Package 'gkrreg'

June 17, 2026

Type Package

Title Gaussian Kernel Robust Regression (GKRReg)

Version 0.4.0

Description Implements the Gaussian Kernel Robust Regression (GKRReg / GKRR) method proposed by De Carvalho, Lima Neto and Ferreira (2017) <doi:10.1016/j.neucom.2016.12.035>. The method re-weights observations iteratively using the Gaussian kernel so that poorly-fitted observations (outliers, leverage points) receive small weights, yielding resistance to Y-space outliers, X-space outliers and leverage points. Convergence is guaranteed by Propositions 4.1 and 4.2 of the original paper. Three estimators for the kernel width hyper-parameter are provided (S1: Caputo, S2: pairwise median, S3: residual variance). Inference is provided via an analytic sandwich variance estimator (default) or via bootstrap (percentile, normal and BCa intervals with p-values) through gkrr_boot(). Six real datasets from the robust regression literature are included to facilitate reproducible comparisons.

License GPL-3

Encoding UTF-8

LazyData true

LazyDataCompression xz

Depends R (>= 4.0.0)

Imports stats, graphics, grDevices, MASS, sm

Suggests robustbase, quantreg, testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

URL https://github.com/marcelorpf/gkrreg

BugReports https://github.com/marcelorpf/gkrreg/issues

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Eufrásio de Andrade Lima Neto [aut],
Marcelo Rodrigo Portela Ferreira [aut, cre]

Maintainer Marcelo Rodrigo Portela Ferreira <marcelo@de.ufpb.br>

Repository CRAN

Date/Publication 2026-06-17 13:40:02 UTC

Contents

`belgium_calls` *International Telephone Calls from Belgium (1950–1973)*

Description

Number of international telephone calls made from Belgium between 1950 and 1973, taken from the Belgian Statistical Survey published by the Ministry of Economy. The data contain a block of Y-space outliers (observations 15–20) caused by a recording error in the original source: calls were recorded in total minutes instead of number of calls for those years.

Usage

```
belgium_calls
```

Format

A data frame with 24 rows and 2 columns:

year Last two digits of the calendar year (50 = 1950, . . . , 73 = 1973).

calls Number of international calls originating from Belgium, in tens of millions.

Details

This dataset is used in De Carvalho et al. (2017), Section 6.2, to illustrate the robustness of GKRR to Y-space outliers. The recommended width estimator for this scenario is `sigma_method = "s3"`.

Outlier structure

Observations 15–20 (years 1964–1969) are Y-space outliers. Their recorded values are anomalously large relative to the linear trend of the remaining years because calls were measured in total minutes rather than number of calls.

Source

P.J. Rousseeuw and A.M. Leroy (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons, Table 6, p. 26.

Available in **robustbase** as `telef`.

References

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). A robust regression method based on exponential-type kernel functions. *Neurocomputing*, 234, 58–74. doi:10.1016/j.neucom.2016.12.035

Examples

```
data(belgium_calls)

# Fit competing methods and compare
fit_ols  <- lm(calls ~ year, data = belgium_calls)
fit_gkrr <- gkrr(calls ~ year, data = belgium_calls, sigma_method = "s3")

plot(belgium_calls$year + 1900, belgium_calls$calls,
     xlab = "Year", ylab = "Calls (tens of millions)",
     main = "Belgium International Calls",
     pch = 19, col = "grey60")
abline(fit_ols,  col = "black",     lwd = 2, lty = 2)
abline(fit_gkrr, col = "firebrick", lwd = 2)
legend("topleft", c("OLS", "GKRR"), col = c("black","firebrick"),
       lty = c(2,1), lwd = 2, bty = "n")
```

`cloud_point` *Cloud Point of a Liquid*

Description

Measurements on the cloud point of a liquid, which is a measure of the degree of crystallization in a stock. The dataset contains three leverage points (baseline measurements at `percentage_i8 = 0`) that pull poorly-specified models away from the underlying trend.

