Package: iv.sensemakr (via r-universe)

October 22, 2026

Type Package

Title Sensitivity Analysis Tools for Instrumental Variable Estimates

Version 0.1.0.9000

Description Implements a suite of sensitivity analysis tools for instrumental variable estimates as described in Cinelli and Hazlett (2025) <doi:10.1093/biomet/asaf004>.

License GPL-3

URL https://carloscinelli.com/iv.sensemakr/, https://github.com/carloscinelli/iv.sensemakr

BugReports https://github.com/carloscinelli/iv.sensemakr/issues

Encoding UTF-8

LazyData true

Suggests testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

RoxygenNote 7.3.3

Imports sensemakr (>= 0.1.5)

Depends R (>= 3.1.0)

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

Date/Publication 2026-04-06 16:08:11 UTC

RemoteUrl https://github.com/carloscinelli/iv.sensemakr

RemoteRef HEAD

RemoteSha 4d8890874caf5619650495ede7d4071adb65b642

Contents

card . . . 2
coef.iv.sensemakr . . . 4
coef.iv_fit . . . 5
iv_fit . . . 6
ovb_contour_plot . . . 7
ovb_minimal_reporting . . . 10
plot.iv.sensemakr . . . 11
robustness_value . . . 12
sensemakr . . . 14
sensitivity_stats . . . 16
summary.iv.sensemakr . . . 17
summary.iv_fit . . . 18

Index **20**

`card` *National Longitudinal Survey of Young Men (NLSYM)*

Description

Data used in Card (1995). Consists of a sample of 3,010 individuals from the National Longitudinal Survey of Young Men (NLSYM).

The treatment is `educ`, the outcome is `lwage` and the instrument is `nearc4`.

Usage

```
data('card')
```

Format

A data.frame with 3010 observations and 34 variables:

- **id:** person identifier
- **nearc2:** =1 if near 2 yr college, 1966
- **nearc4:** =1 if near 4 yr college, 1966
- **educ:** years of schooling, 1976
- **age:** in years
- **fatheduc:** father's schooling
- **motheduc:** mother's schooling
- **weight:** NLS sampling weight, 1976
- **momdad14:** =1 if live with mom, dad at 14
- **sinmom14:** =1 if with single mom at 14
- **step14:** =1 if with step parent at 14

- **reg661:** =1 for region 1, 1966
- **reg662:** =1 for region 2, 1966
- **reg663:** =1 for region 3, 1966
- **reg664:** =1 for region 4, 1966
- **reg665:** =1 for region 5, 1966
- **reg666:** =1 for region 6, 1966
- **reg667:** =1 for region 7, 1966
- **reg668:** =1 for region 8, 1966
- **reg669:** =1 for region 9, 1966
- **south66:** =1 if in south in 1966
- **black:** =1 if black
- **smsa:** =1 in in SMSA, 1976
- **south:** =1 if in south, 1976
- **smsa66:** =1 if in SMSA, 1966
- **wage:** hourly wage in cents, 1976
- **enroll:** =1 if enrolled in school, 1976
- **KWW:** knowledge world of work score
- **IQ:** IQ score
- **married:** =1 if married, 1976
- **libcrd14:** =1 if lib. card in home at 14
- **exper:** age - educ - 6
- **lwage:** log(wage)
- **expersq:** exper^2

References

Card, D. "Using Geographic Variation in College Proximity to Estimate the Return to Schooling". In L.N. Christofides, E.K. Grant, and R. Swidinsky, editors, Aspects of Labor Market Behaviour: Essays in Honour of John Vanderkamp. Toronto: University of Toronto Press, 1995.

Examples

```
data('card')
head(card)
```

`coef.iv.sensemakr` *Extract estimates of an* `iv.sensemakr` *object*

Description

This function extracts the estimate, lower limit, upper limit, t-value, and (extreme) robustness values of an `iv.sensemakr` object, created with the function `sensemakr`.

Usage

```
## S3 method for class 'iv.sensemakr'
coef(object, parm = "iv", ...)
```

Arguments

`object`	an object of class `iv_fit`.
`parm`	which estimate to return. Options are `"iv"` for instrumental variable estimate, `"fs"` for the first-stage estimate and `"rf"` for the reduced-form estimate.
`...`	arguments passed to other methods.

