Package ‘ivcheck’

May 30, 2026

Title Tests for Instrumental Variable Validity

Version 0.1.2

Description Implements tests for the identifying assumptions of instrumental variable models, the local exclusion restriction and monotonicity conditions required for local average treatment effect identification. Covers Kitagawa (2015) <doi:10.3982/ECTA11974>, Mourifie and Wan (2017) <doi:10.1162/REST_a_00622>, and Frandsen, Lefgren, and Leslie (2023) <doi:10.1257/aer.20201860>. Includes a one-shot wrapper that runs all applicable tests on a fitted instrumental variable model. Dispatches on 'fixest' and 'ivreg' model objects.

Depends R (>= 4.1.0)

License MIT + file LICENSE

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

Imports cli (>= 3.6.0), stats, parallel

Suggests testthat (>= 3.0.0), fixest, ivreg, modelsummary, broom, spelling, knitr, rmarkdown

Config/testthat/edition 3

URL https://charlescoverdale.github.io/ivcheck/, https://github.com/charlescoverdale/ivcheck

BugReports https://github.com/charlescoverdale/ivcheck/issues

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author Charles Coverdale [aut, cre]

Maintainer Charles Coverdale <charlesfcoverdale@gmail.com>

Repository CRAN

Date/Publication 2026-05-30 08:10:08 UTC

Contents

`card1995` *Card (1995) proximity-to-college extract*

Description

A data extract from the National Longitudinal Survey of Young Men, as used in Card (1995) to estimate the return to schooling using proximity to a four-year college as an instrument for years of schooling. The extract adds a binary `college` indicator (16+ years of schooling) so the data can be used with IV-validity tests that require a binary treatment.

Usage

```
card1995
```

Format

A data frame with 2991 rows and 11 variables:

id Integer row identifier.

lwage Log hourly wage in 1976 (outcome in Card's specification).

educ Years of completed schooling (continuous; Card's endogenous regressor).

college Integer 0/1 indicator for `educ >= 16`. Use this when a test requires a binary treatment.

near_college Integer 0/1 indicator for growing up near a four-year college (Card's instrument).

age Age in 1976.

exper Years of potential labour-market experience (age minus schooling minus six).

black Integer 0/1 indicator for black respondents.

south Integer 0/1 indicator for residence in the US south.

smsa Integer 0/1 indicator for residence in a Standard Metropolitan Statistical Area.

married Integer 0/1 indicator for married respondents.

Source

Card, D. (1995). Using Geographic Variation in College Proximity to Estimate the Return to Schooling. In *Aspects of Labour Market Behaviour: Essays in Honour of John Vanderkamp*, ed. L. N. Christofides, E. K. Grant, and R. Swidinsky, 201-222. University of Toronto Press. Original data from the 1966-1976 National Longitudinal Survey of Young Men. Cleaned extract via the `wooldridge` package on CRAN.

References

Card, D. (1995). Using Geographic Variation in College Proximity to Estimate the Return to Schooling. In Christofides, Grant, and Swidinsky (eds.), *Aspects of Labour Market Behaviour: Essays in Honour of John Vanderkamp*, 201-222.

Wooldridge, J. M. (2020). *wooldridge: 115 Data Sets from "Introductory Econometrics: A Modern Approach"*. R package.

Examples

```
data(card1995)
summary(card1995$lwage)
table(near_college = card1995$near_college,
      college      = card1995$college)
```

`format.iv_test` *Format method for an IV-validity test*

Description

Used when an `iv_test` object is included as a column of a data frame or tibble.

Usage

```
## S3 method for class 'iv_test'
format(x, ...)
```

Arguments

`x`	An object of class `iv_test`.
`...`	Ignored.

Value

A one-line character summary.

iv_check *Run all applicable IV-validity tests on a fitted model*

Description

Detects which tests are applicable from the structure of the fitted instrumental variable model and runs them. Returns a tidy summary with a one-line verdict.

Usage

```
iv_check(model, tests = "all", alpha = 0.05, n_boot = 1000, ...)
```

Arguments

`model`	A fitted IV model from fixest::feols or `ivreg::ivreg()`.
`tests`	Character vector of test names to run, or `"all"` (the default) to run every applicable test.
`alpha`	Significance level for the verdict. Default 0.05.
`n_boot`	Number of bootstrap replications. Default 1000.
`...`	Further arguments passed to each underlying test.

