# Package: fixedCV (via r-universe)

June 30, 2026

**Type** Package

**Title** Fixed-b Critical Values for Robust Inference with Time Series Data

**Version** 0.1.0

**Maintainer** Rebecca Kurtz-Garcia <rkurtzgarcia@smith.edu>

**Description** Provides functions for computing fixed-b critical values and conducting robust inference procedures for time series data with unknown correlation structures. Implements long-run variance estimators using various kernel functions and lugsail transformations for improved finite-sample properties as described by Kurtz-Garcia and Flegal (2026) <doi:10.48550/arXiv.2606.17369>.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.5)

**Imports** Matrix

**Suggests** aTSA, tseries, dplyr, lubridate, lmtest

**Config/roxygen2/version** 8.0.0

**NeedsCompilation** no

**Author** Rebecca Kurtz-Garcia [aut, cre], Thomas Robacker [aut]

**Repository** https://cran.r-universe.dev

**Date/Publication** 2026-06-30 19:00:02 UTC

**RemoteUrl** https://github.com/cran/fixedCV

**RemoteRef** HEAD

**RemoteSha** ce09f4c0a15d02ab86fbd14c5b2a04c73d21b59c

# Contents

fitted_CV . . . . . 2
generate_cv . . . . . 3
get_b . . . . . 4
get_cv . . . . . 6
GillStream . . . . . 7
kernels . . . . . 7
robust_lm . . . . . 8
simulated_CV . . . . . 10

**Index** **13**

---

`fitted_CV` *Fixed Critical Values using the Fitted Method*

---

## Description

A look up table containing the fitted coefficients for robust critical values. It also contains the adjusted $R^2$ values for the respective fit.

## Usage

```
fitted_CV
```

## Format

A data frame with the following columns:

- intercept:a parameter for the approximation
- beta1: a parameter for the approximation
- beta2: a parameter for the approximation
- beta3: a parameter for the approximation
- adj.r.sq: Adjusted $R^2$
- d: dimensions
- kernel: mother kernel selected
- lugsail: lugsail setting
- alpha: objective type 1 error rate

## Details

This data set contains critical values that the `get_cv` and `get_cv_fitted` functions search for. For a given kernel, lugsail setting, dimension, and desired $\alpha$, the critical value is approximated via the formula, $CV = \text{intercept} + \beta_1 * b + \beta_2 * b^2 + \beta_3 * b^3$

`generate_cv` *Generate Fixed-b Critical Values*

## Description

Generates fixed-b critical values for robust hypothesis testing. Simulate from a fixed-b distribution to get custom critical values.

## Usage

```
generate_cv(
  b,
  d = 2,
  alpha = 0.05,
  the_kernel = bartlett,
  lugsail_type = "Mother",
  q = 1,
  return_F_stats = FALSE,
  num_replicates = 50000,
  replicate_size = 1000
)
```

## Arguments

| | |
|---|---|
| `b` | Numeric vector of bandwidth values between 0 and 1. The bandwidth represents the proportion of autocovariances given non-zero weight. |
| `d` | Integer, the dimension (degrees of freedom) of the test statistic. Default is 1. |
| `alpha` | Numeric vector of desired significance levels between 0 and 1. Default is 0.05. |
| `the_kernel` | A function for the kernel used to generate the weights. The package supports Bartlett (`bartlett`), Parzen (`parzen`), Tukey-Hanning (`th`), or Quadratic Spectral `qs`. User supplied functions are also supported. |
| `lugsail_type` | Character string specifying the lugsail transformation. Options are `"Mother"` (default), `"Zero"`, or `"Over"`. |
| `q` | The Parzen characteristic exponent of the kernel supplied. The default is 1. |
| `return_F_stats` | A logical value indicating if test statistics used to approximate the fixed-b distribution should be returned. The default value is FALSE. |
| `num_replicates` | A integer indicating of test statistics to be generated used to approximate the fixed-b distribution. Default is 50,000. |
| `replicate_size` | A integer indicating the sample size used to construct the test statistics that are used to approximate the fixed-b distribution. Default is 1000. |

## Details

This function is used to generate custom simulated critical values. Both package kernels and user supplied kernels maybe supplied. It is not recommended to decrease `num_replicates` or `replicate_size`, especially for higher dimensions.