Value

A `data.frame` with the sensitivity statistics for the requested parameters.

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)
card.sens <- sensemakr(card.fit, benchmark_covariates = "black")
coef(card.sens)
coef(card.sens, parm = "fs")
```

coef.iv_fit *Extracts point estimates and confidence intervals of an* iv_fit *model.*

Description

The function coef extracts point estimates of an iv_fit model.

The function confint extracts confidence intervals of an iv_fit model.

Usage

```
## S3 method for class 'iv_fit'
coef(object, parm = "iv", ...)

## S3 method for class 'iv_fit'
confint(object, parm = c("iv", "fs", "rf"), level, ...)
```

Arguments

object	an object of class iv_fit.
parm	which estimate to return. Options are "iv" for instrumental variable estimate, "fs" for the first-stage estimate and "rf" for the reduced-form estimate.
...	arguments passed to other methods.
level	coverage level (i.e, 1-alpha). If not provided, it uses the same level as the one provided in iv_fit.

Value

coef returns a numeric vector with the estimates of interest.

confint returns a numeric vector with the confidence interval of interest.

Examples

```
# prepare data
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)

# fit iv model
card.fit <- iv_fit(y, d, z, x)

# extract coefficients
coef(card.fit)
coef(card.fit, parm = "fs")
```

```
coef(card.fit, parm = "rf")

# extract confidence intervals
confint(card.fit)
confint(card.fit, parm = "fs")
confint(card.fit, parm = "rf")
```

`iv_fit` *Instrumental Variable Estimation using the Anderson-Rubin approach*

Description

`iv_fit` computes instrumental variable estimates and confidence intervals using the Anderson-Rubin (AR) approach (Anderson and Rubin, 1949). This approach is numerically identical to Fieller's theorem (Fieller, 1954). See Cinelli and Hazlett (2025) for further discussion.

The AR point estimate is numerically identical to the point estimate of two-stage least squares (2SLS) and it is given by the ratio of the reduced-form to the first-stage regression coefficient. Confidence intervals, however, are constructed differently. 2SLS is equivalent to using the delta-method to obtain the variance of the ratio estimator, and then proceeding by assuming the ratio is asymptotically normal. This approximation can fail when instruments are "weak." The Anderson-Rubin approach instead uses a test inversion procedure to construct confidence intervals. This procedure has correct coverage regardless of instrument strength, at the (inevitable) cost of eventually obtaining unbounded confidence intervals.

Usage

```
iv_fit(y, d, z, x = NULL, h0 = 0, alpha = 0.05)
```

Arguments

`y`	`numeric vector` with the outcome.
`d`	`numeric vector` with the treatment.
`z`	`numeric vector` with the instrument.
`x`	(optional) `numeric matrix` with observed covariates.
`h0`	null hypothesis for the target parameter (the IV estimate).
`alpha`	significance test for hypothesis tests and confidence intervals.

Value

An object of class `iv_fit`, containing:

`data` A `data.frame` with the data used for fitting the models.

`models` A `list` with the `lm` models used for obtaining the IV estimates. This includes the first-stage (FS), reduced-form (RF), and Anderson-Rubin (AR) regressions.

`estimates` A `list` with the summary information of IV estimates, as well as summary information of the auxiliary estimates of the FS, RF, and AR regression.

`pars` A `list` with the parameters of the call, such as the null hypothesis `h0` and the significance level `alpha`.

References

Anderson, T.W. and Rubin, H. (1949), Estimation of the parameters of a single equation in a complete system of stochastic equations, Annals of Mathematical Statistics, 20, 46-63.

Fieller, E. C. (1954). Some problems in interval estimation. Journal of the Royal Statistical Society: Series B (Methodological), 16(2), 175-185.

Cinelli, C. and Hazlett, C. (2025), "An Omitted Variable Bias Framework for Sensitivity Analysis of Instrumental Variables." Biometrika. doi:10.1093/biomet/asaf004

Examples

```
# prepare data
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)

