Package ‘ivmodel’

January 14, 2021

Type Package
Title Statistical Inference and Sensitivity Analysis for Instrumental Variables Model
Version 1.9.0
Date 2021-01-13
Author Hyunseung Kang, Yang Jiang, Qingyuan Zhao, and Dylan Small
Maintainer Hyunseung Kang <hyunseung@stat.wisc.edu>
Imports stats,Matrix,Formula,reshape2,ggplot2
License GPL-2 | file LICENSE
LazyData true
RoxygenNote 6.0.1
NeedsCompilation no
Repository CRAN
Suggests testthat
Date/Publication 2021-01-14 06:00:03 UTC

R topics documented:

ivmodel-package ...................................................... 2
AR.power ............................................................... 5
AR.size ................................................................. 7
AR.test ................................................................. 8
AR.sens.power ....................................................... 9
AR.sens.size .......................................................... 11
AR.sens.test ........................................................... 13
balanceLovePlot ..................................................... 14
biasLovePlot .......................................................... 15
ivmodel-package

Statistical Inference and Sensitivity Analysis for Instrumental Variables Model

Description

The package fits an instrumental variables (IV) model of the following type. Let $Y$, $D$, $X$, and $Z$ represent the outcome, endogenous variable, $p$ dimensional exogenous covariates, and $L$ dimensional instruments, respectively; note that the intercept can be considered as a vector of ones and a part of the exogenous covariates $X$. The package assumes the following IV model

$$Y = X\alpha + D\beta + \epsilon, \ E(\epsilon|X,Z) = 0$$

It carries out several IV regressions, diagnostics, and tests associated with the parameter $\beta$ in the IV model. Also, if there is only one instrument, the package runs a sensitivity analysis discussed in Jiang et al. (2015).

The package is robust to most data formats, including factor and character data, and can handle very large IV models efficiently using a sparse QR decomposition.
Details

Supply the outcome \( Y \), the endogenous variable \( D \), and a data frame and/or matrix of instruments \( Z \), and a data frame and/or matrix of exogenous covariates \( X \) (optional) and run \texttt{ivmodel}. Alternatively, one can supply a formula. \texttt{ivmodel} will generate all the relevant statistics for the parameter \( \beta \).

The DESCRIPTION file:

Package: ivmodel
Type: Package
Title: Statistical Inference and Sensitivity Analysis for Instrumental Variables Model
Version: 1.9.0
Date: 2021-01-13
Author: Hyunseung Kang, Yang Jiang, Qingyuan Zhao, and Dylan Small
Maintainer: Hyunseung Kang <hyunseung@stat.wisc.edu>
Description: Carries out instrumental variable estimation of causal effects, including power analysis, sensitivity analysis.
Imports: stats, Matrix, Formula, reshape2, ggplot2
License: GPL-2 | file LICENSE
LazyData: true
RoxygenNote: 6.0.1
NeedsCompilation: no
Repository: CRAN
Suggests: testthat

Index of help topics:

\begin{itemize}
\item \texttt{AR.power} \quad Power of the Anderson-Rubin (1949) Test
\item \texttt{AR.size} \quad Sample Size Calculator for the Power of the Anderson-Rubin (1949) Test
\item \texttt{AR.test} \quad Anderson-Rubin (1949) Test
\item \texttt{ARsens.power} \quad Power of the Anderson-Rubin (1949) Test with Sensitivity Analysis
\item \texttt{ARsens.size} \quad Sample Size Calculator for the Power of the Anderson-Rubin (1949) Test with Sensitivity Analysis
\item \texttt{ARsens.test} \quad Sensitivity Analysis for the Anderson-Rubin (1949) Test
\item \texttt{CLR} \quad Conditional Likelihood Ratio Test
\item \texttt{Fuller} \quad Fuller-k Estimator
\item \texttt{IVpower} \quad Power calculation for IV models
\item \texttt{IVsize} \quad Calculating minimum sample size for achieving a certain power
\item \texttt{KClass} \quad k-Class Estimator
\item \texttt{LIML} \quad Limited Information Maximum Likelihood Ratio (LIML) Estimator
\item \texttt{TSLS.power} \quad Power of TSLS Estimator
\item \texttt{TSLS.size} \quad Sample Size Calculator for the Power of Asymptotic T-test
\end{itemize}
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>balanceLovePlot</td>
<td>Create Love plot of standardized covariate mean differences</td>
</tr>
<tr>
<td>biasLovePlot</td>
<td>Create Love plot of treatment bias and instrument bias</td>
</tr>
<tr>
<td>card.data</td>
<td>Card (1995) Data</td>
</tr>
<tr>
<td>coef.ivmodel</td>
<td>Coefficients of the Fitted Model in the 'ivmodel' Object</td>
</tr>
<tr>
<td>coefOther</td>
<td>Exogenous Coefficients of the Fitted Model in the 'ivmodel' Object</td>
</tr>
<tr>
<td>confint.ivmodel</td>
<td>Confidence Intervals for the Fitted Model in 'ivmodel' Object</td>
</tr>
<tr>
<td>distributionBalancePlot</td>
<td>Plot randomization distributions of the Mahalanobis distance</td>
</tr>
<tr>
<td>fitted.ivmodel</td>
<td>Extract Model Fitted values in the 'ivmodel' Object</td>
</tr>
<tr>
<td>getCovMeanDiffs</td>
<td>Get Covariate Mean Differences</td>
</tr>
<tr>
<td>getMD</td>
<td>Get Mahalanobis Distance</td>
</tr>
<tr>
<td>getStandardizedCovMeanDiffs</td>
<td>Get Standardized Covariate Mean Differences</td>
</tr>
<tr>
<td>icu.data</td>
<td>Pseudo-data based on Branson and Keele (2020)</td>
</tr>
<tr>
<td>iv.diagnosis</td>
<td>Diagnostics of instrumental variable analysis</td>
</tr>
<tr>
<td>ivmodel</td>
<td>Fitting Instrumental Variables (IV) Models</td>
</tr>
<tr>
<td>ivmodel-package</td>
<td>Statistical Inference and Sensitivity Analysis for Instrumental Variables Model</td>
</tr>
<tr>
<td>ivmodelFormula</td>
<td>Fitting Instrumental Variables (IV) Models</td>
</tr>
<tr>
<td>model.matrix.ivmodel</td>
<td>Extract Design Matrix for 'ivmodel' Object</td>
</tr>
<tr>
<td>para</td>
<td>Parameter Estimation from Ivmodel</td>
</tr>
<tr>
<td>permTest.absBias</td>
<td>Perform a permutation test using the sum of absolute biases</td>
</tr>
<tr>
<td>permTest.md</td>
<td>Perform a permutation test using the Mahalanobis distance</td>
</tr>
<tr>
<td>residuals.ivmodel</td>
<td>Residuals from the Fitted Model in the 'ivmodel' Object</td>
</tr>
<tr>
<td>vcov.ivmodel</td>
<td>Calculate Variance-Covariance Matrix (i.e. Standard Error) for k-Class Estimators in the 'ivmodel' Object</td>
</tr>
<tr>
<td>vcovOther</td>
<td>Variance of Exogenous Coefficients of the Fitted Model in the 'ivmodel' Object</td>
</tr>
</tbody>
</table>

**Author(s)**

Hyunseung Kang, Yang Jiang, Qingyuan Zhao, and Dylan Small

Maintainer: Hyunseung Kang <hyunseung@stat.wisc.edu>

**References**


**Examples**

```r
data(card.data) # One instrument #
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
X=card.data[,Xname]

card.model1IV = ivmodel(Y=Y,D=D,Z=Z,X=X)

card.model1IV

# Multiple instruments
Z = card.data[,c("nearc4","nearc2")]

card.model2IV = ivmodel(Y=Y,D=D,Z=Z,X=X)

card.model2IV
```

**AR.power**

*Power of the Anderson-Rubin (1949) Test*

**Description**

*AR.power* computes the power of Anderson-Rubin (1949) test based on the given values of parameters.
Usage

AR.power(n, k, l, beta, gamma, Zadj_sq,
       sigmau, sigmav, rho, alpha = 0.05)

Arguments

n       Sample size.
k       Number of exogenous variables.
l       Number of instrumental variables.
beta       True causal effect minus null hypothesis causal effect.
gamma       Regression coefficient for effect of instruments on treatment.
Zadj_sq       Variance of instruments after regressed on the observed variables.
sigmau       Standard deviation of potential outcome under control. (structural error for y)
sigmav       Standard deviation of error from regressing treatment on instruments.
rho       Correlation between u (potential outcome under control) and v (error from re-gressing treatment on instrument).
alpha       Significance level.

Value

Power of the Anderson-Rubin test based on the given values of parameters.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also ivmodel for details on the instrumental variables model.

Examples

# Assume we calculate the power of AR test in a study with one IV (l=1)
# and the only one exogenous variable is the intercept (k=1).

# Suppose the difference between the null hypothesis and true causal
effect is 1 (beta=1).
# The sample size is 250 (n=250), the IV variance is .25 (Zadj_sq =.25).
# The standard deviation of potential outcome is 1(sigmau= 1).
# The coefficient of regressing IV upon exposure is .5 (gamma= .5).
# The correlation between u and v is assumed to be .5 (rho=.5).
# The standard deviation of first stage error is .4 (sigmav=.4).
# The significance level for the study is alpha = .05.

# power of Anderson-Rubin test:
AR.power(n=250, k=1, l=1, beta=1, gamma=.5, Zadj_sq=.25,
         sigmav=1, sigmav=.4, rho=.5, alpha = 0.05)

---

**Description**

AR.size computes the minimum sample size required for achieving certain power of Anderson-Rubin (1949) test for giving value of parameters.

