Package ‘coxphMIC’

April 26, 2017

Type Package

Title Sparse Estimation of Cox Proportional Hazards Models via Approximated Information Criterion

Version 0.1.0

Author Xiaogang Su and Razieh Nabi Abdolyousefi

Maintainer Xiaogang Su <xiaogangsu@gmail.com>

Description Sparse estimation for Cox PH models is done via Minimum approximated Information Criterion (MIC) by Su, Wijayasinghe, Fan, and Zhang (2016) <DOI:10.1111/biom.12484>. MIC mimics the best subset selection using a penalized likelihood approach yet with no need of a tuning parameter. The problem is further reformulated with a re-parameterization step so that it reduces to one unconstrained non-convex yet smooth programming problem, which can be solved efficiently. Furthermore, the re-parameterization tactic yields an additional advantage in terms of circumventing post-selection inference.

License GPL-2

Depends R (>= 3.1.0), stats (>= 3.2.5), graphics (>= 3.2.5), utils (>= 3.2.5)

Imports survival (>= 2.38), numDeriv (>= 2014.2-1)

LazyData TRUE

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-04-26 05:56:07 UTC

R topics documented:

coxphMIC — *Sparse Estimation for a Cox PH model via Approximated Information Criterion*

Description

Sparse Estimation for a Cox PH model via Approximated Information Criterion

Usage

```
coxphMIC(formula = Surv(time, status) ~ ., data, method.beta0 = "MPLE",
  beta0 = NULL, theta0 = 1, method = "BIC", lambda0 = 2, a0 = NULL,
  scale.x = TRUE, maxit.global = 300, maxit.local = 100,
  rounding.digits = 4, zero = sqrt(.Machine$double.eps),
  compute.se.gamma = TRUE, compute.se.beta = TRUE, CI.gamma = TRUE,
  conf.level = 0.95, details = FALSE)
```

Arguments

`formula`	A formula object, with the response on the left of a ~ operator, and the terms on the right. The response must be a survival object as returned by the `Surv` function.
`data`	A data.frame in which to interpret the variables named in the `formula` argument.
`method.beta0`	A method to supply the starting point for beta with choices: `"MPLE"` and `"ridge"`. By default, the maximum partial likelihood estimator (MPLE) is used with `"MPLE"`. The option `"ridge"` asks for a ridge estimator with penalty parameter specified by `theta0`. You may supply a set of values for beta0 of your choice. If `NULL`, then beta0 is set as 0.
`beta0`	User-supplied beta0 value, the starting point for optimization.
`theta0`	Specified the penalty parameter for the ridge estimator when `method.beta0="ridge"`.
`method`	Specifies the model selection criterion used. If `"AIC"`, the complexity penalty parameter (lambda) equals 2; if `"BIC"`, lambda equals ln(n0), where n0 denotes the number of uncensored events. You may specify the penalty parameter of your choice by setting `lambda0`.
`lambda0`	User-supplied penalty parameter for model complexity. If `method="AIC"` or `"BIC"`, the value of `lambda0` will be ignored.
`a0`	The scale (or sharpness) parameter used in the hyperbolic tangent penalty. By default, `a0` is set as n0, where n0 is again the number of uncensored events.
`scale.x`	Logical value: should the predictors X be normalized? Default to `TRUE`.
`maxit.global`	Maximum number of iterations allowed for the global optimization algorithm – `SANN`. Default value is 300.
`maxit.local`	Maximum number of iterations allowed for the local optimizaiton algorithm – `BFGS`. Default value is 100.

rounding.digits
: Number of digits after the decimal point for rounding-up estiamtes. Default value is 4.

zero
: Tolerance level for convergence. Default is `sqrt(.Machine$double.eps)`.

compute.se.gamma
: Logical value indicating whether to compute the standard errors for gamma in the reparameterization. Default is `TRUE`.

compute.se.beta
: Logical value indicating whether to compute the standard errors for nonzero beta estimates. Default is `TRUE`. Note that this result is subject to post-selection inference.