# fit iv model
card.fit <- iv_fit(y, d, z, x)
card.fit
summary(card.fit)
```

`ovb_contour_plot` *Contour plots of omitted variable bias for IV*

Description

Contour plots of omitted variable bias for sensitivity analysis of instrumental variable estimates.

The main inputs are an `iv_fit` model, and the covariates used for benchmarking the strength of omitted variables.

If `parm = "iv"` (default) contour plots of the IV estimate are shown. The horizontal axis of the plot shows hypothetical values of the partial R2 of latent variables with the instrument. The vertical axis shows hypothetical values of the partial R2 of latent variables with the (pot.) outcome. The contour levels represent the adjusted lower limit (or upper limit) of the Anderson-Rubin confidence interval of the IV estimate, or the t-value for testing a specific null hypothesis. The reference points are the bounds on the partial R2 of latent variables if they were k times "as strong" as the observed covariate used for benchmarking (see arguments kz and ky). The dotted red line show the chosen critical threshold (for instance, zero): latent variables with such strength (or stronger) are sufficient to invalidate the research conclusions.

if parm = "fs" or parm = "rf", then contour plots of the first-stage and reduced-form regression are shown. See, e.g, ovb_contour_plot.lm.

See Cinelli and Hazlett (2020, 2025) for details.

Usage

```
ovb_contour_plot(...)

## S3 method for class 'iv_fit'
ovb_contour_plot(
  model,
  benchmark_covariates = NULL,
  kz = 1,
  ky = kz,
  kd = kz,
  sensitivity.of = c("lwr", "upr", "t-value"),
  parm = "iv",
  r2zw.x = NULL,
  r2y0w.zx = r2zw.x,
  bound_label = "manual bound",
  xlab = NULL,
  ylab = NULL,
  ...
)
```

Arguments

...	further arguments and graphical parameters.
model	a model created with the function iv_fit.
benchmark_covariates	character vector of the names of covariates that will be used to bound the plausible strength of the latent variables.
kz	numeric vector. Parameterizes how many times stronger the latent variables are related to the instrument in comparison to the observed benchmark covariates. Default value is 1 (latent variable is as strong as benchmark covariate).
ky	numeric vector. Parameterizes how many times stronger the latent variables are related to the (pot.) outcome in comparison to the observed benchmark covariates.
kd	numeric vector. Parameterizes how many times stronger the latent variables are related to the treatment in comparison to the observed benchmark covariates. Default value is the same as kz.
sensitivity.of	should the contour plot show adjusted lower limits of confidence intervals ("lwr"), upper limit of confidence intervals ("upr") or t-values ("t-value")?
parm	contour plots of which estimate? Options are iv for instrumental variable estimates, fs for first-stage estimates, and rf for reduced-form estimates.
r2zw.x	(optional) hypothetical partial R2 of latent variables W with the instrument Z, given observed covariates X.

r2y0w.zx	(optional) hypothetical partial R2 of latent variables W with the (pot.) outcome Y(0) given Z and X. Default is the same as r2zw.x.
bound_label	label to bounds provided manually in r2zw.x and r2y0w.zx.
xlab	label of x axis. If 'NULL', default label is used.
ylab	label of y axis. If 'NULL', default label is used.

Details

Other parameters include:

alpha significance level.

threshold critical threshold, default is 0.

lim limits for the axes.

lim.x limits for the x axis. Default is lim.

lim.y limits for the y axis. Default is lim.

nlevels number of levels in the contour plot.

col.contour color of the contour lines.

col.thr.line color of the threshold line.

label.text should benchmark label texts be shown? Default is TRUE.

cex.label.text character size of label text. Default is .7.

label.bump.x bump on the x coordinate of label text.

label.bump.y bump on the y coordinate of label text.

cex.lab The magnification to be used for x and y labels relative to the current setting of cex.

cex.main The magnification to be used for main titles relative to the current setting of cex.

cex.axis The magnification to be used for axis annotation relative to the current setting of cex.

asp the y/x aspect ratio. Default is 1.

If parm = "fs" or parm = "rf" the function is simply a wrapper to the sensemakr function ovb_contour_plot.lm on the first-stage or reduced-form lm models.

Value

The function is called for its side effect of producing a contour plot. It invisibly returns a list with the grid values used for the contour plot.

References

Cinelli, C. and Hazlett, C. (2020), "Making Sense of Sensitivity: Extending Omitted Variable Bias." Journal of the Royal Statistical Society, Series B (Statistical Methodology).

Cinelli, C. and Hazlett, C. (2025), "An Omitted Variable Bias Framework for Sensitivity Analysis of Instrumental Variables." Biometrika. doi:10.1093/biomet/asaf004

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)

# contour plot of the lower CI limit
ovb_contour_plot(card.fit, sensitivity.of = "lwr",
                 benchmark_covariates = "black")
```

`ovb_minimal_reporting` *Minimal sensitivity reporting for IV estimates*

Description

This function produces LaTeX or HTML code for a minimal sensitivity analysis table for instrumental variable estimates, as suggested in Cinelli and Hazlett (2025). For objects of class `sensemakr` (from the **sensemakr** package), it dispatches to `sensemakr::ovb_minimal_reporting`.