Usage

```
cloud_point
```

Format

A data frame with 19 rows and 2 columns:

percentage_i8 Percentage of I-8 in the liquid mixture (0–10).

cloud_point Cloud point temperature of the liquid (degrees Celsius).

Details

This dataset is used in De Carvalho et al. (2017), Section 6.3, to illustrate robustness to leverage points. The recommended width estimator is `sigma_method = "s3"`.

Outlier structure

Observations 1, 10 and 16 have `percentage_i8 = 0` and act as leverage points because they correspond to a baseline condition that is far from the centroid of the predictor space.

Source

N.R. Draper and R. Craig Smith (1998). *Applied Regression Analysis*, 3rd edition. John Wiley & Sons, Exercise 4, p. 629.

Available in **robustbase** as `cloud`.

References

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). doi:10.1016/j.neucom.2016.12.035

Examples

```
data(cloud_point)

fit_ols  <- lm(cloud_point ~ percentage_i8, data = cloud_point)
fit_gkrr <- gkrr(cloud_point ~ percentage_i8, data = cloud_point,
                 sigma_method = "s3")

plot(cloud_point$percentage_i8, cloud_point$cloud_point,
     xlab = "Percentage I-8", ylab = "Cloud point (C)",
     main = "Cloud Point Data",
     pch = 19, col = "grey60")
abline(fit_ols,  col = "black",     lwd = 2, lty = 2)
abline(fit_gkrr, col = "firebrick", lwd = 2)
legend("topleft", c("OLS", "GKRR"), col = c("black","firebrick"),
       lty = c(2,1), lwd = 2, bty = "n")
```

`delivery` *Soft-Drink Delivery Time Data*

Description

Data on the time required by a route driver to service vending machines at 25 stops. Two predictors (number of products delivered and distance travelled) are used to model delivery time. Observation 9 is a bad leverage point: it has both a very large number of products (30) and a very large distance (1460 feet), pulling regression fits away from the main trend.

Usage

```
delivery
```

Format

A data frame with 25 rows and 3 columns:

n_products Number of products stocked (cases).

distance Distance walked by the driver (feet).

delivery_time Total service time at the stop (minutes).

Details

This dataset is used in De Carvalho et al. (2017), Section 6.4, to illustrate robustness to leverage points in multiple regression. The recommended width estimator is `sigma_method = "s3"`.

Outlier structure

Observation 9 (`n_products = 30`, `distance = 1460`) is a leverage point that simultaneously deviates in the predictor space and has an unusually large response, making it a bad leverage point.

Source

D.C. Montgomery and E.A. Peck (1992). *Introduction to Linear Regression Analysis*, 2nd edition. John Wiley & Sons, Table B.7.

Available in **robustbase** as `delivery`.

References

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). doi:10.1016/j.neucom.2016.12.035

Examples

```
data(delivery)

fit_ols  <- lm(delivery_time ~ n_products + distance, data = delivery)
fit_gkrr <- gkrr(delivery_time ~ n_products + distance, data = delivery,
                 sigma_method = "s3")

# Compare coefficients
rbind(OLS  = coef(fit_ols),
      GKRR = coef(fit_gkrr))

# Diagnostic plot: kernel weights highlight observation 9
plot(fit_gkrr, which = 4, ask = FALSE)
```

---

gkrr *Gaussian Kernel Robust Regression (GKRReg)*

Description

Fits a robust linear regression model using the Gaussian kernel to down-weight poorly fitted observations (outliers and leverage points). Weights $k_{ii} = G(y_i, \hat{\mu}_i) \in (0, 1]$ are updated at each iteration of an IRWLS until the objective function $S(\beta) = 2\sum_{i=1}^{n}[1 - G(y_i, \hat{\mu}_i)]$ converges.