CI.gamma
: Logical indicator of whether the confidence inverval for gamma is outputed. Default is `TRUE`.

conf.level
: Specifies the confidence level for `CI.gamma`. Defaulted as 0.95.

details
: Logical value: if `TRUE`, detailed results will be printed out when running `coxphMIC`.

Details

The main idea of MIC involves approximation of the l0 norm with a continuous or smooth unit dent function. This method bridges the best subset selection and regularization by borrowing strength from both. It mimics the best subset selection using a penalized likelihood approach yet with no need of a tuning parameter.

The problem is further reformulated with a reparameterization step by relating `beta` to `gamma`. There are two benefits of doing so: first, it reduces the optimization to one unconstrained nonconvex yet smooth programming problem, which can be solved efficiently as in computing the maximum partial likelihood estimator (MPLE); furthermore, the reparameterization tactic yields an additional advantage in terms of circumventing post-selection inference. Significance testing on `beta` can be done through `gamma`.

The solve the smooth yet nonconvex optimization, a simulated annealing (`method="SANN"` option in `optim`) global optimization algorithm is first applied. The resultant estimator is then used as the starting point for another local optimization algorithm. The quasi-Newton BFGS method (`method="BFGS"` in `optim`) is used.

In its current version, some appropriate data preparation might be needed. For example, nomincal variables (especially character-valued ones) needed to be coded with dummy variables; missing values would cause errors too and hence need prehanlding too.

Value

A list containing the following component is returned.

opt.global Results from the preliminary run of a global optimization procedure (`SANN` as default).

opt.local Results from the second run of a local optimization procedure (`BFGS` as default).

min.Q Value of the minimized objective function.

gamma Estimated gamma;

beta Estimated beta;

VCOV.gamma The estimated variance-covariance matrix for the gamma estimate;

se.gamma Standard errors for the gamma estimate;

se.beta Standard errors for the beta estimate (post-selection);

BIC The BIC value for the *selected* model;

result A summary table of the fitting results.

call the matched call.

References

- Abdolyousefi, R. N. and Su, X. (2016). **coxphMIC**: An R package for sparse estimation of Cox PH Models via approximated information criterion. Tentatively accepted, *The R Journal*.
- Su, X. (2015). Variable selection via subtle uprooting. *Journal of Computational and Graphical Statistics*, **24**(4): 1092–1113. URL http://www.tandfonline.com/doi/pdf/10.1080/10618600.2014.955176
- Su, X., Wijayasinghe, C. S., Fan, J., and Zhang, Y. (2015). Sparse estimation of Cox proportional hazards models via approximated information criteria. *Biometrics*, **72**(3): 751–759. URL http://onlinelibrary.wiley.com/doi/10.1111/biom.12484/epdf

See Also

coxph

Examples

```
# PREPARE THE PBC DATA
library(survival); data(pbc);
dat <- pbc; dim(dat);
dat$status <- ifelse(pbc$status==2, 1, 0)
# HANDLE CATEGORICAL VARIABLES
dat$sex <- ifelse(pbc$sex=="f", 1, 0)
# LISTWISE DELETION USED TO HANDLE MISSING VALUES
dat <- stats::na.omit(dat);
dim(dat); utils::head(dat)

fit.mic <- coxphMIC(formula=Surv(time, status)~.-id, data=dat, method="BIC", scale.x=TRUE)
names(fit.mic)
print(fit.mic)
plot(fit.mic)
```

`LoglikPen` *Compute the penalized log partial likelihood for a Cox PH model with MIC penalty*

Description

Compute the penalized log partial likelihood for a Cox PH model with MIC penalty

Usage

```
LoglikPen(beta, time, status, X, lambda, a)
```

Arguments

`beta`	A p-dimensional vector containing the regression ceofficients in the CoxPH model.
`time`	The observed survival time.
`status`	The status indicator: 1 for event and 0 for censoring.
`X`	An n by p design matrix.
`lambda`	The penalty parameter euqals either 2 in AIC or ln(n0) in BIC (by default), where n0 is the number of uncensored survival times observed in the data. You can also specify it to a specific value of your own choice.
`a`	The scale parameter in the hyperbolic tangent function of the MIC penalty. By default, $a = n0$, i.e., the number of uncensored survival times observed in the data.