Usage

```
ovb_minimal_reporting(
  x,
  digits = 3,
  verbose = TRUE,
  format = c("latex", "html", "pure_html"),
  ...
)
```

Arguments

`x`	an object of class `iv.sensemakr` or `sensemakr`.
`digits`	minimal number of *significant* digits.
`verbose`	if `TRUE`, the function prints the code with `cat`.
`format`	code format to print: `"latex"`, `"html"` (requires mathjax), or `"pure_html"`.
`...`	further arguments passed to the table-building functions. Optional overrides include `outcome_label` and `treatment_label`.

Value

The function returns the LaTeX or HTML code invisibly as a character string and also prints it with `cat` when `verbose = TRUE`.

References

Cinelli, C. and Hazlett, C. (2025), "An Omitted Variable Bias Framework for Sensitivity Analysis of Instrumental Variables." Biometrika. doi:10.1093/biomet/asaf004

Examples

```
# loads package
library(iv.sensemakr)

# loads dataset
data("card")

# prepares data
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)

# fits IV model and runs sensitivity analysis
card.fit <- iv_fit(y, d, z, x)
card.sens <- sensemakr(card.fit, benchmark_covariates = c("black", "smsa"))

# latex code
ovb_minimal_reporting(card.sens)

# html code (pure html, no mathjax needed)
ovb_minimal_reporting(card.sens, format = "pure_html")
```

`plot.iv.sensemakr` *Sensitivity analysis plots for IV sensemakr*

Description

This function provides the contour plots of the sensitivity analysis results obtained with the function `sensemakr` for IV. It is basically a dispatcher to the core plot function `ovb_contour_plot`.

Usage

```
## S3 method for class 'iv.sensemakr'
plot(x, sensitivity.of = c("ci", "lwr", "upr", "t-value"), parm = "iv", ...)
```

Arguments

`x`	an object of class `iv.sensemakr` created with the `sensemakr` function.
`sensitivity.of`	should the contour plot show adjusted lower limits of confidence intervals (`"lwr"`), upper limit of confidence intervals (`"upr"`) or t-values (`"t-value"`)?

parm	contour plots of which estimate? Options are `iv` for instrumental variable estimates, `fs` for first-stage estimates, and `rf` for reduced-form estimates.
...	further arguments and graphical parameters.

Value

The function is called for its side effect of producing contour plots. It invisibly returns the output from `ovb_contour_plot`.

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)
card.sens <- sensemakr(card.fit, benchmark_covariates = "black")
plot(card.sens, lim = 0.09)
```

---

robustness_value *Computes the (extreme) robustness value for IV*

Description

Computes robustness values for `iv_fit` objects, adapting the robustness value definitions of **sensemakr** to instrumental variables as described in Cinelli and Hazlett (2025). For `parm = "iv"`, returns robustness values for the IV estimate; for `parm = "fs"` or `parm = "rf"`, dispatches to **sensemakr** methods on the corresponding `lm` models.

Usage

```
robustness_value(...)

extreme_robustness_value(...)

xrv(...)

rv(...)

