while probG1 is set to equal 0.5.

data data frame or list containing the variables of the model.

Details

Let’s consider the model:

\[
Y_t = \beta_0 + \alpha P_t + \epsilon_t \\
P_t = \pi' Z_t + \nu_t
\]

where \( t = 1, \ldots, T \) indexes either time or cross-sectional units, \( Y_t \) is the dependent variable, \( P_t \) is a \( k \times 1 \) continuous, endogenous regressor, \( \epsilon_t \) is a structural error term with mean zero and \( E(\epsilon^2) = \sigma_\epsilon^2 \), \( \alpha \) and \( \beta \) are model parameters. \( Z_t \) is a \( l \times 1 \) vector of instruments, and \( \nu_t \) is a random error with mean zero and \( E(\nu^2) = \sigma_\nu^2 \). The endogeneity problem arises from the correlation of \( P \) and \( \epsilon_t \) through \( E(\epsilon \nu) = \sigma_{\epsilon \nu} \).

latentIV considers \( Z_t' \) to be a latent, discrete, exogenous variable with an unknown number of groups \( m \) and \( \pi \) is a vector of group means. It is assumed that \( Z \) is independent of the error terms \( \epsilon \) and \( \nu \) and that it has at least two groups with different means. The structural and random errors are considered normally distributed with mean zero and variance-covariance matrix \( \Sigma \):

\[
\Sigma = \begin{pmatrix}
\sigma_\epsilon^2 & \sigma_{\epsilon \nu} \\
\sigma_{\epsilon \nu} & \sigma_\nu^2
\end{pmatrix}
\]

The identification of the model lies in the assumption of the non-normality of \( P_t \), the discreteness of the unobserved instruments and the existence of at least two groups with different means.

The method has been programmed such that the latent variable has two groups. Ebbes et al.(2005) show in a Monte Carlo experiment that even if the true number of the categories of the instrument is larger than two, latentIV estimates are approximately consistent. Besides, overfitting in terms of the number of groups/categories reduces the degrees of freedom and leads to efficiency loss. When provided by the user, the initial parameter values for the two group means have to be different, otherwise the model is not identified. For a model with additional explanatory variables a Bayesian approach is needed, since in a frequentist approach identification issues appear. The optimization algorithm used is BFGS.

Value

Returns the optimal values of the parameters as computed by maximum likelihood using BFGS algorithm.

coefficients the value of the parameters for the intercept and the endogenous regressor as computed with maximum likelihood.
fitted.values  the fitted values.
meansthe value of the parameters for the means of the two categories/groups of the
latent instrumental variable.
sigmathe variance-covariance matrix sigma, where on the main diagonal are the vari-
ances of the structural error and that of the endogenous regressor and the off-
diagonal terms are equal to the covariance between the errors.
probG1  the probability of being in group one. Since the model assumes that the latent
instrumental variable has two groups, 1-probG1 gives the probability of group 2.
value  the value of the log-likelihood function corresponding to the optimal parameters.
AIC  Akaike Information Criterion.
BIC  Bayesian Information Criterion.
convcodena integer code, the same as the output returned by optimx. 0 indicates success-
ful completion. A possible error code is 1 which indicates that the iteration limit
maxit had been reached.
hessian  a symmetric matrix giving an estimate of the Hessian at the solution found.

Author(s)
The implementation of the model formula by Raluca Gui based on the paper of Ebbes et al. (2005).

References
for Regressor-Error (in)Dependence When no Instrumental Variables are Available: With New Ev-

Examples
# load data
data(dataLatentIV)
# function call without any initial parameter values
l <- latentIV(y ~ P, data = dataLatentIV)
summary(l)
# function call with initial parameter values given by the user
l1 <- latentIV(y ~ P, c(2.9,-0.85,0.0.1,1,1,0.5), data = dataLatentIV)
summary(l1)

Description
Estimates multilevel models (max. 3 levels) employing the GMM approach presented in Kim and
Frees (2007). One of the important features is that, using the hierarchical structure of the data, no
external instrumental variables are needed, unlike traditional instrumental variable techniques.
Usage

multilevelIV(formula, endoVar, data = NULL)

Arguments

formula an object of type 'formula': a symbolic description of the model to be fitted.
endoVar a matrix or data frame containing the variables assumed to be endogenous.
data optional data frame or list containing the variables of the model.

Details

When all model variables are assumed exogenous the GMM estimator is the usual GLS estimator. While the GLS model assumes all explanatory variables are uncorrelated with the random intercepts and slopes in the model, fixed effects models allow for endogeneity of all effects but sweeps out the random components as well as the explanatory variables at the same levels. The more general estimator presented here allows for some of the explanatory variables to be endogenous and uses this information to build internal instrumental variables. The multilevel GMM estimator uses both the between and within variations of the exogenous variables, but only the within variation of the variables assumed endogenous. The mixed GMM estimator equals the random effects estimator when all variables are assumed exogenous and is equal to the fixed effects estimator when all variables are assumed endogenous. In between different GMM estimators are obtained for different sets of endogenous/exogenous variables.

Value

returns the estimated coefficients together with their standard errors and p-values. It also returns the variance -covariance matrix and the weight matrix used in estimation.

coefficients the estimated coefficients.
coefsSdErr the standard errors of the estimated coefficients.
v covMat the variance-covariance matrix.
weigthMat the weight matrix used in estimation.
mixedCall the call of the estimated model.

Author(s)

The implementation of the model formula by Raluca Gui based on the paper of Kim and Frees (2007).

References


See Also

internalIV, ivreg, latentIV, copulaCorrection
Examples

```r
## Not run:
data(dataMultilevelIV)
endoVars <- matrix(dataMultilevelIV[, "X15"])
colnames(endoVars) <- c("X15")
formula1 <- y ~ X11 + X12 + X13 + X14 + X15 + X21 + X22 + X23 + X24 +
X31 + X32 + X33 + (1 + X11 | CID) + (1|SID)
model1 <- multilevelIV(formula=formula1, endoVar=endoVars, data=dataMultilevelIV)
summary(model1,"REF")

## End(Not run)
```
Index

*Topic GMM
  multilevelIV, 17
*Topic copula
  copulaPStar, 5
*Topic datasets
  dataCopC1, 6
  dataCopC2, 6
  dataCopDis, 7
  dataHetIV, 8
  dataHigherMoments, 8
  dataLatentIV, 9
  dataMultilevelIV, 9
*Topic endogeneity
  copulaPStar, 5
  internalIV, 14
  multilevelIV, 17
*Topic endogenous
  hetErrorsIV, 10
  higherMomentsIV, 12
  latentIV, 15
*Topic heteroskedastic
  hetErrorsIV, 10
*Topic instrumental
  multilevelIV, 17
*Topic instruments
  hetErrorsIV, 10
  higherMomentsIV, 12
  internalIV, 14
  latentIV, 15
  multilevelIV, 17
*Topic latent
  higherMomentsIV, 12
  latentIV, 15
*Topic lewbel
  hetErrorsIV, 10
  internalIV, 14

boots, 2, 4

copulaCorrection, 2, 3, 6, 7, 11, 18

copulaPStar, 5
dataCopC1, 6
dataCopC2, 6
dataCopDis, 7
dataHetIV, 8
dataHigherMoments, 8
dataLatentIV, 9
dataMultilevelIV, 9
hetErrorsIV, 10
higherMomentsIV, 4, 8, 11, 12, 14, 15
internalIV, 9, 14, 14, 18
ivreg, 11, 14, 18
latentIV, 9, 11, 14, 15, 18
lewbel, 11
multilevelIV, 17