Value

The value of the penalized log partial likelihood function evaluated at beta.

References

- Abdolyousefi, R. N. and Su, X. (2016). **coxphMIC**: An R package for sparse estimation of Cox PH Models via approximated information criterion. Tentatively accepted, *The R Journal*.
- Su, X. (2015). Variable selection via subtle uprooting. *Journal of Computational and Graphical Statistics*, **24**(4): 1092–1113. URL http://www.tandfonline.com/doi/pdf/10.1080/10618600.2014.955176
- Su, X., Wijayasinghe, C. S., Fan, J., and Zhang, Y. (2015). Sparse estimation of Cox proportional hazards models via approximated information criteria. *Biometrics*, **72**(3): 751–759. URL http://onlinelibrary.wiley.com/doi/10.1111/biom.12484/epdf

See Also

`coxph`

`plot.coxphMIC` *The Generic* `plot` *Function for Object of* `coxphMIC` *Class*

Description

The Generic `plot` Function for Object of `coxphMIC` Class

Usage

```
## S3 method for class 'coxphMIC'
plot(x, conf.level = 0.95, horizontal = TRUE,
  mar = rep(4.5, 4), ...)
```

Arguments

`x`	an object of `coxphMIC` class.
`conf.level`	confidence level used for error bar plots. Default is 0.95.
`horizontal`	Logical indicator of horizontal alignment. Default is `TRUE`.
`mar`	margin in terms of the number of lines to be specified on the four sides of the plot
`...`	further arguments passed to or from other methods.

Details

The (generic) plot method for an `coxphMIC` object. It plots MIC estimates of gamma and beta. For 0 beta estimates, their corresponding SE are reset to 0 to make the plot.

Value

Error bar plots for estimated gamma and beta at a given confidenve level.

References

- Abdolyousefi, R. N. and Su, X. (2016). **coxphMIC**: An R package for sparse estimation of Cox PH Models via approximated information criterion. Tentatively accepted, *The R Journal*.
- Su, X. (2015). Variable selection via subtle uprooting. *Journal of Computational and Graphical Statistics*, **24**(4): 1092–1113. URL `http://www.tandfonline.com/doi/pdf/10.1080/10618600.2014.955176`
- Su, X., Wijayasinghe, C. S., Fan, J., and Zhang, Y. (2015). Sparse estimation of Cox proportional hazards models via approximated information criteria. *Biometrics*, **72**(3): 751–759. URL `http://onlinelibrary.wiley.com/doi/10.1111/biom.12484/epdf`

See Also

`coxphMIC`

print.coxphMIC — *The Generic* `print` *Function for Object of* `coxphMIC` *Class*

Description

The Generic `print` Function for Object of `coxphMIC` Class

Usage

```
## S3 method for class 'coxphMIC'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

`x`	an object of `coxphMIC` class.
`digits`	the minimal number of significant digits. See `print.default`.
`...`	further arguments passed to or from other methods.

Details

The (generic) print method for an `coxphMIC` object. The results include info on the estimated gamma and beta. Depending on the options, significance testing and confidence intervals are also provided.

Value

The table of estimated regression coefficients beta and the reparameterized gamma.

References

- Abdolyousefi, R. N. and Su, X. (2016). **coxphMIC**: An R package for sparse estimation of Cox PH Models via approximated information criterion. Tentatively accepted, *The R Journal*.
- Su, X. (2015). Variable selection via subtle uprooting. *Journal of Computational and Graphical Statistics*, **24**(4): 1092–1113. URL http://www.tandfonline.com/doi/pdf/10.1080/10618600.2014.955176
- Su, X., Wijayasinghe, C. S., Fan, J., and Zhang, Y. (2015). Sparse estimation of Cox proportional hazards models via approximated information criteria. *Biometrics*, **72**(3): 751–759. URL http://onlinelibrary.wiley.com/doi/10.1111/biom.12484/epdf

See Also

`coxphMIC`

Index