**Usage**

AR.size(power, k, l, beta, gamma, Zadj_sq,
         sigmav, sigmav, rho, alpha = 0.05)

**Arguments**

- **power**: The desired power over a constant.
- **k**: Number of exogenous variables.
- **l**: Number of instrumental variables.
- **beta**: True causal effect minus null hypothesis causal effect.
- **gamma**: Regression coefficient for the effect of instrument on treatment.
- **Zadj_sq**: Variance of instruments after regressed on the observed variables.
- **sigmav**: Standard deviation of potential outcome under control (structural error for y).
- **sigmav**: Standard deviation of error from regressing treatment on instruments.
- **rho**: Correlation between u (potential outcome under control) and v (error from regressing treatment on instrument).
- **alpha**: Significance level.

**Value**

Minimum sample size required for achieving certain power of Anderson-Rubin (1949) test.

**Author(s)**

Yang Jiang, Hyunseung Kang, and Dylan Small

**References**

See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

# Assume we performed an AR test in a study with one IV (l=1) and the
# only one exogenous variable is the intercept (k=1). We want to know
# the minimum sample size for this test to have an at least 0.8 power.

# Suppose the difference between the null hypothesis and true causal
# effect is 1 (beta=1).
# The IV variance is .25 (Zadj_sq =.25).
# The standard deviation of potential outcome is 1(sigmu= 1).
# The coefficient of regressing IV upon exposure is .5 (gamma= .5).
# The correlation between u and v is assumed to be .5 (rho=.5).
# The standard deviation of first stage error is .4 (sigmv=.4).
# The significance level for the study is alpha = .05.

# minimum sample size required for Anderson-Rubin test:
AR.size(power=0.8, k=1, l=1, beta=1, gamma=.5, Zadj_sq=.25,
        sigmu=1, sigmv=.4, rho=.5, alpha = 0.05)

---

**AR.test**

Anderson-Rubin (1949) Test

Description

AR.test computes the Anderson-Rubin (1949) test for the ivmodel object as well as the associated confidence interval.

Usage

AR.test(ivmodel, beta0 = 0, alpha = 0.05)

Arguments

- **ivmodel**: ivmodel object
- **beta0**: Null value $\beta_0$ for testing null hypothesis $H_0 : \beta = \beta_0$ in ivmodel. Default is 0.
- **alpha**: The significance level for hypothesis testing. Default is 0.05.

Value

AR.test returns a list containing the following components

- **Fstat**: The value of the test statistic for testing the null hypothesis $H_0 : \beta = \beta_0$ in ivmodel
- **df**: degree of freedom for the test statistic
ARsens.power

p.value The p value of the test under the null hypothesis $H_0: \beta = \beta_0$ in ivmodel
ci A matrix of two columns, each row contains an interval associated with the confidence interval
ci.info A human-readable string describing the confidence interval

Author(s)
Yang Jiang, Hyunseung Kang, and Dylan Small

References

See Also
See also ivmodel for details on the instrumental variables model.

Examples
data(card.data)
Y=card.data[,“lwage”]
D=card.data[,“educ”]
Z=card.data[,“nearc4”]
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
AR.test(foo)

ARsens.power

Power of the Anderson-Rubin (1949) Test with Sensitivity Analysis

Description
ARsens.power computes the power of sensitivity analysis, which is based on an extension of Anderson-Rubin (1949) test and allows IV be possibly invalid within a certain range.

Usage
ARsens.power(n, k, beta, gamma, Zadj_sq, sigmau, sigmav, rho, alpha = 0.05, deltarange = deltarange, delta = NULL)
Arguments

- **n**: Sample size.
- **k**: Number of exogenous variables.
- **beta**: True causal effect minus null hypothesis causal effect.
- **gamma**: Regression coefficient for effect of instruments on treatment.
- **Zadj_sq**: Variance of instruments after regressed on the observed variables.
- **sigmav**: Standard deviation of error from regressing treatment on instruments.
- **rho**: Correlation between u (potential outcome under control) and v (error from regressing treatment on instrument).
- **alpha**: Significance level.
- **deltarange**: Range of sensitivity allowance. A numeric vector of length 2.
- **delta**: True value of sensitivity parameter when calculating the power. Usually take delta = 0 for the favorable situation or delta = NULL for unknown delta.

Value

Power of sensitivity analysis for the proposed study, which extends the Anderson-Rubin (1949) test with possibly invalid IV. The power formula is derived in Jiang, Small and Zhang (2015).

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

# Assume we calculate the power of sensitivity analysis in a study with # one IV (l=1) and the only exogenous variable is the intercept (k=1).

# Suppose the difference between the null hypothesis and true causal # effect is 1 (beta=1).
# The sample size is 250 (n=250), the IV variance is .25 (Zadj_sq =.25).
# The standard deviation of potential outcome is 1(sigmav= 1).
# The coefficient of regressing IV upon exposure is .5 (gamma=.5).
# The correlation between u and v is assumed to be .5 (rho=.5).
# The standard deviation of first stage error is .4 (sigmav=.4).
# The significance level for the study is alpha = .05.

# power of sensitivity analysis under the favorable situation,
# assuming the range of sensitivity allowance is (-0.1, 0.1)
ARsens.power(n=250, k=1, beta=1, gamma=.5, Zadj_sq=.25, sigmав=1,
           sigмав=.4, rho=.5, alpha = 0.05, deltarange=c(-0.1, 0.1), delta=0)

# power of sensitivity analysis with unknown delta,
# assuming the range of sensitivity allowance is (-0.1, 0.1)
ARsens.power(n=250, k=1, beta=1, gamma=.5, Zadj_sq=.25, sigmав=1,
           sigмав=.4, rho=.5, alpha = 0.05, deltarange=c(-0.1, 0.1))

---

**Sample Size Calculator for the Power of the Anderson-Rubin (1949) Test with Sensitivity Analysis**

**Description**

ARsens.size computes the minimum sample size required for achieving certain power of sensitivity analysis, which is based on an extension of Anderson-Rubin (1949) test and allows IV be possibly invalid within a certain range.

**Usage**

```
ARsens.size(power, k, beta, gamma, Zadj_sq, sigmав, sigмав, rho,
            alpha = 0.05, deltarange = deltarange, delta = NULL)
```

**Arguments**

- **power**: The desired power over a constant.
- **k**: Number of exogenous variables.
- **beta**: True causal effect minus null hypothesis causal effect.
- **gamma**: Regression coefficient for effect of instruments on treatment.
- **Zadj_sq**: Variance of instruments after regressed on the observed covariates.
- **sigmав**: Standard deviation of potential outcome under control (structural error for y).
- **sigмав**: Standard deviation of error from regressing treatment on instruments.
- **rho**: Correlation between u (potential outcome under control) and v (error from regressing treatment on instruments).
- **alpha**: Significance level.
- **deltarange**: Range of sensitivity allowance. A numeric vector of length 2.
- **delta**: True value of sensitivity parameter when calculating power. Usually take delta = 0 for the favorable situation or delta = NULL for unknown delta.
Value

Minimum sample size required for achieving certain power of sensitivity analysis for the proposed study, which extends the Anderson-Rubin (1949) test with possibly invalid IV. The power formula is derived in Jiang, Small and Zhang (2015).

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

# Assume we performed a sensitivity analysis in a study with one
# IV (l=1) and the only exogenous variable is the intercept (k=1).
# We want to calculate the minimum sample size needed for this
# sensitivity analysis to have an at least 0.8 power.

# Suppose the difference between the null hypothesis and true causal
# effect is 1 (beta=1).
# The IV variance is .25 (Zadj_sq =.25).
# The standard deviation of potential outcome is 1(sigmau= 1).
# The coefficient of regressing IV upon exposure is .5 (gamma=.5).
# The correlation between u and v is assumed to be .5 (rho=.5).
# The standard deviation of first stage error is .4 (sigmav=.4).
# The significance level for the study is alpha = .05.

# minimum sample size for sensitivity analysis under the favorable
# situation, assuming the range of sensitivity allowance is (-0.1, 0.1)
ARsens.size(power=0.8, k=1, beta=1, gamma=.5, Zadj_sq=.25, sigmau=1,
sigmav=.4, rho=.5, alpha = 0.05, deltarange=c(-0.1, 0.1), delta=0)

# minimum sample size for sensitivity analysis with unknown delta,
# assuming the range of sensitivity allowance is (-0.1, 0.1)
ARsens.size(power=0.8, k=1, beta=1, gamma=.5, Zadj_sq=.25, sigmau=1,
sigmav=.4, rho=.5, alpha = 0.05, deltarange=c(-0.1, 0.1))
**Description**

ARsens.test computes sensitivity analysis with possibly invalid instruments, which is an extension of the Anderson-Rubin (1949) test. The formula for sensitivity analysis is derived in Jiang, Small and Zhang (2015).

**Usage**

```r
ARsens.test(ivmodel, beta0 = 0, alpha = 0.05, deltarange = NULL)
```

**Arguments**

- `ivmodel`: ivmodel object.
- `beta0`: Null value $\beta_0$ for testing null hypothesis $H_0: \beta = \beta_0$ in `ivmodel`.
- `alpha`: The significance level for hypothesis testing. Default is 0.05.

**Value**

ARsens.test returns a list containing the following components:

- `ncFstat`: The value of the test statistic for testing the null hypothesis $H_0: \beta = \beta_0$ in `ivmodel`.
- `df`: Degree of freedom for the test statistic.
- `ncp`: Non-central parameter for the test statistic.
- `p.value`: The p value of the test under the null hypothesis $H_0: \beta = \beta_0$ in `ivmodel`.
- `ci`: A matrix of two columns, each row contains an interval associated with the confidence interval.
- `ci.info`: A human-readable string describing the confidence interval.
- `deltarange`: The inputted range of sensitivity allowance.

**Author(s)**

Yang Jiang, Hyunseung Kang, and Dylan Small

**References**


balanceLovePlot  

See Also 

See also ivmodel for details on the instrumental variables model.