Usage

```
gkrr(
  formula,
  data = NULL,
  sigma_method = c("s1", "s2", "s3", "s4", "auto"),
  alpha = 0.5,
  tol = 1e-10,
  maxit = 100L,
  boot = FALSE,
  boot_args = list(),
  auto_args = list(),
  conf = 0.95
)
```

Arguments

formula	Model formula, e.g. y ~ x1 + x2.
data	Optional data.frame containing the model variables.
sigma_method	Estimator for γ^2: "s1" (Caputo sub-sample quantiles), "s2" (pairwise median), "s3" (residual variance), "s4" (AICc bandwidth via sm::h.select(), recommended for $n \geq 200$), or "auto" (selects among S1–S4 by out-of-bag bootstrap MSE; incurs additional computational cost — see auto_args). A single positive numeric value fixes γ^2 directly.

`alpha` Fraction of the sample used in estimator S1 (default `0.5`).

`tol` Convergence tolerance (default `1e-10`).

`maxit` Maximum number of iterations (default `100`).

`boot` Logical or a `"gkrr_boot"` object.

- `FALSE` (default): sandwich standard errors, confidence intervals and Wald p-values are computed analytically and stored in `fit$sandwich`. `summary()` uses them automatically.
- `TRUE`: runs `gkrr_boot()` with default arguments ($B = 999$, `type = "bca"`) and stores the result inside the fitted object. Bootstrap inference takes precedence over the sandwich in `summary()`.
- A pre-computed `"gkrr_boot"` object: stored directly, avoiding redundant computation.

`boot_args` Named list of extra arguments passed to `gkrr_boot` when `boot = TRUE` (e.g. `list(B = 499, type = "percentile", seed = 1)`).

`auto_args` Named list of arguments controlling the `"auto"` selection bootstrap. Recognised elements: `B` (number of replicates, default `99`) and `seed` (integer seed for reproducibility). Ignored when `sigma_method != "auto"`.

`conf` Confidence level for sandwich intervals (default `0.95`; ignored when `boot != FALSE`).

Details

Convergence is guaranteed by Propositions 4.1 and 4.2 of De Carvalho et al. (2017).

When `boot = TRUE` a `gkrr_boot` object is computed and stored inside the fitted model, making it available automatically to `summary.gkrr` for inference (confidence intervals and bootstrap p-values).

Value

An object of class `"gkrr"` containing:

`coefficients` Named vector of estimated coefficients.

`fitted.values` Fitted values $\hat{\mu}$.

`residuals` Residuals $y - \hat{\mu}$.

`weights` Final kernel weights k_{ii}.

`gamma2` Value of $\hat{\gamma}^2$ used.

`criterion` Sequence of S values across iterations.

`iterations` Number of iterations performed.

`converged` Logical; was convergence achieved?

`sigma_method` Label of the γ^2 estimator.

`sandwich` A list with sandwich inference components: `vcov`, `se`, `tval`, `pval`, `ci_lo`, `ci_hi`, `conf`. Always computed.

`boot` A `"gkrr_boot"` object if `boot != FALSE`, otherwise `NULL`.

`call`, `terms`, `model` Model metadata.

References

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). A robust regression method based on exponential-type kernel functions. *Neurocomputing*, 234, 58–74. doi:10.1016/j.neucom.2016.12.035

See Also

gkrr_boot for bootstrap inference, summary.gkrr for the inference table.

Examples

```
set.seed(1)
n  <- 80
x  <- runif(n, 0, 10)
y  <- 2 + 3 * x + rnorm(n)
y[1:12] <- y[1:12] + 25      # 15% Y-space outliers

# Basic fit with sandwich inference (default)
fit <- gkrr(y ~ x, sigma_method = "s3")
summary(fit)

# Using S4 estimator (AICc bandwidth)

fit_s4 <- gkrr(y ~ x, sigma_method = "s4")
summary(fit_s4)


# Automatic estimator selection

fit_auto <- gkrr(y ~ x, sigma_method = "auto",
                 auto_args = list(B = 49, seed = 1))
summary(fit_auto)


# Fit with bootstrap inference (BCa, B = 999 by default)

fit_b <- gkrr(y ~ x, sigma_method = "s3", boot = TRUE,
              boot_args = list(B = 499, seed = 1))
summary(fit_b)
plot(fit_b)
```

---

gkrr_boot *Bootstrap inference for GKRReg*

Description

Runs a pairs bootstrap to estimate standard errors, confidence intervals and (optionally) p-values for the coefficients of a gkrr model. The full fitting algorithm is re-executed on every replicate, including re-estimation of γ^2.