Details

Applicability is determined by:

- Kitagawa (2015) applies to any binary treatment with a discrete instrument.
- Mourifie-Wan (2017) applies to the same case, and additionally supports covariates.
- Frandsen-Lefgren-Leslie (2023) applies when the instrument is a set of mutually exclusive dummy variables (judge-IV / group design).

Value

An object of class `iv_check` containing a data frame with one row per test (test name, statistic, p-value, verdict) plus an overall verdict string.

Examples

```
if (requireNamespace("fixest", quietly = TRUE)) {
  set.seed(1)
  n <- 500
  df <- data.frame(
    z = sample(0:1, n, replace = TRUE),
    x = rnorm(n)
  )
  df$d <- rbinom(n, 1, 0.3 + 0.4 * df$z)
  df$y <- rnorm(n, mean = df$d + 0.5 * df$x)
  m <- fixest::feols(y ~ x | d ~ z, data = df)
```

```
  iv_check(m, n_boot = 200)
}
```

`iv_kitagawa` *Kitagawa (2015) / Sun (2023) test for instrument validity*

Description

Tests the joint implication of the local exclusion restriction and the local monotonicity condition in a discrete-instrument setting. Supports binary treatment (Kitagawa 2015), ordered multivalued treatment (Sun 2023 section 3), and unordered multivalued treatment (Sun 2023 section 3.3) under a user-supplied monotonicity set. The null is that the instrument is valid. Under the null, the conditional joint distribution of `(Y, D | Z)` must satisfy stochastic dominance inequalities on cumulative-tail events. Rejection is evidence that at least one of exclusion or monotonicity fails.

Usage

```
iv_kitagawa(object, ...)

## Default S3 method:
iv_kitagawa(
  object,
  d,
  z,
  n_boot = 1000,
  alpha = 0.05,
  weighting = c("variance", "unweighted"),
  weights = NULL,
  parallel = TRUE,
  se_floor = 0.15,
  treatment_order = c("ordered", "unordered"),
  monotonicity_set = NULL,
  multiplier = c("rademacher", "gaussian", "mammen"),
  ...
)

## S3 method for class 'fixest'
iv_kitagawa(
  object,
  n_boot = 1000,
  alpha = 0.05,
  weighting = c("variance", "unweighted"),
  weights = NULL,
  parallel = TRUE,
  treatment_order = c("ordered", "unordered"),
```

```
  monotonicity_set = NULL,
  multiplier = c("rademacher", "gaussian", "mammen"),
  ...
)

## S3 method for class 'ivreg'
iv_kitagawa(
  object,
  n_boot = 1000,
  alpha = 0.05,
  weighting = c("variance", "unweighted"),
  weights = NULL,
  parallel = TRUE,
  treatment_order = c("ordered", "unordered"),
  monotonicity_set = NULL,
  multiplier = c("rademacher", "gaussian", "mammen"),
  ...
)
```

Arguments

`object`	For the default method: a numeric outcome vector. For the `fixest` and `ivreg` methods: a fitted instrumental variable model from fixest::feols or `ivreg::ivreg()`.
`...`	Further arguments passed to methods.
`d`	Binary 0/1 treatment vector (default method only).
`z`	Discrete instrument (numeric or factor, default method only).
`n_boot`	Number of multiplier-bootstrap replications. Default 1000.
`alpha`	Significance level for the returned verdict. Default 0.05.
`weighting`	Test-statistic weighting. `"variance"` (default) divides each pointwise difference by its plug-in standard error estimator before taking the sup, as in Kitagawa (2015) section 4. `"unweighted"` uses the raw positive-part KS of section 3. The two are asymptotically equivalent at the boundary of the null; `"variance"` has better finite-sample power when instrument cells have unequal sizes.
`weights`	Optional survey weights. A non-negative numeric vector of length equal to the sample size. Scaled internally so the mean weight is 1.0 (preserving effective sample-size interpretation). Applied to the empirical CDFs, the bootstrap multiplier process, and the variance-weighted standard errors.
`parallel`	Logical. Run bootstrap replications in parallel on POSIX systems via parallel::mclapply. Default `TRUE`.
`se_floor`	Trimming constant `\xi` for the plug-in standard- error denominator in the variance-weighted form. Default 0.15. Kitagawa (2015) section 4 informally recommends `\xi \in [0.05, 0.1]` for balanced-Z designs. Monte Carlo at skewed Z-cell distributions with weak first stages suggests a slightly larger floor (0.15) keeps empirical size near nominal 5% without measurable power loss in the designs tested. Users reproducing Kitagawa's published examples may set `se_floor = 0.1` to match.

