Package 'hetid'

June 13, 2025

Title A package for identification through heteroskedasticity a la Lewbel (2012)

Version 0.1.0

Description This package implements the identification through heteroskedasticity method of Lewbel (2012) for time-series models with endogenous regressors. It provides tools for estimation and inference when traditional instruments are not available.

License MIT + file LICENSE

URL https://github.com/fernando-duarte/heteroskedasticity_identification

BugReports
https://github.com/fernando-duarte/heteroskedasticity_identification/issues

Depends R (>= 4.2)

Imports AER,
boot,
dplyr,
furrr,
future,
ggplot2,
knitr,
magrittr,
purrr,
rlang,
stats,
tidyr,
utils

Suggests knitr,
rmarkdown,
testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Contents

---

analyze_bootstrap_results

Analyze Bootstrap Results

Description

Analyzes and displays bootstrap standard error results for set identification bounds.

Usage

```
analyze_bootstrap_results(results_main, bootstrap_demo, config, verbose = TRUE)
```

Arguments

results_main	Data.frame. Main simulation results.
bootstrap_demo	Data.frame. Bootstrap demonstration results.
config	List. Configuration object.
verbose	Logical. Whether to print output (default: TRUE).

Value

A data.frame with bootstrap examples.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_main <- run_main_simulation(config, seeds)
bootstrap_demo <- run_bootstrap_demonstration(config, seeds)
bootstrap_analysis <- analyze_bootstrap_results(results_main, bootstrap_demo, config)

## End(Not run)
```

analyze_main_results   *Analyze Main Simulation Results*

Description

Provides comprehensive analysis of the main Monte Carlo simulation results, including performance metrics for point estimators and set identification.

Usage

```
analyze_main_results(results, config, verbose = TRUE)
```

Arguments

results	Data.frame. Results from run_main_simulation().
config	List. Configuration object used for the simulation.
verbose	Logical. Whether to print detailed output (default: TRUE).

Value

A list containing summary tables and statistics.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results <- run_main_simulation(config, seeds)
analysis <- analyze_main_results(results, config)

## End(Not run)
```

---

analyze_sample_size_results
Analyze Sample Size Results

Description

Analyzes consistency of estimators across different sample sizes.

Usage

```
analyze_sample_size_results(results_by_n, config, verbose = TRUE)
```

Arguments

results_by_n   Data.frame. Results from run_sample_size_analysis().

config   List. Configuration object.

verbose   Logical. Whether to print output (default: TRUE).

Value

A data.frame with sample size analysis.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_by_n <- run_sample_size_analysis(config, seeds)
size_analysis <- analyze_sample_size_results(results_by_n, config)

## End(Not run)
```

---

analyze_sensitivity_results
Analyze Sensitivity Results

Description

Analyzes sensitivity of results to heteroscedasticity strength.

Usage

```
analyze_sensitivity_results(results_by_delta, config, verbose = TRUE)
```

Arguments

results_by_delta

        Data.frame. Results from run_sensitivity_analysis().

config   List. Configuration object.

verbose   Logical. Whether to print output (default: TRUE).

Value

A data.frame with sensitivity analysis.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_by_delta <- run_sensitivity_analysis(config, seeds)
sensitivity_analysis <- analyze_sensitivity_results(results_by_delta, config)

## End(Not run)
```

---

calculate_lewbel_bounds

Calculate Set Identification Bounds for Lewbel Estimator

Description

Computes set identification bounds for the endogenous parameter under a relaxed covariance restriction. Optionally computes bootstrap standard errors for the bounds.

Usage

```
calculate_lewbel_bounds(data, tau, compute_se = FALSE, B = 100)
```

Arguments

data	Data.frame. Dataset containing Y1, Y2, Xk, Z variables.
tau	Numeric. Relaxation parameter for covariance restriction (0 <= tau < 1). When tau = 0, gives point identification.
compute_se	Logical. Whether to compute bootstrap standard errors (default: FALSE).
B	Integer. Number of bootstrap replications if compute_se = TRUE (default: 100).