Usage

```
gkrr_boot(
  object,
  B = 999L,
  type = c("bca", "percentile", "normal"),
  conf = 0.95,
  seed = NULL,
  verbose = FALSE
)
```

Arguments

`object`	A `"gkrr"` object (output of `gkrr()`).
`B`	Number of bootstrap replicates (default `999`).
`type`	CI type: `"bca"` (default, BCa), `"percentile"` or `"normal"`.
`conf`	Confidence level, scalar in (0, 1) (default `0.95`).
`seed`	Integer seed for reproducibility (optional).
`verbose`	Logical. If `TRUE`, prints progress every 100 replicates. Default `FALSE`.

Value

An object of class `"gkrr_boot"` containing:

`t0` Original coefficient estimates (length $p + 1$).

`t` Matrix $B_{\text{ok}} \times (p + 1)$ of bootstrap estimates.

`se` Bootstrap standard errors.

`bias` Bootstrap bias $E^*[\hat{\beta}^*] - \hat{\beta}$.

`ci` Matrix $2 \times (p + 1)$ of CIs with rows `"lower"` and `"upper"`.

`pval` Bootstrap p-values (centred-t, two-sided).

`conf` Confidence level used.

`type` CI type used.

`B` Number of successful replicates.

`B_failed` Replicates discarded due to non-convergence.

`call` The matched call.

References

Efron, B. (1987). Better bootstrap confidence intervals. *Journal of the American Statistical Association*, 82(397), 171–185.

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). doi:10.1016/j.neucom.2016.12.035

See Also

`gkrr`, `plot.gkrr_boot`

Examples

```
set.seed(42)
n  <- 80
x  <- runif(n, 0, 10)
y  <- 2 + 3 * x + rnorm(n)
y[1:12] <- y[1:12] + 25      # 15% Y-space outliers

fit  <- gkrr(y ~ x, sigma_method = "s3")
boot <- gkrr_boot(fit, B = 499, seed = 1)
print(boot)
summary(boot)
plot(boot)
```

`kootenay` *Kootenay River Water-Flow Measurements*

Description

Annual water-flow measurements (in units of 10^6 cubic feet) at two gauging stations on the Kootenay River: Libby (Montana, USA) and Newgate (British Columbia, Canada). The dataset contains a single X-space outlier (year 1934) where the Libby measurement is anomalously large, likely due to an upstream dam operation.

Usage

```
kootenay
```

Format

A data frame with 13 rows and 2 columns, with row names giving the year of measurement (1931–1943):

libby Water flow at Libby, Montana (10^6 cubic feet).

newgate Water flow at Newgate, British Columbia (10^6 cubic feet).

Details

This dataset is used in De Carvalho et al. (2017), Section 6.5, to illustrate robustness to X-space outliers. The recommended width estimator is `sigma_method = "s1"`.

Outlier structure

The observation for year 1934 has an anomalously large `libby` value (77.6 vs. a typical range of 17–39), making it an X-space outlier that strongly influences naive regression fits.

Source

J. Neter, M.H. Kutner, C.J. Nachtsheim, and W. Wasserman (1996). *Applied Linear Statistical Models*, 4th edition. Irwin/McGraw-Hill, p. 414.

Available in **robustbase** as `kootenay`.