treatment_order
: Either "ordered" (default) or "unordered". Binary D is handled identically under both. For multivalued D, "ordered" uses cumulative-tail inequalities P(Y <= y, D <= ell | Z) and P(Y <= y, D >= ell | Z) across all pairs of instrument values, a stronger family of implications than Sun (2023) equation 10's d_min-and-d_max subset (but still valid under Sun's Assumption 2.2). "unordered" requires a user-specified monotonicity_set naming the (level, z_from, z_to) triples for which 1{D_{z_to} = d} <= 1{D_{z_from} = d} is assumed almost surely (Sun 2023 Assumption 2.4(iii)).

monotonicity_set
: A data.frame with columns d, z_from, z_to listing the triples that pin down the direction of the monotonicity restriction for treatment_order = "unordered". Ignored when treatment_order = "ordered".

multiplier
: Choice of bootstrap multiplier: "rademacher" (default; +/-1 two-point), "gaussian" (standard normal), or "mammen" (Mammen 1993 asymmetric two-point).

Details

Kitagawa (2015) equation 2.1 defines the statistic as the max over instrument-level pairs (z_low, z_high), treatment status d in {0, 1}, and intervals [y, y'] with y <= y', of the positive-part interval-probability difference normalised by the binomial-mixture plug-in standard error: T_n = sqrt(n_low * n_high / (n_low + n_high)) * max [P([y, y'], d | z_low) - P([y, y'], d | z_high)]^+ / sigma_hat. (The denominator is the pair total, not the full sample size.) The sign flips for d = 0. Instrument levels are pre-ordered by first-stage E_hat[D | Z] so the inequalities are one-sided and T_n >= 0. The implementation evaluates the sup on a quantile grid of observed outcomes (default 50 points); this is equivalent to evaluation at every sample-point pair under Kitagawa's Theorem 2.1. Critical values come from a multiplier bootstrap (section 3.2) of the pooled empirical distribution; bootstrap statistics reuse the data-derived standard-error denominator.

Value

An object of class iv_test with elements:

test
: "Kitagawa (2015)" for binary treatment; "Sun (2023)" for multivalued ordered treatment.

statistic
: Numeric test statistic (Kolmogorov-Smirnov positive-part, scaled by sqrt(n)).

p_value
: Bootstrap p-value.

alpha
: Supplied significance level.

n_boot
: Number of bootstrap replications used.

boot_stats
: Numeric vector of bootstrap test statistics.

binding
: List identifying the binding (z, z', d, y) configuration of the observed statistic.

n
: Sample size.

call
: Matched call.

References

Kitagawa, T. (2015). A Test for Instrument Validity. *Econometrica*, 83(5), 2043-2063. doi:10.3982/ECTA11974

Sun, Z. (2023). Instrument validity for heterogeneous causal effects. *Journal of Econometrics*, 237(2), 105523. doi:10.1016/j.jeconom.2023.105523

Imbens, G. W. and Angrist, J. D. (1994). Identification and Estimation of Local Average Treatment Effects. *Econometrica*, 62(2), 467-475. doi:10.2307/2951620

See Also

`iv_mw()` for the conditional version with covariates, `iv_testjfe()` for the judge-design test, and `iv_check()` for a one-shot wrapper that runs all applicable tests.