Details

Under the relaxed assumption $|\text{Corr}(Z, \epsilon_1\epsilon_2)| <= \tau\ |\text{Corr}(Z, \epsilon_2^2)|$, the parameter gamma_1 is set-identified. The bounds are computed as the real roots of a quadratic equation in gamma_1.

Value

A list containing:

- bounds: Numeric vector of length 2 with lower and upper bounds

- se: Numeric vector of length 2 with bootstrap standard errors (if requested)

Examples

```
## Not run:
params <- list(
  beta1_0 = 0.5, beta1_1 = 1.5, gamma1 = -0.8,
  beta2_0 = 1.0, beta2_1 = -1.0,
  alpha1 = -0.5, alpha2 = 1.0, delta_het = 1.2
)
data <- generate_lewbel_data(1000, params)

# Point identification (tau = 0)
bounds_point <- calculate_lewbel_bounds(data, tau = 0)

# Set identification with bootstrap SE
bounds_set <- calculate_lewbel_bounds(data, tau = 0.2, compute_se = TRUE, B = 100)

## End(Not run)
```

---

create_default_config   *Create Default Configuration for Lewbel Monte Carlo Simulations*

Description

Creates a comprehensive configuration list with all parameters needed for running Lewbel (2012) Monte Carlo simulations.

Usage

```
create_default_config(
  num_simulations = 200,
  main_sample_size = 500,
  sample_sizes = c(250, 500, 1000, 2000),
  delta_het = 0.8,
  delta_het_values = c(0.4, 0.8, 1.2),
  n_reps_by_n = 100,
  n_reps_by_delta = 100,
  bootstrap_reps = 100,
  bootstrap_subset_size = 10,
  bootstrap_demo_size = 5,
  beta1_0 = 0.5,
  beta1_1 = 1.5,
  gamma1 = -0.8,
  beta2_0 = 1,
  beta2_1 = -1,
  alpha1 = -0.5,
  alpha2 = 1,
  tau_set_id = 0.2,
  endog_var_name = "Y2",
  exog_var_names = "Xk"
)
```

Arguments

num_simulations
>   Integer. Number of main simulation runs (default: 1000).

main_sample_size
>   Integer. Primary sample size for main results (default: 1000).

sample_sizes
>   Integer vector. Sample sizes for consistency analysis (default: c(500, 1000, 2000)).

delta_het
>   Numeric. Heteroscedasticity strength parameter (default: 1.2).

bootstrap_reps
>   Integer. Number of bootstrap replications (default: 100).

gamma1
>   Numeric. True value of the endogenous parameter (default: -0.8).

tau_set_id
>   Numeric. Tau parameter for set identification (default: 0.2).

set_seed
>   Integer. Base seed for reproducibility (default: 123).

Value

A list containing all configuration parameters.

Examples

```
## Not run:
config <- create_default_config()
config$gamma1 # -0.8

# Custom configuration
custom_config <- create_default_config(
  num_simulations = 500,
  gamma1 = -0.5
)

## End(Not run)
```

---

generate_all_plots   *Generate All Simulation Plots*

Description

Generates all visualization plots for the Lewbel simulation results.

Usage

```
generate_all_plots(
  results_main,
  results_by_n,
  results_by_delta,
  bootstrap_examples,
  config,
  verbose = TRUE
)
```

Arguments

results_main	Data.frame. Main simulation results.
results_by_n	Data.frame. Sample size analysis results.
results_by_delta	
	Data.frame. Sensitivity analysis results.
bootstrap_examples	
	Data.frame. Bootstrap examples.
config	List. Configuration object.
verbose	Logical. Whether to print plots (default: TRUE).

Value

A list of ggplot2 objects.