References

De Carvalho, F.A.T., Lima Neto, E.A., Ferreira, M.R.P. (2017). doi:10.1016/j.neucom.2016.12.035

Examples

```
data(kootenay)

fit_ols  <- lm(newgate ~ libby, data = kootenay)
fit_gkrr <- gkrr(newgate ~ libby, data = kootenay, sigma_method = "s1")

plot(kootenay$libby, kootenay$newgate,
     xlab = "Libby flow (10^6 ft^3)", ylab = "Newgate flow (10^6 ft^3)",
     main = "Kootenay River Water Flow",
     pch = 19, col = "grey60")
# Highlight the X-space outlier (1934)
points(kootenay["1934", "libby"], kootenay["1934", "newgate"],
       col = "red", pch = 17, cex = 1.5)
abline(fit_ols,  col = "black",     lwd = 2, lty = 2)
abline(fit_gkrr, col = "firebrick", lwd = 2)
legend("topleft", c("OLS", "GKRR", "1934 (outlier)"),
       col = c("black","firebrick","red"),
       lty = c(2,1,NA), pch = c(NA,NA,17), lwd = 2, bty = "n")
```

`mammals` *Brain and Body Mass of 62 Mammal Species*

Description

Body mass (kg) and brain mass (g) for 62 species of land mammals. On the natural logarithm scale the relationship is approximately linear, but the dataset contains several leverage points (very large mammals such as African and Asian elephants) that strongly influence OLS estimates.

Usage

```
mammals
```

Format

A data frame with 62 rows and 5 columns:

species Common name of the mammal species (character).

body_mass Body mass in kilograms.

brain_mass Brain mass in grams.

log_body Natural logarithm of body mass.

log_brain Natural logarithm of brain mass.

Outlier structure

On the log scale, African elephant (log_body $\approx$ 8.8) and Asian elephant (log_body $\approx$ 7.8) are high-leverage observations. Additionally, some primates (e.g., humans) deviate from the overall trend (Y-space outliers relative to the bulk regression line).

Source

P.J. Rousseeuw and A.M. Leroy (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons.

Originally from: T. Allison and D.V. Cicchetti (1976). Sleep in mammals: ecological and constitutional correlates. *Science*, 194, 732–734.

Available in **MASS** as `mammals`.

Examples

```
data(mammals)

fit_ols  <- lm(log_brain ~ log_body, data = mammals)
fit_gkrr <- gkrr(log_brain ~ log_body, data = mammals,
                 sigma_method = "s3")

plot(mammals$log_body, mammals$log_brain,
     xlab = "log(body mass, kg)", ylab = "log(brain mass, g)",
     main = "Brain vs. Body Mass (62 Mammal Species)",
     pch = 19, col = "grey60", cex = 0.8)
# Label the leverage points
elephants <- mammals$species %in% c("African elephant","Asian elephant")
points(mammals$log_body[elephants], mammals$log_brain[elephants],
       col = "red", pch = 17, cex = 1.4)
abline(fit_ols,  col = "black",     lwd = 2, lty = 2)
abline(fit_gkrr, col = "firebrick", lwd = 2)
legend("topleft", c("OLS","GKRR","Elephants"),
       col = c("black","firebrick","red"),
       lty = c(2,1,NA), pch = c(NA,NA,17), lwd = 2, bty = "n")
```

---

plot.gkrr *Diagnostic plots for a GKRReg fit*

Description

Produces up to 6 diagnostic panels for a `"gkrr"` object. Point size is inversely proportional to the kernel weight k_{ii}, so outliers (small weights) appear large and red while well-fitted observations appear small and blue.

Usage

```
## S3 method for class 'gkrr'
plot(x, which = 1:5, n_id = 3L, ask = length(which) > 1L, ...)
```

Arguments

x	A "gkrr" object.
which	Integer vector selecting panels to draw (default 1:5).
n_id	Number of extreme observations to label in panels 1–5 (default 3).
ask	Logical; if TRUE waits for user input between panels (default TRUE when length(which) > 1).
...	Additional arguments (ignored; included for S3 compatibility).

Details

which = 1 Residuals vs. fitted values.

which = 2 Observed vs. fitted (y vs. $\hat{\mu}$).

which = 3 Kernel weight vs. residual, with the theoretical curve $G(e) = \exp(-e^2/\hat{\gamma}^2)$ overlaid.

which = 4 Kernel weight vs. observation index.

which = 5 Normal QQ-plot of residuals, coloured by weight.

which = 6 Objective function $S(\beta)$ by iteration.

Value

Invisibly returns x.