Other iv_tests: `iv_mw()`, `iv_testjfe()`

Examples

```
# Valid IV: compliers exist, no violations
set.seed(1)
n <- 500
z <- sample(0:1, n, replace = TRUE)
d <- rbinom(n, 1, 0.3 + 0.4 * z)
y <- rnorm(n, mean = d)
iv_kitagawa(y, d, z, n_boot = 200, parallel = FALSE)
```

`iv_mw` *Mourifie-Wan (2017) test for instrument validity*

Description

Reformulates the testable implications of Kitagawa (2015) as a set of conditional moment inequalities and tests them in the intersection- bounds framework of Chernozhukov, Lee, and Rosen (2013). Without covariates `x`, `iv_mw` tests the same inequalities as `iv_kitagawa` and reduces exactly to the variance-weighted Kitagawa test. With covariates, `iv_mw` estimates the conditional CDFs `F(y, d | X = x, Z = z)` nonparametrically via series regression, computes plug-in heteroscedasticity-robust standard errors, and takes the sup over `(y, x)` of the variance-weighted positive-part violation. Critical values come from a multiplier bootstrap with adaptive moment selection in the style of Andrews and Soares (2010).

Usage

```
iv_mw(object, ...)

## Default S3 method:
iv_mw(
  object,
```

```
  d,
  z,
  x = NULL,
  basis_order = 3L,
  x_grid_size = 20L,
  y_grid_size = 50L,
  adaptive = TRUE,
  grid = NULL,
  n_boot = 1000,
  alpha = 0.05,
  weights = NULL,
  parallel = TRUE,
  ...
)

## S3 method for class 'fixest'
iv_mw(
  object,
  x = NULL,
  basis_order = 3L,
  x_grid_size = 20L,
  y_grid_size = 50L,
  adaptive = TRUE,
  grid = NULL,
  n_boot = 1000,
  alpha = 0.05,
  weights = NULL,
  parallel = TRUE,
  ...
)

## S3 method for class 'ivreg'
iv_mw(
  object,
  x = NULL,
  basis_order = 3L,
  x_grid_size = 20L,
  y_grid_size = 50L,
  adaptive = TRUE,
  grid = NULL,
  n_boot = 1000,
  alpha = 0.05,
  weights = NULL,
  parallel = TRUE,
  ...
)
```

Arguments

`object` For the default method: a numeric outcome vector. For the `fixest` and `ivreg` methods: a fitted instrumental variable model from fixest::feols or `ivreg::ivreg()`.

`...` Further arguments passed to methods.

`d` Binary 0/1 treatment vector (default method only).

`z` Discrete instrument (numeric or factor, default method only).

`x` Optional numeric vector, matrix, or data frame of covariates. If supplied, the test is conditional on the first numeric column of `x`. If `NULL`, the test reduces to the unconditional Mourifie-Wan test.

`basis_order` Polynomial order of the series-regression basis used to estimate `F(y, d | X, Z)`. Default `3L` (cubic). Set to `"auto"` to select the basis order by 5-fold cross-validation over the candidates 2, 3, 4, 5 with squared-error loss on the indicator regression. When `"auto"` is used, the bootstrap becomes post-selection-valid: the test statistic is compared to the maximum of the bootstrap statistics across the candidate orders, which controls size at the nominal level against any selection rule but is mildly conservative relative to a fixed-order test. Runtime with `"auto"` is approximately four times the fixed-order path.

`x_grid_size` Number of quantile points of `x` at which to evaluate the conditional CDFs. Default 20.

`y_grid_size` Number of quantile points of `y` at which to evaluate the inequalities. Default 50.

`adaptive` Logical. If `TRUE` (default), the bootstrap uses the adaptive moment selection of Andrews-Soares (2010) with tuning parameter `kappa_n = sqrt(log(log(n)))`. If `FALSE`, uses the plug-in least-favourable critical value (conservative).

`grid` Deprecated. Ignored; use `y_grid_size` and `x_grid_size` instead.

`n_boot` Number of multiplier-bootstrap replications. Default 1000.

`alpha` Significance level for the returned verdict. Default 0.05.

`weights` Optional survey weights. A non-negative numeric vector of length equal to the sample size. Scaled internally so the mean weight is 1.0 (preserving effective sample-size interpretation). Applied to the empirical CDFs, the bootstrap multiplier process, and the variance-weighted standard errors.

`parallel` Logical. Run bootstrap replications in parallel on POSIX systems via parallel::mclapply. Default `TRUE`.