Examples 

data(card.data)  
Y=card.data[,"lwage"]  
D=card.data[,"educ"]  
Z=card.data[,"nearc4"]  
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",  
"reg668", "smsa66")  
X=card.data[,Xname]  
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)  
ARsens.test(foo, deltarange=c(-0.03, 0.03))

balanceLovePlot  

Create Love plot of standardized covariate mean differences

Description 

balanceLovePlot creates a Love plot of the standardized covariate mean differences across the treatment and the instrument. Can also display the permutation quantiles for these quantities. This function is used to create Figure 3a in Branson and Keele (2020).

Usage 

balanceLovePlot(X, D, Z, permQuantiles = FALSE, alpha = 0.05, perms = 1000)

Arguments 

X  Covariate matrix (with units as rows and covariates as columns).  
D  Indicator vector for a binary treatment (must contain 1 or 0 for each unit).  
Z  Indicator vector for a binary instrument (must contain 1 or 0 for each unit).  
permQuantiles  If TRUE, displays the permutation quantiles for the standardized covariate mean differences.  
alpha  The significance level used for the permutation quantiles. For example, if alpha = 0.05, then the 2.5% and 97.5% permutation quantiles are displayed.  
perms  Number of permutations used to approximate the permutation quantiles.

Value 

Plot of the standardized covariate mean differences across the treatment and the instrument.

Author(s) 

Zach Branson and Luke Keele
biasLovePlot

References


Examples

```r
# load the data
data(icu.data)
# the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
# the treatment
D = icu.data$icu_bed
# the instrument
Z = icu.data$open_bin
# make the Love plot with permutation quantiles
## Not run: balanceLovePlot(X = X, D = D, Z = Z, permQuantiles = TRUE, perms = 500)
```

---

biasLovePlot

Create Love plot of treatment bias and instrument bias

Description

biasLovePlot creates a Love plot of the bias across the treatment and the instrument. Can also display the permutation quantiles for these quantities. Note that the bias is different for the treatment than for the instrument, as discussed in Equation (3) of Branson and Keele (2020). This function is used to create Figure 3b in Branson and Keele (2020).

Usage

```r
biasLovePlot(X, D, Z, permQuantiles = FALSE, alpha = 0.05, perms = 1000)
```

Arguments

- `X`  
  Covariate matrix (with units as rows and covariates as columns).
- `D`  
  Indicator vector for a binary treatment (must contain 1 or 0 for each unit).
- `Z`  
  Indicator vector for a binary instrument (must contain 1 or 0 for each unit).
- `permQuantiles`  
  If TRUE, displays the permutation quantiles for the biases.
- `alpha`  
  The significance level used for the permutation quantiles. For example, if `alpha` = 0.05, then the 2.5% and 97.5% permutation quantiles are displayed.
- `perms`  
  Number of permutations used to approximate the permutation quantiles.

Value

Plot of the bias across the treatment and the instrument.
Author(s)

Zach Branson and Luke Keele

References


Examples

```r
#load the data
data(icu.data)
#the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#the treatment
D = icu.data$icu_bed
#the instrument
Z = icu.data$open_bin
#Make the Love plot with permutation quantiles
## Not run: biasLovePlot(X = X, D = D, Z = Z, permQuantiles = TRUE, perms = 500)
```

---

card.data

Card (1995) Data

Description

Data from the National Longitudinal Survey of Young Men (NLSYM) that was used by Card (1995).

Usage

data(card.data)

Format

A data frame with 3010 observations on the following 35 variables.

- `id` subject id
- `nearc2` indicator for whether a subject grew up near a two-year college
- `nearc4` indicator for whether a subject grew up near a four-year college
- `educ` subject’s years of education
- `age` subject’s age at the time of the survey in 1976
- `fatheduc` subject’s father’s years of education
- `motheduc` subject’s mother’s years of education
- `weight` sampling weight
- `momdad14` indicator for whether subject lived with both mother and father at age 14
sinmom14  indicator for whether subject lived with single mom at age 14
step14  indicator for whether subject lived with step-parent at age 14
reg661  indicator for whether subject lived in region 1 (New England) in 1966
reg662  indicator for whether subject lived in region 2 (Middle Atlantic) in 1966
reg663  indicator for whether subject lived in region 3 (East North Central) in 1966
reg664  indicator for whether subject lived in region 4 (West North Central) in 1966
reg665  indicator for whether subject lived in region 5 (South Atlantic) in 1966
reg666  indicator for whether subject lived in region 6 (East South Central) in 1966
reg667  indicator for whether subject lived in region 7 (West South Central) in 1966
reg668  indicator for whether subject lived in region 8 (Mountain) in 1966
reg669  indicator for whether subject lived in region 9 (Pacific) in 1966
south66  indicator for whether subject lived in South in 1966
black  indicator for whether subject’s race is black
smsa  indicator for whether subject lived in SMSA in 1976
south  indicator for whether subject lived in the South in 1976
smsa66  indicator for whether subject lived in SMSA in 1966
wage  subject’s wage in cents per hour in 1976
enroll  indicator for whether subject is enrolled in college in 1976
KWW  subject’s score on the Knowledge of the World of Work (KWW) test in 1966
IQ  IQ-type test score collected from the high school of the subject.
marr  indicator for whether the subject was married in 1976.
libcrd14  indicator for whether subject had library card at age 14.
exper  subject’s years of labor force experience in 1976
lwage  subject’s log wage in 1976
expersq  square of subject’s years of labor force experience in 1976
region  region in which subject lived in 1976

Source


Examples

data(card.data)
CLR (Conditional Likelihood Ratio Test)

Description

CLR computes the conditional likelihood ratio test (Moreira, 2003) for the ivmodel object as well as the associated confidence interval.

Usage

CLR(ivmodel, beta0 = 0, alpha = 0.05)

Arguments

- **ivmodel**: ivmodel object
- **beta0**: Null value \( \beta_0 \) for testing null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel. Default is 0
- **alpha**: The significance level for hypothesis testing. Default is 0.05

Details

CLR.test computes the conditional likelihood ratio test for the instrumental variables model in ivmodel object, specifically for the parameter \( \beta \). It also computes the \( 1 - \alpha \) confidence interval associated with it by inverting the test. The test is fully robust to weak instruments (Moreira 2003). We use the approximation suggested in Andrews et al. (2006) to evaluate the p value and the confidence interval.

Value

CLR returns a list containing the following components

- **test.stat**: The value of the test statistic for testing the null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel
- **p.value**: The p value of the test under the null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel
- **ci**: A matrix of two columns, each row contains an interval associated with the confidence interval
- **ci.info**: A human-readable string describing the confidence interval

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

```r
data(card.data)
Y = card.data[, "lwage"]
D = card.data[, "educ"]
Z = card.data[, c("nearc4", "nearc2")]
Xname = c("exper", "expersq", "black", "south", "smsa", "reg661",
          "reg668", "smsa66")
X = card.data[, Xname]

card.model2IV = ivmodel(Y = Y, D = D, Z = Z, X = X)
CLR(card.model2IV, alpha = 0.01)
```

---

**coef.ivmodel**

*Coefficients of the Fitted Model in the ivmodel Object*

### Description

This `coef` method returns the point estimation, standard error, test statistic and p value for all specified k-Class estimation from an `ivmodel` object.

### Usage

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

### Arguments

- **object**
  - `ivmodel` object.

- **...**
  - Additional arguments to `coef`.

### Value

A matrix summarizes all the k-Class estimations.

### Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

### See Also

See also `ivmodel` for details on the instrumental variables model.
Examples

data(card.data)
Y=card.data[, "lwage"]
D=card.data[, "educ"]
Z=card.data[, "nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
"reg668", "smsa66")
X=card.data[, Xname]
foo = ivmodel(Y=Y, D=D, Z=Z, X=X)
coef(foo)

c coefOther
- Exogenous Coefficients of the Fitted Model in the ivmodel Object

Description

This coefOther returns the point estimates, standard errors, test statistics and p values for the exogenous covariates associated with the outcome. It returns a list of matrices where each matrix is one of the k-Class estimates from an ivmodel object.

Usage

coefOther(ivmodel)

Arguments

ivmodel ivmodel object.

Value

A list of matrices where each matrix summarizes the estimated coefficients from one of the k-Class estimates.

Author(s)

Hyunseung Kang

See Also

See also ivmodel for details on the instrumental variables model.
Examples

data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
         "reg668", "smsa66")
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
coefOther(foo)

confint.ivmodel  Confidence Intervals for the Fitted Model in ivmodel Object

Description

This confint method returns a matrix of two columns, each row represents a confident interval for different IV approaches, which include k-Class, AR (Anderson and Rubin 1949) and CLR (Moreira 2003) estimations.

Usage

## S3 method for class 'ivmodel'
confint(object, parm, level=NULL,...)

Arguments

object  ivmodel object.
parm    Ignored for our code.
level   The confidence level.
...     Additional argument(s) for methods.

Value

A matrix, each row represents a confidence interval for different IV approaches.

Author(s)

Yag Jiang, Hyunseung Kang, and Dylan Small
References


See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

```r
data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
  "reg668", "smsa66")
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
confint(foo)
```

distributionBalancePlot

Plot randomization distributions of the Mahalanobis distance

Description

distributionBalancePlot displays the randomization distribution of the square root of the Mahalanobis distance across the treatment and/or instrument for different assignment mechanisms. This function supports complete randomization (displayed in black), block randomization (displayed in green), and Bernoulli trials for exposure (displayed in red) and instrument (displayed in blue). This function is used to create Figure 4 of Branson and Keele (2020).