See Also

gkrr, plot.gkrr_boot

---

plot.gkrr_boot *Diagnostic plots for a GKRReg bootstrap object*

Description

Produces up to 2 panels for a "gkrr_boot" object.

Usage

```
## S3 method for class 'gkrr_boot'
plot(x, which = 1:2, ask = length(which) > 1L, ...)
```

Arguments

x	A "gkrr_boot" object.
which	Integer vector: 1 (histograms), 2 (scatter matrix), or 1:2 for both (default).
ask	Logical; waits for user input between panels when TRUE (default TRUE when length(which) > 1).
...	Additional arguments (ignored).

Details

which = 1 For each coefficient: histogram of bootstrap replicates with smoothed density, original estimate and shaded CI region.

which = 2 Scatter-plot matrix of bootstrap replicates across all pairs of coefficients, with a 95% confidence ellipse and Pearson correlation in each off-diagonal cell. Only available when the model has at least 2 coefficients.

Value

Invisibly returns x.

See Also

gkrr_boot, plot.gkrr

---

sigma2_estimators *Width hyper-parameter estimators for GKRReg*

Description

Four estimators for the Gaussian kernel width parameter γ^2. All functions receive y (observed response) and yhat (OLS fitted values) and return a positive scalar $\hat{\gamma}^2$.

Usage

```
sigma2_s1(y, yhat, alpha = 0.5)

sigma2_s2(y, yhat)

sigma2_s3(y, yhat, p)

sigma2_s4(y, yhat)
```

Arguments

y	Numeric vector of observed responses (S4 only).
yhat	Numeric vector of OLS fitted values (S4 only).
alpha	Fraction of the sample used in S1 (default 0.5).
p	Number of predictors excluding the intercept (used in S3 only).

Details

S1 — Caputo Mean of the 0.1 and 0.9 quantiles of the squared residuals on a sub-sample of size $n^* = \lfloor \alpha n \rfloor$. Recommended for clean data.

S2 — Pairwise median Median of $(y_i - \hat{\mu}_j^{\text{OLS}})^2, \backslash\, i \neq j$. Recommended for Y-space outliers up to 10% and X-space outliers up to 15%.

S3 — Residual variance $\hat{\gamma}^2 = \sum (y_i - \hat{\mu}_i)^2 / (n - p - 1)$. Recommended for Y-space outliers $\geq 15\%$ and leverage points.

S4 — AICc bandwidth Squares the bandwidth h selected by minimising the corrected Akaike information criterion in a nonparametric regression of y on $\hat{\mu}^{\text{OLS}}$, via `sm::h.select(..., method = "aicc")`. Recommended for large samples ($n \geq 200$). Requires the **sm** package.

Value

A positive scalar $\hat{\gamma}^2$.

References

De Carvalho et al. (2017) doi:10.1016/j.neucom.2016.12.035

`stars_cyg` *Hertzsprung-Russell Diagram of Star Cluster CYG OB1*

Description

The Hertzsprung-Russell diagram plots the log-luminosity against the log-effective-temperature of 47 stars in the star cluster CYG OB1. The dataset contains 4 giant stars (observations 11, 20, 30 and 34) that are well off the main sequence and act as leverage points, severely distorting OLS estimates of the linear trend in the main-sequence stars.

Usage

```
stars_cyg
```

Format

A data frame with 47 rows and 2 columns:

log_temp Logarithm (base 10) of the effective surface temperature of the star (Kelvin). Higher values correspond to hotter, bluer stars.

log_light Logarithm (base 10) of the light intensity of the star relative to that of the Sun.

Outlier structure

Observations 11, 20, 30 and 34 are giant stars that lie far from the main sequence (low temperature, high luminosity). They are leverage points because their `log_temp` values are much lower than the bulk of the data, and they also deviate from the linear relationship of the main sequence (bad leverage points).

Source

Humphreys, R.M. (1978). Studies of luminous stars in nearby galaxies. *Astrophysical Journal Supplement*, 38, 309–350.

Cited by: P.J. Rousseeuw and A.M. Leroy (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons, p. 27.