Examples

```
## Not run:
# Run full simulation
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_main <- run_main_simulation(config, seeds)
results_by_n <- run_sample_size_analysis(config, seeds)
results_by_delta <- run_sensitivity_analysis(config, seeds)
bootstrap_demo <- run_bootstrap_demonstration(config, seeds)
bootstrap_examples <- analyze_bootstrap_results(results_main, bootstrap_demo, config)

plots <- generate_all_plots(
  results_main, results_by_n, results_by_delta,
  bootstrap_examples, config
)

## End(Not run)
```

---

generate_all_seeds   *Generate All Seeds for Lewbel Simulation*

Description

Pre-generates all seeds needed for different parts of the Lewbel simulation to ensure reproducibility across parallel execution.

Usage

```
generate_all_seeds(config)
```

Arguments

config	List. Configuration object from create_default_config().

Value

A list containing seed vectors/matrices for different simulation parts.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
names(seeds) # "main", "by_n", "by_delta", "bootstrap_demo"

## End(Not run)
```

---

generate_lewbel_data   *Generate Data for Lewbel (2012) Triangular Model*

Description

Creates a dataset based on the triangular model with single-factor error structure that satisfies Lewbel's identifying assumptions. The data generating process uses a common factor structure for the errors to ensure the covariance restriction Cov(Z, $\epsilon_1 \epsilon_2$) = 0 is satisfied.

Usage

```
generate_lewbel_data(n, params)
```

Arguments

n   Integer. Sample size.

params   List. Parameters for the data generating process containing:

- beta1_0, beta1_1: Parameters for first equation
- beta2_0, beta2_1: Parameters for second equation
- gamma1: Endogenous parameter (key parameter of interest)
- alpha1, alpha2: Factor loadings for common factor U
- delta_het: Heteroscedasticity strength parameter

Details

The triangular model is:
$$Y_1 = \beta_{1,0} + \beta_{1,1}X + \gamma_1 Y_2 + \epsilon_1$$
$$Y_2 = \beta_{2,0} + \beta_{2,1}X + \epsilon_2$$

The error structure follows a single-factor model:

$$\epsilon_1 = \alpha_1 U + V_1$$

$$\epsilon_2 = \alpha_2 U + V_2$$

where U, V_1 are independent standard normal, and V_2 ~ N(0, exp(δ Z)) with Z = X^2 - EX^2 being the heteroscedasticity driver.

Value

A data.frame with columns Y1, Y2, Xk, Z, epsilon1, epsilon2.

Examples

```
## Not run:
params <- list(
  beta1_0 = 0.5, beta1_1 = 1.5, gamma1 = -0.8,
  beta2_0 = 1.0, beta2_1 = -1.0,
  alpha1 = -0.5, alpha2 = 1.0, delta_het = 1.2
)
data <- generate_lewbel_data(1000, params)
head(data)

## End(Not run)
```

---

generate_seed_matrix   *Generate Seed Matrix for Reproducible Parallel Simulations*

Description

Pre-generates seeds for reproducible parallel simulations to ensure consistent results across different computing environments.

Usage

```
generate_seed_matrix(base_seed, n_experiments, n_reps_each)
```

Arguments

base_seed	Integer. Base seed for random number generation.
n_experiments	Integer. Number of experiments (rows in matrix).
n_reps_each	Integer. Number of replications per experiment (columns).

Value

A matrix of seeds with dimensions n_experiments x n_reps_each.

Examples

```
## Not run:
seeds <- generate_seed_matrix(123, 3, 100)
dim(seeds) # 3 x 100

## End(Not run)
```

---

plot_bootstrap_ci   *Create Bootstrap Confidence Intervals Plot*

Description

Creates a plot showing set identification bounds with bootstrap confidence intervals.

Usage

```
plot_bootstrap_ci(bootstrap_examples, config)
```

Arguments

bootstrap_examples

                Data.frame. Bootstrap examples with standard errors.

config   List. Configuration object containing true parameter values.

Value

A ggplot2 object or NULL if insufficient data.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_main <- run_main_simulation(config, seeds)
bootstrap_demo <- run_bootstrap_demonstration(config, seeds)
bootstrap_examples <- analyze_bootstrap_results(results_main, bootstrap_demo, config)
p5 <- plot_bootstrap_confidence_intervals(bootstrap_examples, config)
if (!is.null(p5)) print(p5)

## End(Not run)
```

---

plot_estimator_distributions
                         Create Distribution Plot of Estimators

---

Description

Creates a density plot comparing the distributions of OLS and 2SLS estimators.