Usage

distributionBalancePlot(X, D = NULL, Z = NULL, subclass = NULL,
  complete = FALSE, blocked = FALSE, bernoulli = FALSE, perms = 1000)
distributionBalancePlot

Arguments

- **X**: Covariate matrix (with units as rows and covariates as columns).
- **D**: Indicator vector for a binary treatment (must contain 1 or 0 for each unit).
- **Z**: Indicator vector for a binary instrument (must contain 1 or 0 for each unit).
- **subclass**: Vector of subclasses (one for each unit). Subclasses can be numbers or characters, as long as there is one specified for each unit. Only needed if blocked = TRUE.
- **complete**: If TRUE, displays the randomization distribution of the Mahalanobis distance under complete randomization.
- **blocked**: If TRUE, displays the randomization distribution of the Mahalanobis distance under block randomization. Needs subclass specified.
- **bernoulli**: If TRUE, displays the randomization distribution of the Mahalanobis distance under Bernoulli trials for the treatment and for the instrument.
- **perms**: Number of permutations used to approximate the randomization distributions.

Value

Plot of randomization distributions of the square root of the Mahalanobis distance across the treatment and/or instrument for different assignment mechanisms.

Author(s)

Zach Branson and Luke Keele

References


Examples

```r
#load the data
data(icu.data)
#the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#the treatment
D = icu.data$icu_bed
#the instrument
Z = icu.data$open_bin
#the subclass
subclass = icu.data$site
#make distribution plot of sqrt(MD) for
#complete randomization, block randomization, and bernoulli trials
#(just uncomment the code below)
distributionBalancePlot(X = X, D = D, Z = Z, subclass = subclass,
   complete = TRUE, blocked = TRUE, bernoulli = TRUE, perms = 500)
```
fitted.ivmodel  
Extract Model Fitted values in the ivmodel Object

Description

This fitted method returns the fitted values from k-Class estimators inside ivmodel.

Usage

```r
## S3 method for class 'ivmodel'
fitted(object,...)
```

Arguments

- `object`  ivmodel object.
- `...`  Additional arguments to `fitted`.

Value

A matrix of fitted values from the k-Class estimations. Specifically, each column of the matrix represents predicted values of the outcome for each individual based on different estimates of the treatment effect from k-Class estimators. By default, one of the columns of the matrix is the predicted outcome when the treatment effect is estimated by ordinary least squares (OLS). Because OLS is generally biased in instrumental variables settings, the predictions will likely be biased. For consistent estimates, the predictions are estimates of $E[Y \mid D,X]$. In other words, they marginalize over the unmeasured confounder $U$ and estimate the mean outcomes among all individuals with measured confounders $X$ if they were to be assigned treatment value $D$. For example, in the Card study, if $U$ represents the income of the study unit’s parents which were not measured and $X$ represents experience in years, the value of fitted for $E[Y \mid D = 16, X = 4]$ is what the average log income among individuals who had 4 years of experience would be if they were assigned 16 years of education.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

```r
data(card.data)
Y=card.data[,“lwage”]
D=card.data[,“educ”]
Z=card.data[,“nearc4”]
Xname=c(“exper”, “expersq”, “black”, “south”, “smsa”, “reg661”,
```
Fuller

Fuller-k Estimator

Description

Fuller computes the Fuller-k (Fuller 1977) estimate for the ivmodel object.

Usage

Fuller(ivmodel, 
  beta0 = 0, alpha = 0.05, b = 1, 
  manyweakSE = FALSE, heteroSE = FALSE, clusterID=NULL)

Arguments

  ivmodel       ivmodel object.
  beta0         Null value $\beta_0$ for testing null hypothesis $H_0 : \beta = \beta_0$ in ivmodel. Default is 0.
  alpha         The significance level for hypothesis testing. Default is 0.05.
  b             Positive constant $b$ in Fuller-k estimator. Default is 1.
  manyweakSE    Should many weak instrument (and heteroscedastic-robust) asymptotics in Hansen, Hausman and Newey (2008) be used to compute standard errors?
  heteroSE      Should heteroscedastic-robust standard errors be used? Default is FALSE.
  clusterID     If cluster-robust standard errors are desired, provide a vector of length that’s identical to the sample size. For example, if n = 6 and clusterID = c(1,1,1,2,2,2), there would be two clusters where the first cluster is formed by the first three observations and the second cluster is formed by the last three observations. clusterID can be numeric, character, or factor.

Details

Fuller computes the Fuller-k estimate for the instrumental variables model in ivmodel, specifically for the parameter $\beta$. The computation uses KClass with the value of $k = k_{LIML} - b/(n - L - p)$. It generates a point estimate, a standard error associated with the point estimate, a test statistic and a p value under the null hypothesis $H_0 : \beta = \beta_0$ in ivmodel along with a $1 - \alpha$ confidence interval.
Fuller returns a list containing the following components

- **k**: The k value used when computing the Fuller estimate with the k-Class estimator.
- **point.est**: Point estimate of $\beta$.
- **std.err**: Standard error of the estimate.
- **test.stat**: The value of the test statistic for testing the null hypothesis $H_0 : \beta = \beta_0$ in `ivmodel`.
- **p.value**: The p value of the test under the null hypothesis $H_0 : \beta = \beta_0$ in `ivmodel`.
- **ci**: A matrix of one row by two columns specifying the confidence interval associated with the Fuller estimator.

**Author(s)**

Yang Jiang, Hyunseung Kang, Dylan Small

**References**


**See Also**

See also `ivmodel` for details on the instrumental variables model. See also `KClass` for more information about the k-Class estimator.

**Examples**

data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,c("nearc4","nearc2")]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
  "reg668", "smsa66")
X=card.data[,Xname]
card.model2IV = ivmodel(Y=Y,D=D>Z=X)
Fuller(card.model2IV,alpha=0.01)
getCovMeanDiffs

Get Covariate Mean Differences

Description

getcovMeanDiffs returns the covariate mean differences between two groups.

Usage

getcovMeanDiffs(X, indicator)

Arguments

X Covariate matrix (with units as rows and covariates as columns).
indicator Binary indicator vector (must contain 1 or 0 for each unit). For example, could be a binary treatment or instrument.

Value

Covariate mean differences between two groups.

Author(s)

Zach Branson and Luke Keele

References


Examples

#load the data
data(icu.data)
#the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#covariate mean differences across the treatment
getcovMeanDiffs(X = X, indicator = icu.data$icu_bed)
#covariate mean differences across the instrument
getcovMeanDiffs(X = X, indicator = icu.data$open_bin)
Description

getMD returns the Mahalanobis distance between two groups.

Usage

getMD(X, indicator, covX.inv = NULL)

Arguments

X Covariate matrix (with units as rows and covariates as columns).
indicator Binary indicator vector (must contain 1 or 0 for each unit). For example, could be a binary treatment or instrument.
covX.inv Inverse of the covariate covariance matrix. Usually this is left as NULL, because getMD() will compute covX.inv for you. However, if getMD() is used many times (e.g., as in a permutation test), it can be computationally efficient to specify covX.inv beforehand.

Value

Mahalanobis distance between two groups.

Author(s)

Zach Branson and Luke Keele

References


Examples

#load the data
data(icu.data)
#the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#mahalanobis distance across the treatment
getMD(X = X, indicator = icu.data$icu_bed)
#mahalanobis distance across the instrument
getMD(X = X, indicator = icu.data$open_bin)
getStandardizedCovMeanDiffs

Get Standardized Covariate Mean Differences

Description

getStandardizedCovMeanDiffs returns the standardized covariate mean differences between two groups.

Usage

getStandardizedCovMeanDiffs(X, indicator)

Arguments

X Covariate matrix (with units as rows and covariates as columns).
indicator Binary indicator vector (must contain 1 or 0 for each unit). For example, could be a binary treatment or instrument.

Value

Standardized covariate mean differences between two groups.

Author(s)

Zach Branson and Luke Keele

References


Examples

#load the data
data(icu.data)
#the covariate matrix is
X <- as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#standardized covariate mean differences across the treatment
getStandardizedCovMeanDiffs(X = X, indicator = icu.data$icu_bed)
#standardized covariate mean differences across the instrument
getStandardizedCovMeanDiffs(X = X, indicator = icu.data$open_bin)
### icu.data

*Pseudo-data based on Branson and Keele (2020)*

#### Description
Data sampled with replacement from the original data from the (SPOT)light study used in Branson and Keele (2020). Also see Keele et al. (2018) for more details about the variables in this dataset.

#### Usage
```r
data(icu.data)
```

#### Format
A data frame with 13011 observations on the following 18 variables.

- **age**: Age of the patient in years.
- **male**: Whether or not the patient is male; 1 if male and 0 otherwise.
- **sepsis_dx**: Whether or not the patient is diagnosed with sepsis; 1 if so and 0 otherwise.
- **periarrrest**: Whether or not the patient is diagnosed with peri-arrest; 1 if so and 0 otherwise.
- **icnarc_score**: The Intensive Care National Audit and Research Centre physiological score.
- **news_score**: The National Health Service national early warning score.
- **sofa_score**: The sequential organ failure assessment score.
- **v_cc1**: Indicator for level of care at assessment (Level 0, normal ward care).
- **v_cc2**: Indicator for level of care at assessment (Level 1, normal ward care).
- **v_cc4**: Indicator for level of care at assessment (Level 2, care within a high dependency unit).
- **v_cc5**: Indicator for level of care at assessment (Level 3, ICU care).
- **v_cc_r1**: Indicator for recommended level of care at assessment (Level 0, normal ward care).
- **v_cc_r2**: Indicator for recommended level of care after assessment (Level 1, normal ward care).
- **v_cc_r4**: Indicator for recommended level of care after assessment (Level 2, care within a high dependency unit).
- **v_cc_r5**: Indicator for recommended level of care after assessment (Level 3, ICU care).
- **open_bin**: Binary instrument; 1 if the available number of ICU beds was less than 4, and 0 otherwise.
- **icu_bed**: Binary treatment; 1 if admitted to an ICU bed.
- **site**: ID for the hospital that the patient attended.