### Value

If `return_F_stats = FALSE` (default) a matrix is returned where the rows correspond to the values for `b` and the columns correspond to the significance levels (`alpha`) supplied by the user. If `return_F_stats = TRUE`, then a list is returned. The second element in the list is a matrix of the F statistics generated to approximate the fixed-b distribution.

### Examples

```
# Multiple b's and alpha's
set.seed(62)
CVs <- generate_cv(b = c(0.005, 0.006), d=1, alpha = c(0.05, 0.01),
num_replicates = 100)
CVs
# Custom kernel function
my_kernel <- function(x){
  k_x <- exp(-x^2/2)
  return(k_x)
}
CVs <- generate_cv(b = 0.005, d=1, alpha = 0.05, the_kernel = my_kernel,
 q = 2, num_replicates = 100)
CVs
```

---

## get_b *Automatic Bandwidth Selection*

---

### Description

Selects optimal bandwidth for long-run variance estimation by controlling Type I error distortion. The bandwidth determines what proportion of autocovariances receive non-zero weight in the spectral variance estimator.

### Usage

```
get_b(
  the_data,
  alpha = 0.05,
  the_kernel = "Bartlett",
  lugsail = "Mother",
  tau = NA,
  auto_adjust = TRUE
)
```

### Arguments

| | |
|---|---|
| `the_data` | Numeric vector, matrix, or data frame containing the data (typically residuals or moment conditions from a model). For matrices, each column represents a different series. |
| `alpha` | Numeric significance level for hypothesis testing. Default is 0.05. |

| | |
|---|---|
| the_kernel | Character string specifying the kernel function. Options are "Bartlett" (default), "Parzen", "TH", or "QS". |
| lugsail | Character string specifying the lugsail transformation. Options are "Mother" (default), "Zero", or "Over". |
| tau | Numeric tolerance level for acceptable Type I error distortion. If NA (default), uses recommended values: -alpha^(0.5*d)/(T*log(rho)) for Zero lugsail, alpha*0.15 for others. |
| auto_adjust | Logical indicating whether to automatically increase tolerance if no suitable bandwidth is found. Default is TRUE. |

## Details

Returns a bandwidth optimized for hypothesis testing. If no hypothesis test is being conducted, using alpha = 0.05 as a guideline is recommended. The selected bandwidth is the smallest value whose Type I error rate falls within tau of alpha. If no bandwidth yields a Type I error rate within tau of alpha and auto_adjust = TRUE, then tau is increased by 25% iteratively until a bandwidth satisfies the criterion.

When lugsail = "Zero", the default setting tau = NA uses the tolerance recommended by Kurtz-Garcia and Flegal (2026). For other lugsail settings, larger tolerance levels are typically needed, hence when tau = NA, the default is set to alpha * 0.15.

## Value

Numeric scalar, the selected optimal bandwidth value between 0 and 1.

## References

Rebecca P. Kurtz-Garcia and James M. Flegal. Inference optimal long run variance estimation with lugsail kernels, 2026. URL https://arxiv.org/abs/2606.17369.

## Examples

```
# Simulate AR(1) data
set.seed(123)
data <- arima.sim(list(ar = 0.7), n = 100)

# Get optimal bandwidth
get_b(data)

# With different kernel
get_b(data, the_kernel = "QS")

# With custom tolerance
get_b(data, tau = 0.01)
```

## get_cv — *Get Fixed-b Critical Values*

### Description

Retrieves fixed-b critical values for robust hypothesis testing. Critical values can be computed using simulated lookup tables, fitted polynomial approximations, or analytical formulas.