Usage

```
plot_estimator_distributions(results_clean, config)
```

Arguments

results_clean   Data.frame. Cleaned simulation results.

config   List. Configuration object containing true parameter values.

Value

A ggplot2 object.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results <- run_main_simulation(config, seeds)
results_clean <- na.omit(results)
p1 <- plot_estimator_distributions(results_clean, config)
print(p1)

## End(Not run)
```

---

plot_first_stage_f_dist

Create First-Stage F Distribution Plot

Description

Creates a histogram of first-stage F-statistics with weak instrument threshold.

Usage

```
plot_first_stage_f_dist(results_clean, config = NULL, weak_iv_pct = NULL)
```

Arguments

`results_clean`	Data.frame. Cleaned simulation results.
`weak_iv_pct`	Numeric. Percentage of simulations with weak instruments.

Value

A ggplot2 object.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results <- run_main_simulation(config, seeds)
results_clean <- na.omit(results)
weak_iv_pct <- mean(results_clean$first_stage_F < 10) * 100
p4 <- plot_first_stage_f_distribution(results_clean, weak_iv_pct)
print(p4)

## End(Not run)
```

---

plot_het_sensitivity   *Create Heteroscedasticity Sensitivity Plot*

Description

Creates a boxplot showing 2SLS performance by heteroscedasticity strength.

Usage

```
plot_het_sensitivity(results_by_delta, config)
```

Arguments

results_by_delta

        Data.frame. Results from sensitivity analysis.

config   List. Configuration object containing true parameter values.

Value

A ggplot2 object.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_by_delta <- run_sensitivity_analysis(config, seeds)
p3 <- plot_heteroscedasticity_sensitivity(results_by_delta, config)
print(p3)

## End(Not run)
```

---

plot_sample_size_consistency
Create Sample Size Consistency Plot

Description

Creates a boxplot showing 2SLS estimate consistency across sample sizes.

Usage

```
plot_sample_size_consistency(results_by_n, config)
```

Arguments

results_by_n   Data.frame. Results from sample size analysis.

config   List. Configuration object containing true parameter values.

Value

A ggplot2 object.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
results_by_n <- run_sample_size_analysis(config, seeds)
p2 <- plot_sample_size_consistency(results_by_n, config)
print(p2)

## End(Not run)
```

---

print_simulation_summary

Print Simulation Summary

Description

Prints a comprehensive summary of simulation findings.

Usage

```
print_simulation_summary(analysis = NULL, config = NULL, verbose = TRUE)
```

Arguments

verbose   Logical. Whether to print the summary (default: TRUE).

Examples

```
## Not run:
print_simulation_summary()

## End(Not run)
```

---

run_bootstrap_demonstration

Run Bootstrap Demonstration

Description

Runs a separate demonstration of bootstrap standard errors for set identification bounds.

Usage

```
run_bootstrap_demonstration(config, seeds, verbose = TRUE)
```

Arguments

config	List. Configuration object from create_default_config().
seeds	List. Seed object from generate_all_seeds().
verbose	Logical. Whether to print progress messages (default: TRUE).

Value

A data.frame containing bootstrap demonstration results.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
bootstrap_results <- run_bootstrap_demonstration(config, seeds)

## End(Not run)
```

---

run_lewbel_demo	*Run Quick Lewbel Monte Carlo Demo*

Description

Runs a quick demonstration of the Lewbel Monte Carlo simulation with reduced parameters for faster execution.

Usage

```
run_lewbel_demo(num_simulations = 100, verbose = TRUE)
```

Arguments

num_simulations	
	Integer. Number of simulations to run (default: 100).
verbose	Logical. Whether to print output (default: TRUE).

Value

Results from run_lewbel_monte_carlo() with reduced parameters.

Examples