#### References

Examples

data(icu.data)

iv.diagnosis

**Diagnostics of instrumental variable analysis**

**Description**

Diagnostics of instrumental variable analysis

**Usage**

iv.diagnosis(Y, D, Z, X)

iv.diagnosis.plot(output, bias.ratio = TRUE, base_size = 15, text_size = 5)

**Arguments**

Y  A numeric vector of outcomes.
D  A vector of endogenous variables.
Z  A vector of instruments.
X  A vector, matrix or data frame of (exogenous) covariates.
output  Output from iv.diagnosis.
bias.ratio  Add bias ratios (text) to the plot?
base_size  size of the axis labels
text_size  size of the text (bias ratios)

**Value**

a list or data frame

- **x.mean1** Mean of X under Z = 1 (reported if Z is binary)
- **x.mean0** Mean of X under Z = 0 (reported if Z is binary)
- **coef** OLS coefficient of X ~ Z (reported if Z is not binary)
- **se** Standard error of OLS coefficient (reported if Z is not binary)
- **p.val** p-value of the independence of Z and X (Fisher’s test if both are binary, logistic regression if Z is binary, linear regression if Z is continuous)
- **stand.diff** Standardized difference (reported if Z is binary)
- **bias.ratio** Bias ratio
- **bias.amplify** Amplification of bias ratio
- **bias.ols** Bias of OLS
- **bias.2sls** Bias of two stage least squares)
Functions

- `iv.diagnosis.plot`: IV diagnostic plot

Author(s)

Qingyuan Zhao

References


Examples

```r
n <- 10000
Z <- rbinom(n, 1, 0.5)
X <- data.frame(matrix(c(rnorm(n), rbinom(n * 5, 1, 0.5)), n))
D <- rbinom(n, 1, plogis(Z + X[, 1] + X[, 2] + X[, 3]))
Y <- D + X[, 1] + X[, 2] + rnorm(n)
print(output <- iv.diagnosis(Y, D, Z, X))
iv.diagnosis.plot(output)

Z <- rnorm(n)
D <- rbinom(n, 1, plogis(Z + X[, 1] + X[, 2] + X[, 3]))
Y <- D + X[, 1] + X[, 2] + rnorm(n)
print(output <- iv.diagnosis(Y, D, Z, X)) # stand.diff is not reported
iv.diagnosis.plot(output)
```

ivmodel

**Fitting Instrumental Variables (IV) Models**

Description

`ivmodel` fits an instrumental variables (IV) model with one endogenous variable and a continuous outcome. It carries out several IV regressions, diagnostics, and tests associated this IV model. It is robust to most data formats, including factor and character data, and can handle very large IV models efficiently.
ivmodel

Usage

\texttt{ivmodel(Y, D, Z, X, intercept = TRUE,}
\texttt{ beta0 = 0, alpha = 0.05, k = c(0, 1),}
\texttt{ manyweakSE = FALSE, heteroSE = FALSE, clusterID = NULL,}
\texttt{ deltarange = NULL, na.action = na.omit)}

Arguments

\begin{itemize}
\item \texttt{Y} A numeric vector of outcomes.
\item \texttt{D} A vector of endogenous variables.
\item \texttt{Z} A matrix or data frame of instruments.
\item \texttt{X} A matrix or data frame of (exogenous) covariates.
\item \texttt{intercept} Should the intercept be included? Default is \texttt{TRUE} and if so, you do not need to add a column of 1s in \texttt{X}.
\item \texttt{beta0} Null value $\beta_0$ for testing null hypothesis $H_0 : \beta = \beta_0$ in \texttt{ivmodel}. Default is $0$.
\item \texttt{alpha} The significance level for hypothesis testing. Default is 0.05.
\item \texttt{k} A numeric vector of k values for k-class estimation. Default is 0 (OLS) and 1 (TSLS).
\item \texttt{manyweakSE} Should many weak instrument (and heteroscedastic-robust) asymptotics in Hansen, Hausman and Newey (2008) be used to compute standard errors? (Not supported for \texttt{k ==0})
\item \texttt{heteroSE} Should heteroscedastic-robust standard errors be used? Default is \texttt{FALSE}.
\item \texttt{clusterID} If cluster-robust standard errors are desired, provide a vector of length that’s identical to the sample size. For example, if n = 6 and \texttt{clusterID = c(1,1,1,2,2,2)}, there would be two clusters where the first cluster is formed by the first three observations and the second cluster is formed by the last three observations. \texttt{clusterID} can be numeric, character, or factor.
\item \texttt{deltarange} Range of $\delta$ for sensitivity analysis with the Anderson-Rubin (1949) test.
\item \texttt{na.action} NA handling. There are \texttt{na.fail}, \texttt{na.omit}, \texttt{na.exclude}, \texttt{na.pass} available. Default is \texttt{na.omit}.
\end{itemize}

Details

Let $Y$, $D$, $X$, and $Z$ represent the outcome, endogenous variable, $p$ dimensional exogenous covariates, and $L$ dimensional instruments, respectively. Note that the intercept is a type of exogenous covariate and can be added to $X$ by specifying \texttt{intercept} as \texttt{TRUE} (the default behavior); the user does not have to manually add an intercept column in $X$. \texttt{ivmodel} assumes the following IV model

$$Y = X\alpha + D\beta + \epsilon, E(\epsilon|X, Z) = 0$$

and produces statistics for $\beta$. In particular, \texttt{ivmodel} computes the OLS, TSLS, k-class, limited information maximum likelihood (LIML), and Fuller-k (Fuller 1977) estimates of $\beta$ using \texttt{KClass}, \texttt{LIML}, and \texttt{codeFuller}. Also, \texttt{ivmodel} computes confidence intervals and hypothesis tests of the type
\( H_0 : \beta = \beta_0 \) versus \( H_0 : \beta \neq \beta_0 \) for the said estimators as well as two weak-IV confidence intervals, Anderson and Rubin (Anderson and Rubin 1949) confidence interval (Anderson and Rubin 1949) and the conditional likelihood ratio confidence interval (Moreira 2003). Finally, the code also conducts a sensitivity analysis if \( Z \) is one-dimensional (i.e. there is only one instrument) using the method in Jiang et al. (2015).

Some procedures (e.g. conditional likelihood ratio test, sensitivity analysis with Anderson-Rubin) assume an additional linear model

\[
D = Z\gamma + X\kappa + \xi, E(\xi|X, Z) = 0
\]

**Value**

\texttt{ivmodel} returns an object of class "ivmodel".

An object class "ivmodel" is a list containing the following components

- \( n \) : Sample size.
- \( L \) : Number of instruments.
- \( p \) : Number of exogenous covariates (including intercept).
- \( Y \) : Outcome, cleaned for use in future methods.
- \( D \) : Treatment, cleaned for use in future methods.
- \( Z \) : Instrument(s), cleaned for use in future methods.
- \( X \) : Exogenous covariates (if provided), cleaned for use in future methods.
- \( \text{Yadj} \) : Adjusted outcome, projecting out \( X \).
- \( \text{Dadj} \) : Adjusted treatment, projecting out \( X \).
- \( \text{Zadj} \) : Adjusted instrument(s), projecting out \( X \).
- \( \text{ZadjQR} \) : QR decomposition for adjusted instrument(s).
- \( \text{ZXQR} \) : QR decomposition for concatenated matrix of \( Z \) and \( X \).
- \( \text{alpha} \) : Significance level for the hypothesis tests.
- \( \text{beta0} \) : Null value of the hypothesis tests.
- \( \text{kClass} \) : A list from \texttt{KClass} function.
- \( \text{LIML} \) : A list from \texttt{LIML} function.
- \( \text{Fuller} \) : A list from \texttt{Fuller} function.
- \( \text{AR} \) : A list from \texttt{AR} function.
- \( \text{CLR} \) : A list from \texttt{CLR} function.

In addition, if there is only one instrument, \texttt{ivreg} will generate an "ARsens" list within "ivmodel" object.

**Author(s)**

Yang Jiang, Hyunseung Kang, and Dylan Small
ivmodel

References


See Also

See also KClass, LIML, Fuller, AR.test, and CLR for individual methods associated with ivmodel. For extracting the estimated effect of the exogenous covariates on the outcome, see coefOther. For sensitivity analysis with the AR test, see ARsens.test. ivmodel has vcov.ivmodel, model.matrix.ivmodel, summary.ivmodel, confint.ivmodel, fitted.ivmodel, residuals.ivmodel and coef.ivmodel methods associated with it.

Examples

data(card.data)
# One instrument #
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
X=card.data[,Xname]
card.model1IV = ivmodel(Y=Y,D=D,Z=Z,X=X)
card.model1IV

# Multiple instruments
Z = card.data[,c("nearc4","nearc2")]
card.model2IV = ivmodel(Y=Y,D=D,Z=Z,X=X)
ivmodelFormula

Fitting Instrumental Variables (IV) Models

Description

ivmodelFormula fits an instrumental variables (IV) model with one endogenous variable and a continuous outcome. It carries out several IV regressions, diagnostics, and tests associated this IV model. It is robust to most data formats, including factor and character data, and can handle very large IV models efficiently.