### Usage

```
get_cv(
  new_b,
  d = 1,
  alpha = 0.05,
  the_kernel = "Bartlett",
  lugsail = "Mother",
  method = "simulated"
)
```

### Arguments

| | |
|---|---|
| `new_b` | Numeric vector of bandwidth values between 0 and 1. The bandwidth represents the proportion of autocovariances given non-zero weight. |
| `d` | Integer, the dimension (degrees of freedom) of the test statistic. Default is 1 for t-tests. |
| `alpha` | Numeric significance level for the test. Common values are 0.01, 0.025, 0.05, or 0.10. Default is 0.05. |
| `the_kernel` | Character string specifying the kernel function. Options are `"Bartlett"` (default), `"Parzen"`, `"TH"`, or `"QS"`. |
| `lugsail` | Character string specifying the lugsail transformation. Options are `"Mother"` (default), `"Zero"`, or `"Over"`. |
| `method` | Character string specifying computation method. Options are `"simulated"` (default, uses pre-computed tables), `"fitted"` (polynomial approximation), `"analytical"` (closed-form formulas), `"analytical"` (closed-form formulas), or `"adaptive"` (classical limiting distributions). . |

### Value

Numeric vector of critical values corresponding to each bandwidth in `new_b`. When `b = 0`, returns chi-square critical values.

### Examples

```
# Get critical value for single bandwidth
get_cv(0.1, d = 1, alpha = 0.05)
```

```
# Get critical values for multiple bandwidths
get_cv(c(0, 0.1, 0.2, 0.3), d = 1, alpha = 0.05)

# Using different methods
get_cv(0.1, method = "fitted")
get_cv(0.1, method = "analytical")

# For F-test with 3 degrees of freedom
get_cv(0.1, d = 3, alpha = 0.05)
```

---

## GillStream — *Stream Temperature data*

---

### Description

This data set contains daily average temperature readings in Celsius for a stream in Gill, Massachusetts. The original data was recorded via a submerged logger at 15 minutes intervals from August to December 2024 by the Organismic & Evolutionary Biology Department at the University of Massachusetts Amherst.

### Usage

```
GillStream
```

### Format

A data frame with 153 rows and 3 columns:

**Date** The date the temperature was recorded.

**AvgTemp** Temperature in Celsius

**time_index** index values for the date

---

## kernels — *Kernel Functions for Long-Run Variance Estimation*

---

### Description

Kernel weight functions used in spectral variance (long-run variance) estimation. These functions generate weights for autocovariances in robust inference procedures.

### Usage

```
bartlett(x)

qs(x)

parzen(x)

th(x)
```

### Arguments

| | |
|---|---|
| x | Numeric value, typically in the range [0, 1], representing the scaled lag. |

### Details

The kernel functions are:

- bartlett: Bartlett (Newey-West) kernel. q = 1 kernel.
- parzen: Parzen kernel. q = 2 kernel.
- th: Tukey-Hanning kernel. q = 2 kernel.
- qs: Quadratic Spectral kernel. q = 2 kernel.

### Value

Numeric kernel weight value.

### Examples

```
# Evaluate kernels at x = 0.5
bartlett(0.5)
parzen(0.5)
th(0.5)
qs(0.5)

# Plot kernel shapes
x <- seq(-1.5, 1.5, length.out = 200)
plot(x, sapply(x, bartlett), type = "l", ylab = "Weight", main = "Bartlett Kernel")
```

---

## robust_lm — *Robust Linear Model Inference with Fixed-b Critical Values*

---

### Description

Conducts robust inference for linear models using fixed-b asymptotics. This function computes robust standard errors, test statistics, and p-values for linear model coefficients, accounting for unknown serial correlation in the errors.

### Usage

```
robust_lm(
  fit,
  the_kernel = "Bartlett",
  lugsail = "Mother",
  method = "simulated",
  tau = NA,
  alpha = 0.05,
  conf.int = FALSE
)
```

### Arguments

| | |
|---|---|
| `fit` | An `lm` object from a linear regression model. |
| `the_kernel` | Character string specifying the kernel function. Options are `"Bartlett"` (default), `"Parzen"`, `"TH"` (Tukey-Hanning), or `"QS"` (Quadratic Spectral). |
| `lugsail` | Character string specifying the lugsail transformation. Options are `"Mother"` (default), `"Zero"`, or `"Over"`. |
| `method` | Character string specifying how critical values are computed. Options are `"simulated"` (default), `"fitted"`, `"analytical"`, or `"adaptive"`. |
| `tau` | Numeric tolerance level for bandwidth selection. If `NA` (default), uses recommended values based on lugsail type. |
| `alpha` | Numeric significance level for hypothesis tests (default is 0.05). |
| `conf.int` | Logical indicating whether to compute confidence intervals (default is FALSE). If TRUE and alpha is one of 0.10, 0.05, 0.025, or 0.01, confidence intervals are added. |

### Details

This function largely follows the robust inference procedure described in Kurtz-Garcia and Flegal (2026). Standard errors are estimated using a HAC estimator with a choice of kernel functions. Both classical adaptive limiting theory `` `method = "adaptive"` `` and fixed limiting theory are supported, with multiple approximations available for the fixed-limiting theory.