Available in **robustbase** as `starsCYG`.

References

Rousseeuw, P.J. and Leroy, A.M. (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons.

Examples

```
data(stars_cyg)

fit_ols  <- lm(log_light ~ log_temp, data = stars_cyg)
fit_gkrr <- gkrr(log_light ~ log_temp, data = stars_cyg,
                 sigma_method = "s3")

plot(stars_cyg$log_temp, stars_cyg$log_light,
     xlab = "log10(Temperature, K)", ylab = "log10(Luminosity, Sun = 1)",
     main = "Hertzsprung-Russell Diagram (CYG OB1)",
     pch = 19, col = "grey60")
# Highlight the 4 giant stars
giants <- c(11, 20, 30, 34)
points(stars_cyg$log_temp[giants], stars_cyg$log_light[giants],
       col = "red", pch = 17, cex = 1.5)
abline(fit_ols,  col = "black",     lwd = 2, lty = 2)
abline(fit_gkrr, col = "firebrick", lwd = 2)
legend("topleft", c("OLS","GKRR","Giant stars"),
       col = c("black","firebrick","red"),
       lty = c(2,1,NA), pch = c(NA,NA,17), lwd = 2, bty = "n")
```

`summary.gkrr` *Summary of a GKRReg fit*

Description

Prints a coefficient table modelled after `summary.lm`. By default inference is based on the sandwich variance estimator (always computed at fit time). When a `gkrr_boot` object is available it takes precedence, replacing the sandwich standard errors and p-values with their bootstrap counterparts.

Usage

```
## S3 method for class 'gkrr'
summary(
  object,
  boot = NULL,
  digits = 4L,
  signif_stars = TRUE,
  se_tol = 0.25,
  ...
)
```

Arguments

`object`	A `"gkrr"` object.
`boot`	Optional `"gkrr_boot"` object. If `NULL` (default), the method checks whether `object$boot` already contains one (set via `boot = TRUE` in `gkrr`). If neither is available, the sandwich estimator is used.
`digits`	Number of significant digits (default `4`).
`signif_stars`	Logical; print significance stars (default `TRUE`).
`se_tol`	Relative divergence threshold between sandwich and bootstrap standard errors. When both are available and $\lvert \mathrm{SE}_{\mathrm{sw}} - \mathrm{SE}_{\mathrm{boot}} \rvert / \mathrm{SE}_{\mathrm{boot}} > \mathrm{se\_tol}$ for any coefficient, a warning is issued listing which coefficients diverge and in which direction. Set to `Inf` to suppress the check. Default `0.25` (25%).
`...`	Currently ignored.

Details

Sandwich inference (default, `boot = FALSE` in `gkrr`): Standard errors are derived from the asymptotic sandwich covariance matrix $\hat{V} = A^{-1}BA^{-1}/n$, where $A = n^{-1}X^\top K \,\mathrm{diag}(1 - 2e_i^2/\hat{\gamma}^2)X$ and $B = n^{-1}X^\top K^2 \,\mathrm{diag}(e_i^2)X$. P-values use the two-sided Wald z-test $p_j = 2\,\Phi(-|\hat{\beta}_j/\mathrm{se}_j|)$. Confidence intervals are $\hat{\beta}_j \pm z_{\alpha/2}\,\mathrm{se}_j$.

Note: the sandwich estimator is asymptotically valid but does not account for the variability introduced by estimating $\hat{\gamma}^2$. For small samples or when γ^2 estimation has high variance, bootstrap inference (`boot = TRUE`) is more reliable.

Bootstrap inference (when `boot != FALSE`): Bootstrap p-values use the centred-t approach:

$$p_j = \frac{2}{B}\#\{|\hat{\beta}_b^* - \hat{\beta}_j| \geq |\hat{\beta}_j|\}$$

which counts how many bootstrap replicates are at least as extreme as the observed estimate relative to zero.

Value

Invisibly returns a list with components `coefficients` (the printed table as a matrix) and `r_squared`.

See Also

`gkrr`, `gkrr_boot`

Index