Usage

ivmodelFormula(formula, data, subset,
  beta0=0, alpha=0.05, k=c(0,1),
  manyweakSE = FALSE,
  heteroSE = FALSE, clusterID = NULL,
  deltarange=NULL, na.action = na.omit)

Arguments

formula a formula describing the model to be fitted. For example, the formula \( Y \sim D + X_1 + X_2 \mid Z_1 + Z_2 + X_1 + X_2 \) describes the mode where

\[
Y = \alpha_0 + D \beta + X_1 \alpha_1 + X_2 \alpha_2 + \epsilon
\]

and

\[
D = \gamma_0 + Z_1 \gamma_1 + Z_2 \gamma_2 + X_1 \kappa_1 + X_2 \kappa_2 + \xi
\]

The outcome is \( Y \), the endogenous variable is \( D \). The exogenous covariates are \( X_1 \) and \( X_2 \). The instruments are \( Z_1 \) and \( Z_2 \). The formula environment follows the formula environment in the ivreg function in the AER package.

data an optional data frame containing the variables in the model. By default the variables are taken from the environment which ivmodel is called from

subset an index vector indicating which rows should be used.

beta0 Null value \( \beta_0 \) for testing null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel. Default is \$0\$.

alpha The significance level for hypothesis testing. Default is 0.05.

k A numeric vector of k values for k-class estimation. Default is 0 (OLS) and 1 (TSLS).

manyweakSE Should many weak instrument (and heteroscedastic-robust) asymptotics in Hansen, Hausman and Newey (2008) be used to compute standard errors? (Not supported for k == 0)

heteroSE Should heteroscedastic-robust standard errors be used? Default is FALSE.
clusterID  If cluster-robust standard errors are desired, provide a vector of length that’s identical to the sample size. For example, if n = 6 and clusterID = c(1,1,2,2,2), there would be two clusters where the first cluster is formed by the first three observations and the second cluster is formed by the last three observations. clusterID can be numeric, character, or factor.

deltarange  Range of \( \delta \) for sensitivity analysis with the Anderson-Rubin (1949) test.

na.action  NA handling. There are na.fail, na.omit, na.exclude, na.pass available. Default is na.omit.

Details

Let \( Y, D, X, \) and \( Z \) represent the outcome, endogenous variable, \( p \) dimensional exogenous covariates, and \( L \) dimensional instruments, respectively. \texttt{ivmodel} assumes the following IV model

\[
Y = X\alpha + D\beta + \epsilon, E(\epsilon|X,Z) = 0
\]

and produces statistics for \( \beta \). In particular, \texttt{ivmodel} computes the OLS, TSLS, k-class, limited information maximum likelihood (LIML), and Fuller-k (Fuller 1977) estimates of \( \beta \) using \texttt{KClass}, \texttt{LIML}, and \texttt{codeFuller}. Also, \texttt{ivmodel} computes confidence intervals and hypothesis tests of the type \( H_0 : \beta = \beta_0 \) versus \( H_0 : \beta \neq \beta_0 \) for the said estimators as well as two weak-IV confidence intervals, Anderson and Rubin (Anderson and Rubin 1949) confidence interval (Anderson and Rubin 1949) and the conditional likelihood ratio confidence interval (Moreira 2003). Finally, the code also conducts a sensitivity analysis if \( Z \) is one-dimensional (i.e. there is only one instrument) using the method in Jiang et al. (2015).

Some procedures (e.g. conditional likelihood ratio test, sensitivity analysis with Anderson-Rubin) assume an additional linear model

\[
D = Z\gamma + X\kappa + \xi, E(\xi|X,Z) = 0
\]

Value

\texttt{ivmodel} returns an object of class "ivmodel".

An object class "ivmodel" is a list containing the following components

\begin{itemize}
  \item \texttt{n}  Sample size.
  \item \texttt{L}  Number of instruments.
  \item \texttt{p}  Number of exogenous covariates (including intercept).
  \item \texttt{Y}  Outcome, cleaned for use in future methods.
  \item \texttt{D}  Treatment, cleaned for use in future methods.
  \item \texttt{Z}  Instrument(s), cleaned for use in future methods.
  \item \texttt{X}  Exogenous covariates (if provided), cleaned for use in future methods.
  \item \texttt{Yadj}  Adjusted outcome, projecting out \( X \).
  \item \texttt{Dadj}  Adjusted treatment, projecting out \( X \).
  \item \texttt{Zadj}  Adjusted instrument(s), projecting out \( X \).
  \item \texttt{ZadjQR}  QR decomposition for adjusted instrument(s).
\end{itemize}
ZXQR  QR decomposition for concatenated matrix of Z and X.
alpha  Significance level for the hypothesis tests.
beta0  Null value of the hypothesis tests.
kClass  A list from KClass function.
LIML  A list from LIML function.
Fuller  A list from Fuller function.
AR  A list from AR.test.
CLR  A list from CLR.

In addition, if there is only one instrument, ivreg will generate an "ARsens" list within "ivmodel" object.

Author(s)
Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also
See also KClass, LIML, Fuller, AR.test, and CLR for individual methods associated with ivmodel. For extracting the estimated effect of the exogenous covariates on the outcome, see coefOther. For sensitivity analysis with the AR test, see ARsens.test. ivmodel has vcov.ivmodel, model.matrix.ivmodel, summary.ivmodel, confint.ivmodel, fitted.ivmodel, residuals.ivmodel and coef.ivmodel methods associated with it.
Examples

data(card.data)

# One instrument #
Y = card.data[, "lwage"]
D = card.data[, "educ"]
Z = card.data[, "nearc4"]
Xname = c("exper", "expersq", "black", "south", "smsa", "reg661",
          "reg668", "smsa66")
X = card.data[, Xname]
card.model1IV = ivmodelFormula(lwage ~ educ + exper + expersq + black +
               south + smsa + reg661 +
               reg662 + reg663 + reg664 +
               reg665 + reg666 + reg667 +
               reg668 + smsa66 | nearc4 +
               exper + expersq + black +
               south + smsa + reg661 +
               reg662 + reg663 + reg664 +
               reg665 + reg666 + reg667 +
               reg668 + smsa66, data = card.data)
card.model1IV

# Multiple instruments
Z = card.data[, c("nearc4","nearc2")]
card.model2IV = ivmodelFormula(lwage ~ educ + exper + expersq + black +
               south + smsa + reg661 +
               reg662 + reg663 + reg664 +
               reg665 + reg666 + reg667 +
               reg668 + smsa66 | nearc4 + nearc2 +
               exper + expersq + black +
               south + smsa + reg661 +
               reg662 + reg663 + reg664 +
               reg665 + reg666 + reg667 +
               reg668 + smsa66, data = card.data)
card.model2IV

---

IVpower

Power calculation for IV models

Description

IVpower computes the power for one of the following tests: two stage least square estimates; Anderson-Rubin (1949) test; Sensitivity analysis.

Usage

IVpower(ivmodel, n = NULL, alpha = 0.05, beta = NULL, type = "TSLS",
deltarange = NULL, delta = NULL)
Arguments

- `ivmodel`: ivmodel object.
- `n`: number of sample size, if missing, will use the sample size from the input `ivmodel` object.
- `alpha`: The significance level for hypothesis testing. Default is 0.05.
- `beta`: True causal effect minus null hypothesis causal effect. If missing, will use the beta calculated from the input `ivmodel` object.
- `type`: Determines which test will be used for power calculation. "TSLS" for two stage least square estimates; "AR" for Anderson-Rubin test; "ARsens" for sensitivity analysis.
- `deltarange`: Range of sensitivity allowance. A numeric vector of length 2. If missing, will use the deltarange from the input `ivmodel` object.
- `delta`: True value of sensitivity parameter when calculating the power. Usually take delta = 0 for the favorable situation or delta = NULL for unknown delta.

Details

`IVpower` computes the power for one of the following tests: two stage least square estimates; Anderson-Rubin (1949) test; Sensitivity analysis. The related value of parameters will be inferred from the input of `ivmodel` object.

Value

- a power value for the specified type of test.

Author(s)

- Yang Jiang, Hyunseung Kang, Dylan Small

References


See Also

- See also `ivmodel` for details on the instrumental variables model. See also `TLSL.power`, `AR.power`, `ARsens.power` for details on the power calculation.
**Examples**

```r
data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
        "reg668", "smsa66")
X=card.data[,Xname]
card.model = ivmodel(Y=Y,D=D,Z=Z,X=X)
IVpower(card.model)
IVpower(card.model, n=10^4, type="AR")
```

---

**IVsize**

*Calculating minimum sample size for achieving a certain power*

**Description**

IVsize calculates the minimum sample size needed for achieving a certain power in one of the following tests: two stage least square estimates; Anderson-Rubin (1949) test; Sensitivity analysis.

**Usage**

```r
IVsize(ivmodel, power, alpha = 0.05, beta = NULL, type = "TSLS",
       deltarange = NULL, delta = NULL)
```

**Arguments**

- `ivmodel` : ivmodel object.
- `power` : The power threshold to achieve.
- `alpha` : The significance level for hypothesis testing. Default is 0.05.
- `beta` : True causal effect minus null hypothesis causal effect. If missing, will use the beta calculated from the input ivmodel object.
- `type` : Determines which test will be used for power calculation. "TSLS" for two stage least square estimates; "AR" for Anderson-Rubin test; "ARsens" for sensitivity analysis.
- `deltarange` : Range of sensitivity allowance. A numeric vector of length 2. If missing, will use the deltarange from the input ivmodel object.
- `delta` : True value of sensitivity parameter when calculating the power. Usually take delta = 0 for the favorable situation or delta = NULL for unknown delta.

**Details**

IVsize calculates the minimum sample size needed for achieving a certain power for one of the following tests: two stage least square estimates; Anderson-Rubin (1949) test; Sensitivity analysis. The related value of parameters will be inferred from the input of ivmodel object.
KClass

Value
minimum sample size needed for achieving a certain power

Author(s)
Yang Jiang, Hyunseung Kang, Dylan Small

References

See Also
See also ivmodel for details on the instrumental variables model. See also TSLS.size, AR.size, ARsens.size for calculation details.