Details

The CLR framework targets conditional moment inequalities of the form `E[m(W; theta) | X] <= 0` for all `X`. Applied to Kitagawa's (2015) inequalities, the relevant moments are the positive-part differences of the conditional joint CDFs `F(y, d | X, Z)` for each `(d, z_low, z_high, y, x)` index. `iv_mw` estimates `F(y, d | X, Z)` by series regression of the indicator `1{Y <= y, D = d}` on a polynomial basis of `X` within each `Z` cell. Robust standard errors come from the heteroscedasticity-consistent sandwich of the series regression. Critical values are drawn by multiplier bootstrap: the bootstrap process reuses the plug-in SE denominator and perturbs the residuals by Rademacher weights, projected back through the basis. Adaptive moment selection includes only moments whose observed studentised statistic is within `kappa_n` of the inequality boundary, giving tighter critical values when some inequalities are strictly slack.

Value

An object of class `iv_test`; see iv_kitagawa for element descriptions. Additional elements:

`conditional`	Logical, whether covariates were supplied.
`kappa_n`	Andrews-Soares tuning parameter used (`NA` if not applicable).

References

Mourifie, I. and Wan, Y. (2017). Testing Local Average Treatment Effect Assumptions. *Review of Economics and Statistics*, 99(2), 305-313. doi:10.1162/REST_a_00622

Chernozhukov, V., Lee, S., and Rosen, A. M. (2013). Intersection Bounds: Estimation and Inference. *Econometrica*, 81(2), 667-737. doi:10.3982/ECTA8718

Imbens, G. W. and Angrist, J. D. (1994). Identification and Estimation of Local Average Treatment Effects. *Econometrica*, 62(2), 467-475. doi:10.2307/2951620

See Also

`iv_kitagawa()` for the unconditional case, `iv_testjfe()` for the judge-design test, and `iv_check()` for a one-shot wrapper that runs all applicable tests.

Other iv_tests: `iv_kitagawa()`, `iv_testjfe()`

Examples

```
set.seed(1)
n <- 500
z <- sample(0:1, n, replace = TRUE)
d <- rbinom(n, 1, 0.3 + 0.4 * z)
y <- rnorm(n, mean = d)
iv_mw(y, d, z, n_boot = 200, parallel = FALSE)
```

`iv_power` *Monte Carlo power curve for IV-validity tests*

Description

Simulates data under a user-specified deviation from validity and estimates the rejection probability of the chosen test at each deviation size. Useful for sample-size planning and for benchmarking different tests on the same design.

Usage

```
iv_power(
  y,
  d,
  z,
  method = c("kitagawa", "mw", "testjfe"),
  alpha = 0.05,
  n_sims = 500,
  delta_grid = NULL,
  n_boot = 200,
  parallel = TRUE,
  ...
)
```

Arguments

`y, d, z`	Observed data used to anchor the DGP (sample size, cell counts, empirical first-stage).
`method`	Which test to benchmark. One of `"kitagawa"`, `"mw"`, or `"testjfe"`.
`alpha`	Significance level.
`n_sims`	Number of Monte Carlo simulations per deviation.
`delta_grid`	Numeric vector of deviation sizes to evaluate. If `NULL`, defaults to `seq(0, 0.3, by = 0.05)`.
`n_boot`	Number of bootstrap replications per simulation (for tests that use bootstrap). Default 200, which trades some Monte Carlo noise for tractable runtime.
`parallel`	Logical. Run simulations in parallel on POSIX systems via parallel::mclapply. Default `TRUE`.
`...`	Further arguments passed to the underlying test.

Details

The deviation is parameterised as the size of a **D-specific direct effect of the instrument on the outcome** (a clean exclusion violation that the Kitagawa and Mourifie-Wan tests are designed to detect). Specifically, the simulated outcome is `Y = mu_hat[D + 1] + delta * sigma_hat * D * (Z - Z_low) + noise`, so `delta = 0` corresponds to the null and larger values produce larger violations of the testable inequality for the d = 1 cells. The simulator preserves the observed sample size, first-stage propensities, and outcome scale.

Value

A data frame with columns `delta` (deviation size) and `power` (estimated rejection probability at level `alpha`).