```
## Not run:
# Quick demo with 50 simulations
demo_results <- run_lewbel_demo(50)

# Silent demo
demo_results <- run_lewbel_demo(100, verbose = FALSE)

## End(Not run)
```

---

run_lewbel_monte_carlo

Run Complete Lewbel (2012) Monte Carlo Simulation

Description

Executes a comprehensive Monte Carlo simulation to evaluate the performance of Lewbel's (2012) heteroscedasticity-based identification strategy. This is the main function that orchestrates all simulation components.

Usage

```
run_lewbel_monte_carlo(
  config = NULL,
  run_verification = TRUE,
  run_bootstrap_demo = TRUE,
  run_sample_analysis = TRUE,
  run_sensitivity = TRUE,
  generate_plots = TRUE,
  verbose = TRUE
)
```

Arguments

config	List. Configuration object from create_default_config(). If NULL, uses default configuration.
run_verification	
	Logical. Whether to run assumption verification (default: TRUE).
run_bootstrap_demo	
	Logical. Whether to run bootstrap demonstration (default: TRUE).
run_sample_analysis	
	Logical. Whether to run sample size analysis (default: TRUE).
run_sensitivity	
	Logical. Whether to run sensitivity analysis (default: TRUE).
generate_plots	Logical. Whether to generate visualization plots (default: TRUE).
verbose	Logical. Whether to print progress and results (default: TRUE).

Details

This function runs a complete Monte Carlo evaluation including:

- Verification of Lewbel's identifying assumptions
- Main simulation comparing OLS vs 2SLS (Lewbel) estimators
- Bootstrap demonstration for set identification bounds
- Sample size consistency analysis
- Sensitivity analysis to heteroscedasticity strength
- Comprehensive result analysis and visualization

The simulation uses parallel processing for efficiency and includes proper seed management for reproducibility.

Value

A list containing:

- config: Configuration used
- results_main: Main simulation results
- results_by_n: Sample size analysis results (if run)
- results_by_delta: Sensitivity analysis results (if run)
- bootstrap_demo: Bootstrap demonstration results (if run)
- analysis: Summary analysis
- plots: Visualization plots (if generated)

Examples

```
## Not run:
# Run with default settings
results <- run_lewbel_monte_carlo()

# Run with custom configuration
custom_config <- create_default_config(
  num_simulations = 500,
  gamma1 = -0.5,
  delta_het = 1.5
)
results <- run_lewbel_monte_carlo(custom_config)

# Run only main simulation without extras
results <- run_lewbel_monte_carlo(
  run_bootstrap_demo = FALSE,
  run_sample_analysis = FALSE,
  run_sensitivity = FALSE
)

## End(Not run)
```

---

run_main_simulation   *Run Main Lewbel Monte Carlo Simulation*

Description

Executes the main Monte Carlo simulation to evaluate the performance of Lewbel's (2012) heteroscedasticity-based identification strategy.

Usage

```
run_main_simulation(config, seeds, verbose = TRUE)
```

Arguments

config	List. Configuration object from create_default_config().
seeds	List. Seed object from generate_all_seeds().
verbose	Logical. Whether to print progress messages (default: TRUE).

Value

A data.frame containing results from all simulation runs.

Examples

```
## Not run:
config <- create_default_config(num_simulations = 100)
seeds <- generate_all_seeds(config)
results <- run_main_simulation(config, seeds)

## End(Not run)
```

---

run_sample_size_analysis

Run Sample Size Analysis

Description

Analyzes the consistency of estimators across different sample sizes.

Usage

```
run_sample_size_analysis(config, seeds, verbose = TRUE)
```

Arguments

config	List. Configuration object from create_default_config().
seeds	List. Seed object from generate_all_seeds().
verbose	Logical. Whether to print progress messages (default: TRUE).

Value

A data.frame containing results for different sample sizes.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
size_results <- run_sample_size_analysis(config, seeds)

## End(Not run)
```

```
run_sensitivity_analysis
```
Run Sensitivity Analysis

Description

Analyzes sensitivity of results to heteroscedasticity strength.