The `method = "simulated"` option is only available for problems with up to 12 dimensions. In general, `method = "analytical"` is recommended for F-tests and for non-univariate tests.

By default, the null hypothesis for the F-test is that all slope coefficients are zero. If the model contains no slope coefficients, the null hypothesis is that the intercept is zero.

Because obtaining quantiles for the fixed-limit distribution is computationally expensive, exact p-values are not reported. Instead, thresholds are given indicating if the p-value is above 0.10 (>= 0.10), between 0.10 and 0.05 (<0.10), between 0.05 and 0.025 (<0.05.), between 0.025 and 0.01 (<0.025.), and less than 0.01 (<0.01*). For the same reason, confidence intervals are only available for a limited set of confidence levels.

### Value

A list containing:

| | |
|---|---|
| `Summary_Table` | Data frame with coefficient estimates, robust standard errors, t-statistics, and p-values. |
| `F_test` | Data frame with F-statistic and p-value for joint significance test. |
| `CV_table` | Data frame showing bandwidth (b), dimension, and critical values at different significance levels for each coefficient and the F-test. |
| `vcov` | The variance covariance matrix of the regression coefficients. |

### References

Rebecca P. Kurtz-Garcia and James M. Flegal. Inference optimal long run variance estimation with lugsail kernels, 2026. URL https://arxiv.org/abs/2606.17369.

## Examples

```
# Simulate AR(1) data
set.seed(123)
n <- 100
x <- arima.sim(list(ar = 0.5), n)
y <- 2 + 3*x + arima.sim(list(ar = 0.7), n)
fit <- lm(y ~ x)

# Robust inference with default settings
robust_lm(fit)

# With different kernel and method
robust_lm(fit, the_kernel = "QS", method = "fitted")

# With confidence intervals
robust_lm(fit, conf.int = TRUE)
```

---

# simulated_CV *Fixed Critical Values using the Simulated Method*

---

## Description

The title of each data set is the `<Mother>_<Lugsail>_<alpha>_Master`.

## Usage

```
Bartlett_Mother_01_Master

Bartlett_Mother_025_Master

Bartlett_Mother_05_Master

Bartlett_Mother_10_Master

Bartlett_Zero_01_Master

Bartlett_Zero_025_Master

Bartlett_Zero_05_Master

Bartlett_Zero_10_Master

Bartlett_Over_01_Master

Bartlett_Over_025_Master

Bartlett_Over_05_Master
```

Bartlett_Over_10_Master

QS_Mother_01_Master

QS_Mother_025_Master

QS_Mother_05_Master

QS_Mother_10_Master

QS_Zero_01_Master

QS_Zero_025_Master

QS_Zero_05_Master

QS_Zero_10_Master

QS_Over_01_Master

QS_Over_025_Master

QS_Over_05_Master

QS_Over_10_Master

Parzen_Mother_01_Master

Parzen_Mother_025_Master

Parzen_Mother_05_Master

Parzen_Mother_10_Master

Parzen_Zero_01_Master

Parzen_Zero_025_Master

Parzen_Zero_05_Master

Parzen_Zero_10_Master

Parzen_Over_01_Master

Parzen_Over_025_Master

Parzen_Over_05_Master

`Parzen_Over_10_Master`

`TH_Mother_01_Master`

`TH_Mother_025_Master`

`TH_Mother_05_Master`

`TH_Mother_10_Master`

`TH_Zero_01_Master`

`TH_Zero_025_Master`

`TH_Zero_05_Master`

`TH_Zero_10_Master`

`TH_Over_01_Master`

`TH_Over_025_Master`

`TH_Over_05_Master`

`TH_Over_10_Master`

## Format

A look up table containing the simulated robust critical values. The 'b' column contains the bandwidth, and subsequent columns are the dimensions

- b: bandwidth of the test statistics. The proportion of autocovariances given a non-zero weight.
- d: the dimension of the test statistic.