Examples

```
# Headline power curve for a small-N design
set.seed(1)
```

```
n <- 300
z <- sample(0:1, n, replace = TRUE)
d <- rbinom(n, 1, 0.3 + 0.4 * z)
y <- rnorm(n, mean = d)
iv_power(y, d, z, method = "kitagawa", n_sims = 50, n_boot = 100)
```

---

iv_testjfe *Frandsen-Lefgren-Leslie (2023) test for instrument validity in judge-fixed-effects designs*

Description

Jointly tests the local exclusion and monotonicity assumptions when the instruments are a set of mutually exclusive dummy variables (the leniency-of-assigned-judge design). Supports binary and multivalued discrete treatments. Under the joint null, the per-judge mean outcome `mu_j = E[Y | J = j]` must be a linear function of the per-judge treatment propensities `P(D = d | J = j)`. Rejection is evidence that at least one of exclusion or monotonicity fails.

Usage

```
iv_testjfe(object, ...)

## Default S3 method:
iv_testjfe(
  object,
  d,
  z,
  x = NULL,
  n_boot = 1000,
  alpha = 0.05,
  method = c("asymptotic", "bootstrap"),
  weights = NULL,
  basis_order = 1L,
  parallel = TRUE,
  ...
)

## S3 method for class 'fixest'
iv_testjfe(
  object,
  x = NULL,
  n_boot = 1000,
  alpha = 0.05,
  method = c("asymptotic", "bootstrap"),
  weights = NULL,
```

```
  basis_order = 1L,
  parallel = TRUE,
  ...
)

## S3 method for class 'ivreg'
iv_testjfe(
  object,
  x = NULL,
  n_boot = 1000,
  alpha = 0.05,
  method = c("asymptotic", "bootstrap"),
  weights = NULL,
  basis_order = 1L,
  parallel = TRUE,
  ...
)
```

Arguments

`object`	For the default method: a numeric outcome vector. For the `fixest` and `ivreg` methods: a fitted instrumental variable model from fixest::feols or `ivreg::ivreg()`.
`...`	Further arguments passed to methods.
`d`	Binary 0/1 treatment vector (default method only).
`z`	Factor, integer, or matrix of mutually exclusive dummy variables identifying the judge (or other random-assignment unit).
`x`	Optional numeric vector, matrix, or data frame of covariates. If supplied, `y` and `d` are residualised on `x` before the per- judge means are computed.
`n_boot`	Number of multiplier-bootstrap replications. Default 1000.
`alpha`	Significance level for the returned verdict. Default 0.05.
`method`	Reference distribution for the p-value. `"asymptotic"` (default) uses the chi-squared with `K - (basis_order + 1)` degrees of freedom. `"bootstrap"` uses the multiplier bootstrap of the restricted-model residual process. Asymptotic is fast and accurate for moderate `K`; bootstrap is preferred for small `K` or if errors are far from normal.
`weights`	Optional survey weights. A non-negative numeric vector of length equal to the sample size. Scaled internally so the mean weight is 1.0 (preserving effective sample-size interpretation). Applied to the empirical CDFs, the bootstrap multiplier process, and the variance-weighted standard errors.
`basis_order`	Order of the polynomial basis used to approximate the outcome / propensity function `phi(p)` in Frandsen-Lefgren-Leslie (2023) step 1. Default `1L` reduces to the Sargan-Hansen overidentification form, which imposes constant treatment effects. Values above 1 relax this to `phi(p) = delta_0 + delta_1 p + delta_2 p^2 + ... + delta_m p` and test the joint-zero restriction on judge residuals under the richer fit. Only binary treatment is supported when `basis_order > 1`. The slope-bounded moment-inequality component of the FLL test is not implemented in v0.1.0 (deferred to v0.2.0).

`parallel`	Logical. Run bootstrap replications in parallel on POSIX systems via parallel::mclapply. Default `TRUE`.

Details

Under the joint null, each pair of judges `(j, k)` identifies the same complier LATE via the Wald estimator `(mu_j - mu_k) / (p_j - p_k)`. The Frandsen-Lefgren-Leslie (2023) test is the overidentification test of "all pairwise LATEs equal". Under binary treatment with WLS weighting, that overidentification test is algebraically the weighted sum of squared residuals from the linear fit `mu_j = alpha + beta * p_j`, divided by a pooled variance estimator. `iv_testjfe` computes this quadratic form and, by default, compares to a chi-squared distribution with `K - 2` degrees of freedom (the FLL asymptotic form). The multiplier bootstrap of the restricted residual process is available via `method = "bootstrap"` for small-K robustness.

