Package: HiGarrote (via r-universe)

October 13, 2026

Type Package

Title Nonnegative Garrote Method Incorporating Hierarchical Relationships

Version 1.0

Date 2024-09-24

Author Wei-Yang Yu [aut, cre], V. Roshan Joseph [aut]

Maintainer Wei-Yang Yu <wyu322@gatech.edu>

Description An implementation of the nonnegative garrote method that incorporates hierarchical relationships among variables. The core function, HiGarrote, offers an automated approach for analyzing experiments while respecting hierarchical structures among effects. For methodological details, refer to Yu and Joseph (2024) <https://arxiv.org/abs/2411.01383>. This work is supported by U.S. National Foundation grant DMS-2310637.

License GPL (>= 2)

Imports faux, Matrix, MaxPro, nloptr, purrr, quadprog, Rcpp (>= 1.0.12), RcppArmadillo, rlist, scales, stringr

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.3.2

Encoding UTF-8

Depends R (>= 2.10)

LazyData true

URL https://github.com/weiyangyu/HiGarrote

BugReports https://github.com/weiyangyu/HiGarrote/issues

Config/pak/sysreqs cmake make libicu-dev libxml2-dev

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

Date/Publication 2024-11-09 02:03:31 UTC

RemoteUrl https://github.com/weiyangyu/higarrote

RemoteRef HEAD

RemoteSha ebec81f1fdd5dc7dd647bf8e3a1cd27543910c5a

Contents

blood_glucose 2
cast_fatigue 2
HiGarrote 3
nnGarrote 5
router_bit 6

Index **7**

`blood_glucose` *Blood Glucose Experiment*

Description

Hamada and Wu (1992) analyzed an 18-run experiment designed to study blood glucose readings of a clinical testing device. The experiment contains one two-level factor and seven three-level quantitative factors, which are denoted by A through H.

Usage

```
data(blood_glucose)
```

Format

A data frame with 18 rows and 9 columns.

Source

Hamada, M. and Wu, C. F. J. (1992) "Analysis of Designed Experiments with Complex Aliasing," *Journal of Quality Technology*, 24, 130—137.

`cast_fatigue` *Cast Fatigue Experiment*

Description

Hunter et al. (1982) used a 12-run Plackett-Burman design to investigate the effects of seven two-level factors on the fatigue life of weld-repaired castings. The seven factors are denoted by capital letters A through G.

Usage

```
data(cast_fatigue)
```

Format

A data frame with 12 rows and 8 columns.

Source

Hunter, G. B., Hodi, F. S., and Eagar, T. W. (1982) "High Cycle Fatigue of Weld Repaired Cast Ti-6AI-4V," *Metallurgical Transactions A*, 13, 1589–1594.

`HiGarrote` *An Automatic Method for the Analysis of Experiments using Hierarchical Garrote*

Description

'HiGarrote()' provides an automatic method for analyzing experimental data. This function applies the nonnegative garrote method to select important effects while preserving their hierarchical structures. It first estimates regression parameters using generalized ridge regression, where the ridge parameters are derived from a Gaussian process prior placed on the input-output relationship. Subsequently, the initial estimates will be used in the nonnegative garrote for effects selection.

Usage

```
HiGarrote(
  D,
  y,
  quali_id = NULL,
  quanti_id = NULL,
  heredity = "weak",
  U = NULL,
  me_num = NULL,
  quali_contr = NULL
)
```

Arguments

`D`	An $n \times p$ data frame for the unreplicated design matrix, where n is the run size and p is the number of factors.
`y`	A vector for the responses corresponding to `D`. For replicated experiments, `y` should be an $n \times r$ matrix, where r is the number of replicates.
`quali_id`	A vector indexing qualitative factors.
`quanti_id`	A vector indexing quantitative factors.
`heredity`	Choice of heredity principles: weak or strong. The default is weak.
`U`	Optional. An $n \times P$ model matrix, where P is the number of potential effects. The inclusion of potential effects supports only main effects and two-factor interactions. Three-factor and higher order interactions are not supported. The colon symbol ":" must be included in the names of a two-factor interaction for separating its parent main effects. By default, `U` will be automatically constructed. The potential effects will then include all the main effects of qualitative factors, the first two main effects (linear and quadratic) of all the quantitative

factors, and all the two-factor interactions generated by those main effects. By default, the coding systems of qualitative and quantitative factors are Helmert coding and orthogonal polynomial coding, respectively.

`me_num` Optional. A $p \times 1$ vector for the main effects number of each factor. `me_num` is required when `U` is not `NULL` and must be consistent with the main effects number specified in `U`.

`quali_contr` Optional. A list specifying the contrasts of factors. `quali_contr` is required only when the main effects of a qualitative factor are not generated by the default Helmert coding.

Value

A vector for the nonnegative garrote estimates of the identified effects.

References

Yu, W. Y. and Joseph, V. R. (2024) "Automated Analysis of Experiments using Hierarchical Garrote," arXiv preprint arXiv:2411.01383.