## Details

The title of each data set is the `<Mother>_<Lugsail>_<alpha>_Master`. Supported mother kernels are Bartlett, Parzen, Tukey-Hanning (TH), and Quadratic Spectral (QS). Supported lugsail settings are mother, zero, and over. The available critical values are 0.01, 0.025, 0.05, and 0.10. Note that when b = 0 the robust critical value are equivalent to chi-square critical values.

# Index


∗ **datasets**
fitted_CV, 2
GillStream, 7
simulated_CV, 10

bartlett *(kernels)*, 7
Bartlett_Mother_01_Master *(simulated_CV)*, 10
Bartlett_Mother_025_Master *(simulated_CV)*, 10
Bartlett_Mother_05_Master *(simulated_CV)*, 10
Bartlett_Mother_10_Master *(simulated_CV)*, 10
Bartlett_Over_01_Master *(simulated_CV)*, 10
Bartlett_Over_025_Master *(simulated_CV)*, 10
Bartlett_Over_05_Master *(simulated_CV)*, 10
Bartlett_Over_10_Master *(simulated_CV)*, 10
Bartlett_Zero_01_Master *(simulated_CV)*, 10
Bartlett_Zero_025_Master *(simulated_CV)*, 10
Bartlett_Zero_05_Master *(simulated_CV)*, 10
Bartlett_Zero_10_Master *(simulated_CV)*, 10

fitted_CV, 2

generate_cv, 3
get_b, 4
get_cv, 6
GillStream, 7

kernels, 7

parzen *(kernels)*, 7
Parzen_Mother_01_Master *(simulated_CV)*, 10
Parzen_Mother_025_Master *(simulated_CV)*, 10
Parzen_Mother_05_Master *(simulated_CV)*, 10
Parzen_Mother_10_Master *(simulated_CV)*, 10
Parzen_Over_01_Master *(simulated_CV)*, 10
Parzen_Over_025_Master *(simulated_CV)*, 10
Parzen_Over_05_Master *(simulated_CV)*, 10
Parzen_Over_10_Master *(simulated_CV)*, 10
Parzen_Zero_01_Master *(simulated_CV)*, 10
Parzen_Zero_025_Master *(simulated_CV)*, 10
Parzen_Zero_05_Master *(simulated_CV)*, 10
Parzen_Zero_10_Master *(simulated_CV)*, 10

qs *(kernels)*, 7
QS_Mother_01_Master *(simulated_CV)*, 10
QS_Mother_025_Master *(simulated_CV)*, 10
QS_Mother_05_Master *(simulated_CV)*, 10
QS_Mother_10_Master *(simulated_CV)*, 10
QS_Over_01_Master *(simulated_CV)*, 10
QS_Over_025_Master *(simulated_CV)*, 10
QS_Over_05_Master *(simulated_CV)*, 10
QS_Over_10_Master *(simulated_CV)*, 10
QS_Zero_01_Master *(simulated_CV)*, 10
QS_Zero_025_Master *(simulated_CV)*, 10
QS_Zero_05_Master *(simulated_CV)*, 10
QS_Zero_10_Master *(simulated_CV)*, 10

robust_lm, 8

simulated_CV, 10

th *(kernels)*, 7
TH_Mother_01_Master *(simulated_CV)*, 10
TH_Mother_025_Master *(simulated_CV)*, 10

TH_Mother_05_Master *(*simulated_CV*)*, 10
TH_Mother_10_Master *(*simulated_CV*)*, 10
TH_Over_01_Master *(*simulated_CV*)*, 10
TH_Over_025_Master *(*simulated_CV*)*, 10
TH_Over_05_Master *(*simulated_CV*)*, 10
TH_Over_10_Master *(*simulated_CV*)*, 10
TH_Zero_01_Master *(*simulated_CV*)*, 10
TH_Zero_025_Master *(*simulated_CV*)*, 10
TH_Zero_05_Master *(*simulated_CV*)*, 10
TH_Zero_10_Master *(*simulated_CV*)*, 10