Usage

```
run_sensitivity_analysis(config, seeds, verbose = TRUE)
```

Arguments

config	List. Configuration object from create_default_config().
seeds	List. Seed object from generate_all_seeds().
verbose	Logical. Whether to print progress messages (default: TRUE).

Value

A data.frame containing results for different heteroscedasticity parameters.

Examples

```
## Not run:
config <- create_default_config()
seeds <- generate_all_seeds(config)
sensitivity_results <- run_sensitivity_analysis(config, seeds)

## End(Not run)
```

```
run_single_lewbel_simulation
```
Run Single Lewbel Simulation

Description

Executes a single Monte Carlo simulation run comparing OLS, 2SLS (Lewbel), and set identification approaches for estimating the endogenous parameter.

Usage

```
run_single_lewbel_simulation(
  sim_id,
  params,
  endog_var = "Y2",
  exog_vars = "Xk",
  compute_bounds_se = FALSE
)
```

Arguments

    `sim_id`   Integer. Simulation run identifier.

    `params`   List. Parameters for data generation and estimation.

    `endog_var`   Character. Name of endogenous variable (default: "Y2").

    `exog_vars`   Character vector. Names of exogenous variables (default: "Xk").

    `compute_bounds_se`

                Logical. Whether to compute bootstrap SE for bounds (default: FALSE).

Value

A data.frame with one row containing simulation results including: OLS and 2SLS estimates, coverage indicators, first-stage F-statistic, and identification bounds.

Examples

```
## Not run:
config <- create_default_config()
params <- list(
  sample_size = config$main_sample_size,
  beta1_0 = config$beta1_0, beta1_1 = config$beta1_1, gamma1 = config$gamma1,
  beta2_0 = config$beta2_0, beta2_1 = config$beta2_1,
  alpha1 = config$alpha1, alpha2 = config$alpha2,
  delta_het = config$delta_het, tau_set_id = config$tau_set_id,
  bootstrap_reps = config$bootstrap_reps
)
result <- run_single_lewbel_simulation(1, params)

## End(Not run)
```

`verify_lewbel_assumptions`

Verify Lewbel's Key Identifying Assumptions

Description

Tests whether the data generating process satisfies the key assumptions required for Lewbel's (2012) identification strategy. This includes testing the covariance restriction and instrument relevance condition.

Usage

```
verify_lewbel_assumptions(
  data = NULL,
  config = NULL,
  n = 10000,
  params = NULL,
  verbose = TRUE
)
```

Arguments

`data`	Data.frame. Optional. Pre-generated data to verify (alternative to n/params).
`config`	List. Optional. Configuration object (used with data parameter).
`n`	Integer. Sample size for verification (default: 10000, used with params).
`params`	List. Parameters for the DGP (same format as generate_lewbel_data).
`verbose`	Logical. Whether to print detailed output (default: TRUE).

Details

The function tests:

- Assumption A2: $\text{Cov}(Z, \epsilon_1 \epsilon_2) = 0$ (covariance restriction)
- Assumption A3: $\text{Cov}(Z, \epsilon_2^2) \mathrel{!=} 0$ (instrument relevance)
- Endogeneity: $\text{Cov}(\epsilon_1, \epsilon_2) \mathrel{!=} 0$

Can be called in two ways:

1. verify_lewbel_assumptions(data, config) - using pre-generated data
2. verify_lewbel_assumptions(n = 10000, params = params) - generating new data

Value

Invisibly returns a list with verification results and data.

Examples

```
## Not run:
config <- create_default_config()
params <- list(
  beta1_0 = config$beta1_0, beta1_1 = config$beta1_1, gamma1 = config$gamma1,
  beta2_0 = config$beta2_0, beta2_1 = config$beta2_1,
  alpha1 = config$alpha1, alpha2 = config$alpha2, delta_het = config$delta_het
)
verify_lewbel_assumptions(params = params)

## End(Not run)
```

Index