Examples

```
# Cast fatigue experiment
data(cast_fatigue)
X <- cast_fatigue[,1:7]
y <- cast_fatigue[,8]
HiGarrote::HiGarrote(X, y)

# Blood glucose experiment
data(blood_glucose)
X <- blood_glucose[,1:8]
y <- blood_glucose[,9]
HiGarrote::HiGarrote(X, y, quanti_id = 2:8)

# Router bit experiment
data(router_bit)
X <- router_bit[, 1:9]
y <- router_bit[,10]
for(i in c(4,5)){
my.contrasts <- matrix(c(-1,-1,1,1,1,-1,-1,1,-1,1,-1,1), ncol = 3)
X[,i] <- as.factor(X[,i])
contrasts(X[,i]) <- my.contrasts
colnames(contrasts(X[,i])) <- paste0(".",1:(4-1))
}
U <- model.matrix(~.^2, X)
U <- U[, -1]  # remove the unnecessary intercept terms from the model matrix
me_num = c(rep(1,3), rep(3,2), rep(1, 4))
quali_contr <- list(NULL, NULL, NULL,
                    matrix(c(-1,-1,1,1,1,-1,-1,1,-1,1,-1,1), ncol = 3),
                    matrix(c(-1,-1,1,1,1,-1,-1,1,-1,1,-1,1), ncol = 3),
                    NULL, NULL, NULL, NULL)
HiGarrote::HiGarrote(X, y, quali_id = c(4,5), U = U,
```

```
me_num = me_num, quali_contr = quali_contr)

# Experiments with replicates
# Generate simulated data
data(cast_fatigue)
X <- cast_fatigue[,1:7]
U <- data.frame(model.matrix(~.^2, X)[,-1])
error <- matrix(rnorm(24), ncol = 2) # two replicates for each run
y <- 20*U$A + 10*U$A.B + 5*U$A.C + error
HiGarrote::HiGarrote(X, y)
```

---

nnGarrote *Nonnegative Garrote Method with Hierarchical Structures*

Description

'nnGarrote()' implements the nonnegative garrote method, as described in Yuan et al. (2009), for selecting important variables while preserving hierarchical structures. The method begins by obtaining the least squares estimates of the regression parameters under a linear model. These initial estimates are then used in the nonnegative garrote to perform variable selection. This function supports prediction based on the linear model fitted with the selected variables and their nonnegative garrote estimates. Note that this method is suitable only when the number of observations is much larger than the number of variables, ensuring that the least squares estimation remains reliable.

Usage

```
nnGarrote(U, y, new_U = NULL, heredity = "weak")
```

Arguments

U	An $n \times P$ model matrix, where n is the number of data and P is the number of potential variables. The inclusion of potential variables supports only up to second-order interactions. Three-order and higher order interactions are not supported. The colon symbol ":" must be included in the names of a second-order interaction for separating its parent variables. Please see the example for the naming format.
y	A vector for the responses.
new_U	Optional. A matrix or data frame of the new model matrix for prediction.
heredity	Choice of heredity principles: weak or strong. The default is weak.

Value

If new_U is NULL, the function returns a vector for the nonnegative garrote estimates of the identified variables.

If new_U is not NULL, the function returns a list with:

- beta_nng: a vector for the nonnegative garrote estimates of the identified variables.
- pred: predictions for the output corresponding to new_U.

References

Yuan, M., Joseph, V. R., and Zou H. (2009) "Structured Variable Selection and Estimation," The Annals of Applied Statistics, 3(4):1738–1757.

Examples

```
x1 <- runif(1000)
x2 <- runif(1000)
x3 <- runif(1000)
error <- rnorm(1000)
X <- data.frame(x1, x2, x3)
U_all <- data.frame(model.matrix(~. + x1:x2 + x1:x3 + x2:x3 + I(x1^2) + I(x2^2) + I(x3^2), X))
colnames(U_all) <- c("X.Intercept.", "x1", "x2", "x3", "x1:x1", "x2:x2", "x3:x3",
 "x1:x2", "x1:x3", "x2:x3")
# ":" is required for detecting the parent variables of a second-order interaction.

new_idx <- sample(1:1000, 800)
new_U <- U_all[new_idx,]
U_idx <- setdiff(1:1000, new_idx)
U <- U_all[U_idx,]
y_all <- 20*U_all$x1 + 15*U_all$`x1:x1` + 10*U_all$`x1:x2` + error
y <- y_all[U_idx]
nnGarrote(U, y, new_U)
```

`router_bit` *Router Bit Experiment*

Description

Phadke (1986) described a 32-run experiment aimed at increasing the lifespan of router bits used in a routing process to cut printed wiring boards from a panel. The experiment contains seven two-level factors and two four-level qualitative factors which are denoted by A through J with the exclusion of I.

Usage

```
data(router_bit)
```

Format

A data frame with 32 rows and 10 columns.

Source

Phadke, M. S. (1986) "Design Optimization Case Studies," *AT&T Technical Journal*, 65, 51–68.

Index

∗ **datasets**
    blood_glucose, 2
    cast_fatigue, 2
    router_bit, 6

blood_glucose, 2

cast_fatigue, 2

HiGarrote, 3

nnGarrote, 5

router_bit, 6