Examples
```r
data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "sma", "reg661",
        "reg668", "sma66")
X=card.data[,Xname]
card.model = ivmodel(Y=Y,D=D,Z=X,X=X, deltarange=c(-0.01, 0.01))

IVsize(card.model, power=0.8)
IVsize(card.model, power=0.8, type="AR")
IVsize(card.model, power=0.8, type="ARsens", deltarange=c(-0.01, 0.01))
```

KClass

k-Class Estimator

Description
KClass computes the k-Class estimate for the ivmodel object.
KClass

Usage

KClass(ivmodel,
    beta0 = 0, alpha = 0.05, k = c(0, 1),
    manyweakSE = FALSE, heteroSE = FALSE, clusterID = NULL)

Arguments

- **ivmodel** ivmodel object.
- **beta0** Null value \( \beta_0 \) for testing null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel. Default is 0.
- **alpha** The significance level for hypothesis testing. Default is 0.05.
- **k** A vector of \( k \) values for the k-Class estimator. Default is 0 (OLS) and 1 (TSLS).
- **manyweakSE** Should many weak instrument (and heteroscedastic-robust) asymptotics in Hansen, Hausman and Newey (2008) be used to compute standard errors? (Not supported for \( k=0 \))
- **heteroSE** Should heteroscedastic-robust standard errors be used? Default is FALSE.
- **clusterID** If cluster-robust standard errors are desired, provide a vector of length that’s identical to the sample size. For example, if \( n = 6 \) and clusterID = c(1,1,1,2,2,2), there would be two clusters where the first cluster is formed by the first three observations and the second cluster is formed by the last three observations. clusterID can be numeric, character, or factor.

Details

KClass computes the k-Class estimate for the instrumental variables model in ivmodel, specifically for the parameter \( \beta \). It generates a point estimate, a standard error associated with the point estimate, a test statistic and a p value under the null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel along with a \( 1 - \alpha \) confidence interval.

Value

KClass returns a list containing the following components

- **k** A row matrix of \( k \) values supplied to KClass.
- **point.est** A row matrix of point estimates of \( \beta \), with each row corresponding to the \( k \) values supplied.
- **std.err** A row matrix of standard errors of the estimates, with each row corresponding to the \( k \) values supplied.
- **test.stat** A row matrix of test statistics for testing the null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel, with each row corresponding to the \( k \) values supplied.
- **p.value** A row matrix of p value of the test under the null hypothesis \( H_0 : \beta = \beta_0 \) in ivmodel, with each row corresponding to the \( k \) values supplied.
- **ci** A matrix of two columns specifying the confidence interval, with each row corresponding to the \( k \) values supplied.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small
See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

data(card.data)
Y = card.data[, "lwage"]
D = card.data[, "educ"]
Z = card.data[, c("nearc4", "nearc2")]
Xname = c("exper", "expersq", "black", "south", "smsa", "reg661",
  "reg668", "smsa66")
X = card.data[, Xname]
card.model2IV = ivmodel(Y=Y, D=D, Z=Z, X=X)
KClass(card.model2IV, k=c(0, 1, 0.5))

## Not run:
## The following code tests the mank weak IV standard error for LIML and Fuller.

example <- function(q = 10, rho1 = 0.5, n1 = 10000,
  sigma.uv = 0.5, beta = 1, gamma = rep(1/sqrt(q), q)) {
  Sigma1 <- outer(1:q, 1:q, function(i, j) rho1^abs(i - j))

  library(MASS)
  Z1 <- mvrnorm(n1, rep(1, q), Sigma1)
  Z1 <- matrix(2 * as.numeric(Z1 > 0) - 1, nrow = n1)
  UV1 <- mvrnorm(n1, rep(0, 2), matrix(c(1, sigma.uv, sigma.uv, 1), 2))
  X1 <- Z1
  Y1 <- X1
  list(Z1 = Z1, X1 = X1, Y1 = Y1)
}

one.sim <- function(manyweakSE) {
  data <- example(q = 100, n1 = 100)
  fit <- ivmodel(data$Y1, data$X1, data$Z1, manyweakSE = manyweakSE)
  1 > coef(fit)[, 2] - 1.96 * coef(fit)[, 3] & 1 < coef(fit)[, 2] + 1.96 * coef(fit)[, 3]
}

res <- replicate(200, one.sim(TRUE))
apply(res, 1, mean)

res <- replicate(200, one.sim(FALSE))
apply(res, 1, mean)

## End(Not run)
LIML

Limited Information Maximum Likelihood Ratio (LIML) Estimator

Description

LIML computes the LIML estimate for the ivmodel object.

Usage

LIML(ivmodel, 
  beta0 = 0, alpha = 0.05, 
  manyweakSE = FALSE, heteroSE = FALSE, clusterID = NULL)

Arguments

ivmodel ivmodel object.
beta0 Null value $\beta_0$ for testing null hypothesis $H_0 : \beta = \beta_0$ in ivmodel. Default is 0.
alpha The significance level for hypothesis testing. Default is 0.05.
manyweakSE Should many weak instrument (and heteroscedastic-robust) asymptotics in Hansen, Hausman and Newey (2008) be used to compute standard errors?
heteroSE Should heteroscedastic-robust standard errors be used? Default is FALSE.
clusterID If cluster-robust standard errors are desired, provide a vector of length that’s identical to the sample size. For example, if $n = 6$ and clusterID = c(1,1,2,2,2), there would be two clusters where the first cluster is formed by the first three observations and the second cluster is formed by the last three observations. clusterID can be numeric, character, or factor.

Details

LIML computes the LIML estimate for the instrumental variables model in ivmodel, specifically for the parameter beta. The computation uses KClass with the value of $k = k_{LIML}$, which is the smallest root of the equation

$$det(L^TL - kL^TR_ZL) = 0$$

where $L$ is a matrix of two columns, the first column consisting of the outcome vector, $Y$, and the second column consisting of the endogenous variable, $D$, and $R_Z = I - Z(Z^TZ)^{-1}Z^T$ with $Z$ being the matrix of instruments. LIML generates a point estimate, a standard error associated with the point estimate, a test statistic and a p value under the null hypothesis $H_0 : \beta = \beta_0$ in ivmodel along with a $1 - \alpha$ confidence interval.

Value

LIML returns a list containing the following components

k The k value for LIML.
point.est Point estimate of $\beta$. 
model.matrix.ivmodel

Description

This method extracts the design matrix inside ivmodel.

Usage

```r
## S3 method for class 'ivmodel'
model.matrix(object,...)
```

Arguments

- `object` ivmodel object.
- `...` Additional arguments to fitted.
Value

A design matrix for the ivmodel object.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

See Also

See also ivmodel for details on the instrumental variables model.

Examples

data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
"reg668", "smsa66")
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
model.matrix(foo)
Value

`para` returns a list containing the following components:

- **gamma**: The coefficient of IV in first stage, calculated by linear regression.
- **beta**: The TSLS estimator of the exposure effect.
- **sigmav**: Standard deviation of error from regressing treatment on instruments.
- **rho**: Correlation between u (potential outcome under control) and v (error from regressing treatment on instrument).
- **sigmav**: Standard deviation of potential outcome under control (structural error for y).

Author(s)

Yang Jiang, Hyunseung Kang, Dylan Small

See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

```r
data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
        "reg668", "smsa66")
X=card.data[,Xname]
cardfit=ivmodel(Y=Y, D=D, Z=Z, X=X)
para(cardfit)
```

---

**permTest.absBias**

*Perform a permutation test using the sum of absolute biases*

Description

`permTest.absBias` performs a permutation test for complete randomization using the sum of absolute biases as a test statistic.

Usage

```r
permTest.absBias(X, D = NULL, Z = NULL, assignment = "complete", perms = 1000, subclass = NULL)
```
Arguments

- **X**: Covariate matrix (with units as rows and covariates as columns).
- **D**: Indicator vector for a binary treatment (must contain 1 or 0 for each unit).
- **Z**: Indicator vector for a binary instrument (must contain 1 or 0 for each unit).
- **assignment**: Must be "complete", "block", or "bernoulli". Designates whether to test for complete randomization, block randomization, or Bernoulli trials.
- **subclass**: Vector of subclasses (one for each unit). Subclasses can be numbers or characters, as long as there is one specified for each unit. Only needed if **assignment** = "block".
- **perms**: Number of permutations used to approximate the permutation test.

Value

p-value testing whether or not an indicator (treatment or instrument) is as-if randomized under complete randomization (i.e., random permutations), block randomization (i.e., random permutations within subclasses), or Bernoulli trials.

Author(s)

Zach Branson and Luke Keele

References


Examples

```r
# load the data
data(icu.data)

# the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))

# the treatment
D = icu.data$icu_bed

# the instrument
Z = icu.data$open_bin

# the subclass
subclass = icu.data$site

# can uncomment the following code for examples

# permutation test for complete randomization (for the treatment)
# permTest.absBias(X = X, D = D, 
# assignment = "complete", perms = 500)

# permutation test for complete randomization (for the instrument)
# permTest.absBias(X = X, D = D, Z = Z, 
# assignment = "complete", perms = 500)

# permutation test for block randomization (for the treatment)
# permTest.absBias(X = X, D = D,
```

Perform a permutation test using the Mahalanobis distance

Description

permTest.md performs a permutation test for complete randomization using the Mahalanobis distance as a test statistic.

Usage

permTest.md(X, indicator, assignment = "complete", perms = 1000, subclass = NULL)

Arguments

X Covariate matrix (with units as rows and covariates as columns).

indicator Binary indicator vector (must contain 1 or 0 for each unit). For example, could be a binary treatment or instrument.

assignment Must be "complete", "block", or "bernoulli". Designates whether to test for complete randomization, block randomization, or Bernoulli trials.

subclass Vector of subclasses (one for each unit). Subclasses can be numbers or characters, as long as there is one specified for each unit. Only needed if assignment = "block".

 perms Number of permutations used to approximate the permutation test.

Value

p-value testing whether or not an indicator (treatment or instrument) is as-if randomized under complete randomization (i.e., random permutations), block randomization (i.e., random permutations within subclasses), or Bernoulli trials.