## S3 method for class 'iv_fit'
extreme_robustness_value(
  model,
  parm = "iv",
  q = 1,
  alpha = 0.05,
```

```
  min = TRUE,
  ...
)

## S3 method for class 'iv_fit'
robustness_value(model, parm = "iv", q = 1, alpha = 0.05, min = TRUE, ...)
```

Arguments

...	further arguments passed to or from other methods.
model	an iv_fit model
parm	parameter for which the robustness value is computed. Default is iv, meaning that the robustness value of the IV estimate is computed. Other options are to compute the robustness value of auxiliary estimates, such as the first stage (fs) or the reduced form (rf).
q	percent change of the effect estimate that would be deemed problematic. Default is 1, which means a reduction (increase) of 100% of the current effect estimate (bring estimate to zero). It has to be greater than zero.
alpha	significance level.
min	in many cases, researchers are interested in biases as large or larger than a certain amount (for instance, the strength of confounding to bring a positive estimate to zero or below). Setting min = TRUE (default) computes the robustness value for such cases. Setting min = FALSE computes the robustness value for a bias of exactly q.

Value

A numeric value with the (extreme) robustness value.

References

Cinelli, C. and Hazlett, C. (2025), "An Omitted Variable Bias Framework for Sensitivity Analysis of Instrumental Variables." Biometrika. doi:10.1093/biomet/asaf004

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)

# robustness value of the IV estimate
rv(card.fit)

# extreme robustness value
```

```
xrv(card.fit)

# robustness values for first-stage and reduced-form
rv(card.fit, parm = "fs")
rv(card.fit, parm = "rf")
```

---

sensemakr *Sensitivity Analysis of Instrumental Variable Estimates*

Description

This function performs sensitivity analysis of instrumental variable estimates, as discussed in Cinelli and Hazlett (2025). The main input is an object of class `iv_fit`. It returns an object of class `iv.sensemakr` with several pre-computed sensitivity statistics for reporting. After running `sensemakr` you may directly use the `plot`, `print` and `summary` methods in the returned object.

Usage

```
sensemakr(...)

## S3 method for class 'iv_fit'
sensemakr(
  model,
  benchmark_covariates = NULL,
  kz = 1,
  ky = kz,
  kd = kz,
  r2zw.x = NULL,
  r2y0w.zx = r2zw.x,
  bound_label = "Manual Bound",
  q = 1,
  alpha = 0.05,
  min = TRUE,
  ...
)
```

Arguments

`...`	arguments passed to other methods.
`model`	a model created with the function `iv_fit`.
`benchmark_covariates`	character vector of the names of covariates that will be used to bound the plausible strength of the latent variables.
`kz`	numeric vector. Parameterizes how many times stronger the latent variables are related to the instrument in comparison to the observed benchmark covariates. Default value is 1 (latent variable is as strong as benchmark covariate).

ky	numeric vector. Parameterizes how many times stronger the latent variables are related to the (pot.) outcome in comparison to the observed benchmark covariates.
kd	numeric vector. Parameterizes how many times stronger the latent variables are related to the treatment in comparison to the observed benchmark covariates. Default value is the same as kz.
r2zw.x	(optional) hypothetical partial R2 of latent variables W with the instrument Z, given observed covariates X.
r2y0w.zx	(optional) hypothetical partial R2 of latent variables W with the (pot.) outcome Y(0) given Z and X. Default is the same as r2zw.x.
bound_label	label to bounds provided manually in r2zw.x and r2y0w.zx.
q	percent change of the effect estimate that would be deemed problematic. Default is 1, which means a reduction of 100% of the current effect estimate (bring estimate to zero).
alpha	significance level.
min	should we consider biases as large or larger than a certain amount? Default is TRUE.

Value

An object of class iv.sensemakr, containing:

pars A list with the general parameters used when calling sensemakr.

unadjusted A list with the original, unadjusted results.

sensitivity_stats A list with the sensitivity statistics of the IV, First-Stage, and Reduced-Form regressions.

bounds A list with bounds on the strength of latent variables if they were "k times" as strong as the benchmark covariates.

References

Cinelli, C. and Hazlett, C. (2025), "An Omitted Variable Bias Framework for Sensitivity Analysis of Instrumental Variables." Biometrika. doi:10.1093/biomet/asaf004

Examples

```
# loads package
library(iv.sensemakr)

# loads dataset
data("card")

# prepares data
y <- card$lwage  # outcome
d <- card$educ   # treatment
z <- card$nearc4 # instrument
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
```

```
                    reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                  data = card) # covariates
# fits IV model
card.fit <- iv_fit(y,d,z,x)

# see results
card.fit

# runs sensitivity analysis
card.sens <- sensemakr(card.fit, benchmark_covariates = c("black", "smsa"))

# see results
card.sens

# sensitivity contour plot
plot(card.sens, lim = 0.09)
```

`sensitivity_stats` *Sensitivity statistics for instrumental variable estimates*

Description

Convenience function that computes robustness values for IV estimates as well as auxiliary first stage and reduced form regressions.