Note on finite-sample size. Per-judge propensities `p_j` enter the test as estimated regressors. At modest per-judge sample sizes (`n_j` below a few hundred), finite-sample binomial noise in `hat p_j` compresses the distribution of the test statistic below the asymptotic chi-squared reference, producing a test that is mildly conservative at nominal 5 percent. Empirical size at `K = 20`, `N = 3000` is 1.5 percent under the asymptotic method and 2.5 percent under the bootstrap. Both methods sharpen toward nominal as `n_j` grows. The bootstrap is recommended for publication-grade p-values at modest `n_j`.

The returned object includes `pairwise_late`, the `K x K` matrix of pairwise Wald LATE estimates, and `worst_pair`, the judge pair with the largest absolute deviation from the fitted slope. These are diagnostic outputs in the sense of the paper's Figure 2: a pair whose Wald LATE deviates far from the common slope is the first place to look when investigating a rejection.

Multivalued treatment is supported: for `D` with `M + 1` distinct values (`0, 1, ..., M`), the fit becomes a multiple WLS regression of `mu_j` on the `M`-vector (`P(D = 1 | J), ..., P(D = M | J)`) and the test statistic is compared to `chi^2_{K - M - 1}` (FLL 2023 section 4). `pairwise_late` and `worst_pair` are only defined for binary `D` and return `NULL` otherwise.

Value

An object of class `iv_test`; see iv_kitagawa for element descriptions. Additional elements:

`n_judges`	Number of distinct judges / assignment groups.
`coef`	Fitted weighted-LS slope and intercept of `mu_j` on `p_j`.
`pairwise_late`	`K x K` matrix of pairwise Wald LATE estimates `(mu_j - mu_k) / (p_j - p_k)`. Under the null every entry estimates the common complier LATE.
`worst_pair`	List identifying the judge pair with the largest deviation of its Wald LATE from the fitted slope; useful for diagnosing the source of a rejection.

References

Frandsen, B. R., Lefgren, L. J., and Leslie, E. C. (2023). Judging Judge Fixed Effects. *American Economic Review*, 113(1), 253-277. doi:10.1257/aer.20201860

Imbens, G. W. and Angrist, J. D. (1994). Identification and Estimation of Local Average Treatment Effects. *Econometrica*, 62(2), 467-475. doi:10.2307/2951620

See Also

`iv_kitagawa()` for the unconditional binary-treatment test, `iv_mw()` for the conditional version with covariates, and `iv_check()` for a one-shot wrapper that runs all applicable tests.

Other iv_tests: `iv_kitagawa()`, `iv_mw()`

Examples

```
set.seed(1)
n <- 2000
judge <- sample.int(20, n, replace = TRUE)
d <- rbinom(n, 1, 0.3 + 0.02 * judge)
y <- rnorm(n, mean = d)
iv_testjfe(y, d, judge, n_boot = 200, parallel = FALSE)
```

---

plot.iv_test *Plot method for an IV-validity test*

Description

Plots the bootstrap distribution of the test statistic with the observed statistic and the rejection region highlighted.

Usage

```
## S3 method for class 'iv_test'
plot(x, ...)
```

Arguments

`x`	An object of class `iv_test`.
`...`	Further graphical arguments passed to graphics::hist.

Value

Invisibly returns `x`.

print.iv_check — *Print method for an iv_check result*

Description

Print method for an iv_check result

Usage

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

Arguments

`x`	An object of class `iv_check`.
`digits`	Number of significant digits.
`...`	Ignored.

Value

Invisibly returns `x`.

print.iv_test — *Print method for an IV-validity test*

Description

Print method for an IV-validity test

Usage

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

Arguments

`x`	An object of class `iv_test`.
`digits`	Number of significant digits to display.
`...`	Ignored.

Value

Invisibly returns `x`.

`summary.iv_test` *Summary method for an IV-validity test*

Description

Summary method for an IV-validity test

Usage

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

Arguments

`object`	An object of class `iv_test`.
`...`	Ignored.

Value

Invisibly returns `object`.

Index