Author(s)

Zach Branson and Luke Keele
residuals.ivmodel

References


Examples

#load the data
data(icu.data)
#the covariate matrix is
X = as.matrix(subset(icu.data, select = -c(open_bin, icu_bed)))
#the treatment
D = icu.data$icu_bed
#the instrument
Z = icu.data$open_bin
#the subclass
subclass = icu.data$site

#can uncomment the following code for examples

#permutation test for complete randomization (for the treatment)
#permTest.md(X = X, indicator = D,
#assignment = "complete", perms = 500)
#permutation test for complete randomization (for the instrument)
#permTest.md(X = X, indicator = Z,
#assignment = "complete", perms = 500)
#permutation test for block randomization (for the treatment)
#permTest.md(X = X, indicator = D,
#assignment = "block", subclass = subclass, perms = 500)
#permutation test for block randomization (for the instrument)
#permTest.md(X = X, indicator = Z,
#assignment = "block", subclass = subclass, perms = 500)
#permutation test for bernoulli trials (for the treatment)
#permTest.md(X = X, indicator = D,
#assignment = "bernoulli", perms = 500)
#permutation test for bernoulli randomization (for the instrument)
#permTest.md(X = X, indicator = Z,
#assignment = "bernoulli", perms = 500)

residuals.ivmodel

Description

This function returns the residuals from the k-Class estimators inside the ivmodel object.

Usage

## S3 method for class 'ivmodel'
residuals(object,...)
## S3 method for class 'ivmodel'
resid(object,...)

### Arguments

- **object**: ivmodel object.
- **...**: Additional arguments to residuals or resid.

### Value

A matrix of residuals for each k-Class estimator. Specifically, each column of the matrix represents residuals for each individual based on different estimates of the treatment effect from k-Class estimators. By default, one of the columns of the matrix is the residuals when the treatment effect is estimated by ordinarly least squares (OLS). Because OLS is generally biased in instrumental variables settings, the residuals will likely be biased.

### Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

### See Also

See also `ivmodel` for details on the instrumental variables model.

### Examples

```r
data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
       "reg668", "smsa66")
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
resid(foo)
residuals(foo)
```

---

**TSLS.power**

### Description

`TSLS.power` computes the power of the asymptotic t-test of TSLS estimator.

### Usage

```r
TSLS.power(n, beta, rho_ZD, sigmanu, sigmadsq, alpha = 0.05)
```
Arguments

- `n`: Sample size.
- `beta`: True causal effect minus null hypothesis causal effect.
- `rho_ZD`: Correlation between the IV Z and the exposure D.
- `sigmau`: Standard deviation of potential outcome under control. (structural error for y)
- `sigmaDsq`: The variance of the exposure D.
- `alpha`: Significance level.

Details

The power formula is given in Freeman (2013).

Value

Power of the asymptotic t-test of TSLS estimator based on given values of parameters.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also `ivmodel` for details on the instrumental variables model.

Examples

```r
# Assume we calculate the power of asymptotic t-test of TSLS estimator
# in a study with one IV (l=1) and the only one exogenous variable is
# the intercept (k=1).

# Suppose the difference between the null hypothesis and true causal
# effect is 1 (beta=1).
# The sample size is 250 (n=250).
# The correlation between the IV and exposure is .5 (rho_ZD=.5).
# The standard deviation of potential outcome is 1 (sigmau=1).
# The variance of the exposure is 1 (sigmaDsq=1).
# The significance level for the study is alpha = .05.

# power of asymptotic t-test of TSLS estimator
TSLS.power(n=250, beta=1, rho_ZD=.5, sigmau=1, sigmaDsq=1, alpha = 0.05)
```
Sample Size Calculator for the Power of Asymptotic T-test

Description

TSLS.size computes the minimum sample size required for achieving certain power of asymptotic t-test of TSLS estimator.

Usage

TSLS.size(power, beta, rho_ZD, sigmamu, sigmaDsq, alpha = 0.05)

Arguments

- **power**: The desired power over a constant.
- **beta**: True causal effect minus null hypothesis causal effect.
- **rho_ZD**: Correlation between the IV Z and the exposure D.
- **sigmamu**: Standard deviation of potential outcome under control. (structural error for y)
- **sigmaDsq**: The variance of the exposure D.
- **alpha**: Significance level.

Details

The calculation is based on inverting the power formula given in Freeman (2013).

Value

Minimum sample size required for achieving certain power of asymptotic t-test of TSLS estimator.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

References


See Also

See also **ivmodel** for details on the instrumental variables model.
Examples

# Assume we performed an asymptotic t-test of TSLS estimator in a study
# with one IV (l=1) and the only one exogenous variable is the intercept
# (k=1). We want to calculate the minimum sample size needed for this
# test to have an at least 0.8 power.

# Suppose the null hypothesis causal effect is 0 and the true causal
# effect is 1 (beta=1-0=1).
# The correlation between the IV and exposure is .5 (rho_ZD=.5).
# The standard deviation of potential outcome is 1 (sigma_u= 1).
# The variance of the exposure is 1 (sigma_D^2=1).
# The significance level for the study is alpha = .05.

### minimum sample size required for asymptotic t-test
TSLS.size(power=.8, beta=1, rho_ZD=.5, sigma_u=1, sigma_D^2=1, alpha =.05)

<table>
<thead>
<tr>
<th>vcov.ivmodel</th>
<th>Calculate Variance-Covariance Matrix (i.e. Standard Error) for k-Class Estimators in the ivmodel Object</th>
</tr>
</thead>
</table>

Description

This `vcov` method returns the variance-covariance matrix for all specified k-Class estimation from an `ivmodel` object.

Usage

```r
### S3 method for class 'ivmodel'
vcov(object, ...)
```

Arguments

- `object` : ivmodel object.
- `...` : Additional arguments to `vcov`.

Value

A matrix of standard error estimates for each k-Class estimator.

Author(s)

Yang Jiang, Hyunseung Kang, and Dylan Small

See Also

See also `ivmodel` for details on the instrumental variables model.
Examples

data(card.data)
Y=card.data[, "lwage"]
D=card.data[, "educ"]
Z=card.data[, "nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
   "reg668", "smsa66")
X=card.data[, Xname]
foo = ivmodel(Y=Y, D=D, Z=Z, X=X)
vcov(foo)

vcovOther

Variance of Exogenous Coefficients of the Fitted Model in the ivmodel Object

Description

This vcovOther returns the estimated variances of the estimated coefficients for the exogenous covariates associated with the outcome. All the estimation is based on k-Class estimators.

Usage

vcovOther(ivmodel)

Arguments

ivmodel ivmodel object.

Value

A matrix where each row represents a k-class estimator and each column represents one of the exogenous covariates. Each element is the estimated variance of the estimated coefficients.

Author(s)

Hyunseung Kang

See Also

See also ivmodel for details on the instrumental variables model.
Examples

data(card.data)
Y=card.data[,"lwage"]
D=card.data[,"educ"]
Z=card.data[,"nearc4"]
Xname=c("exper", "expersq", "black", "south", "smsa", "reg661",
"reg668", "smsa66")
X=card.data[,Xname]
foo = ivmodel(Y=Y,D=D,Z=Z,X=X)
vcovOther(foo)
Index

* Anderson-Rubin (1949) test
  AR.power, 5
  AR.size, 7
  AR.test, 8
  ARsens.power, 9
  ARsens.size, 11
  ARsens.test, 13

* Anderson-Rubin test
  ivmodel, 32
  ivmodelFormula, 36

* Balance Plots
  balanceLovePlot, 14
  biasLovePlot, 15
  distributionBalancePlot, 22

* Card (1995) data
  card.data, 16

* Conditional likelihood ratio test
  CLR, 18
  ivmodel, 32
  ivmodelFormula, 36

* Covariate Mean Differences
  getCovMeanDiffs, 27
  getMD, 28
  getStandardizedCovMeanDiffs, 29

* Fuller-k estimator
  Fuller, 25

* Instrumental variables
  CLR, 18
  Fuller, 25
  ivmodel, 32
  ivmodel-package, 2
  ivmodelFormula, 36
  KClass, 42
  LIML, 45

* Limited information maximum likelihood (LIML) estimator
  LIML, 45

* Power
  AR.power, 5
  AR.size, 7
  ARsens.power, 9
  ARsens.size, 11
  TSLs.power, 52

* Sample size
  AR.size, 7
  ARsens.size, 11

* Sensitivity analysis with Anderson-Rubin test
  ivmodel, 32
  ivmodelFormula, 36

* Sensitivity analysis
  ARsens.power, 9
  ARsens.size, 11
  ARsens.test, 13

* datasets
  card.data, 16
  icu.data, 30

* k-Class estimator
  KClass, 42

* kClass estimation
  ivmodel, 32
  ivmodelFormula, 36

* minimum sample size
  IVsize, 41

* package
  ivmodel-package, 2

* power
  IVpower, 39
  IVsize, 41

AR.power, 5, 40
AR.size, 7, 42
AR.test, 8, 35, 38
ARsens.power, 9, 40
ARsens.size, 11, 42
ARsens.test, 13, 35, 38
INDEX

balanceLovePlot, 14
biasLovePlot, 15

card.data, 16
CLR, 18, 35, 38
test.ivmodel, 19, 35, 38
testOther, 20, 35, 38
testconfint.ivmodel, 21, 35, 38
distributionBalancePlot, 22
fitted.ivmodel, 24, 35, 38
Fuller, 25, 35, 38
getCovMeanDiffs, 27
getMD, 28
getStandardizedCovMeanDiffs, 29

icu.data, 30
iv.diagnosis, 31
ivmodel, 6, 8–10, 12, 14, 19, 20, 22, 24, 26,
32, 40, 42, 44, 46–48, 52–56
ivmodel-package, 2
ivmodelFormula, 36
IVpower, 39
IVsize, 41

KClass, 26, 35, 38, 42, 46

LIML, 35, 38, 45

model.matrix.ivmodel, 35, 38, 46

para, 47
permTest.absBias, 48
permTest.md, 50

resid.ivmodel(residuals.ivmodel), 51
residuals.ivmodel, 35, 38, 51

summary.ivmodel, 35, 38

TSLS.power, 40, 52
TSLS.size, 42, 54

vcov.ivmodel, 35, 38, 55
vcovOther, 56