Usage

```
sensitivity_stats(...)

## S3 method for class 'iv_fit'
sensitivity_stats(model, parm = "iv", q = 1, alpha = 0.05, min = TRUE, ...)

## S3 method for class 'iv.sensemakr'
sensitivity_stats(model, parm = "iv", q = 1, alpha = 0.05, min = TRUE, ...)
```

Arguments

`...`	further arguments passed to or from other methods.
`model`	a model created with the function `iv_fit`.
`parm`	contour plots of which estimate? Options are `iv` for instrumental variable estimates, `fs` for first-stage estimates, and `rf` for reduced-form estimates.
`q`	percent change of the effect estimate that would be deemed problematic. Default is 1, which means a reduction of 100% of the current effect estimate (bring estimate to zero).
`alpha`	significance level.
`min`	should we consider biases as large or larger than a certain amount? Default is `TRUE`.

Value

A data.frame with columns for the estimate, confidence interval bounds (lower and upper), t-value, extreme robustness value (xrv_qa), robustness value (rv_qa), and the parameters used (q, min, alpha, dof).

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)

# sensitivity statistics for the IV estimate
sensitivity_stats(card.fit)

# sensitivity statistics for the first-stage
sensitivity_stats(card.fit, parm = "fs")
```

---

summary.iv.sensemakr *Sensitivity analysis print and summary methods for* iv.sensemakr

---

Description

The print and summary methods provide verbal descriptions of the sensitivity analysis results obtained with the function sensemakr.

Usage

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

## S3 method for class 'summary.iv.sensemakr'
print(x, digits = 3, ...)
```

Arguments

object	an object of class sensemakr.
...	arguments passed to other methods.
x	an object of class sensemakr.
digits	minimal number of *significant* digits.

Value

summary.iv.sensemakr returns an object of class summary.iv.sensemakr. The print methods return the object invisibly.

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)
card.sens <- sensemakr(card.fit, benchmark_covariates = "black")
print(card.sens)
summary(card.sens)
```

---

summary.iv_fit *print and summary methods for* iv_fit

Description

The print and summary methods provide verbal descriptions of the results obtained with the function iv_fit.

Usage

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

## S3 method for class 'iv_fit'
print(x, digits = 3, ...)
```

Arguments

object	an object of class iv_fit.
...	arguments passed to other methods.
x	an object of class iv_fit.
digits	minimal number of significant digits

Value

print.iv_fit returns the object x invisibly. summary.iv_fit returns an object of class summary.iv_fit. print.summary.iv_fit returns its argument invisibly.

Examples

```
data("card")
y <- card$lwage
d <- card$educ
z <- card$nearc4
x <- model.matrix( ~ exper + expersq + black + south + smsa + reg661 + reg662 +
                     reg663 + reg664 + reg665+ reg666 + reg667 + reg668 + smsa66,
                   data = card)
card.fit <- iv_fit(y, d, z, x)
print(card.fit)
summary(card.fit)
```

Index

∗ **datasets**
card, 2

card, 2
cat, *10*
coef.iv.sensemakr, 4
coef.iv_fit, 5
confint.iv_fit *(*coef.iv_fit*)*, 5

data.frame, *4*, *6*, *17*

extreme_robustness_value
*(*robustness_value*)*, 12

iv_fit, *4, 5*, 6, *7, 8*, *12–14*, *16*, *18*

list, *6, 7*
lm, *6*, *9*, *12*

matrix, *6*

numeric, *6*

ovb_contour_plot, 7, *11, 12*
ovb_contour_plot.lm, *8, 9*
ovb_minimal_reporting, 10, *10*

plot.iv.sensemakr, 11
print.iv_fit *(*summary.iv_fit*)*, 18
print.summary.iv.sensemakr
*(*summary.iv.sensemakr*)*, 17

robustness_value, 12
rv *(*robustness_value*)*, 12

sensemakr, *4*, *10, 11*, 14, *17*
sensitivity_stats, 16
summary.iv.sensemakr, 17
summary.iv_fit, 18

vector, *6*

xrv *(*robustness_value*)*, 12