Description

cv.smog conducts a greedy-search for optimal lambda’s and yields a sparse model given a provided model selection criterion. When type is “nloglike”, the method allows the nfolds to be processed in parallel for speeding up the cross-validation.

Usage

cv.smog(x, y, g, v, label, type, family = "gaussian", lambda.max = NULL, nlambda.max = 20, delta = 0.9, nfolds = 5, parallel = FALSE, ncores = NULL, ...)

Arguments

x a model matrix, or a data frame of dimensions n by p, in which the columns represents the predictor variables.

y response variable, corresponds to the family description. When family is “gaussian” or “binomial”, y ought to be a numeric vector of observations of length n; when family is “coxph”, y represents the survival objects, containing the survival time and the censoring status. See Surv.

g a vector of group labels for the predictor variables.

v a vector of binary values, represents whether or not the predictor variables are penalized. Note that 1 indicates penalization and 0 for not penalization.

label a character vector, represents the type of predictors in terms of treatment, prognostic, and predictive effects by using “t”, “prog”, and “pred”, respectively.

type model selection criterion, should choose from “nloglike”, “cAIC”, “AIC”, “BIC”, and “GCV”, respectively.

family a description of the distribution family for the response variable variable. For continuous response variable, family is “gaussian”; for multinomial or binary response variable, family is “binomial”; for survival response variable, family is “coxph”, respectively.
The maximum value for lambda’s. If NULL, the default lambda.max is \(1/\lambda_{\text{min}}(x'x)\).

- **nlambda.max**: the maximum number of lambdas’ shrunk down from the maximum lambda lambda.max. Default is 20.
- **delta**: the damping rate for lambda’s such that \(\lambda_k = \delta^k \lambda_0\). Default is 0.9.
- **nfolds**: number of folds. One fold of the observations in the data are used as the testing, and the remaining are fitted for model training. Default is 5.
- **parallel**: Whether or not process the nfolds cross-validations in parallel. If TRUE, use foreach to do each cross-validation in parallel. Default is FALSE.
- **ncores**: number of cpu’s for parallel computing. See makeCluster and registerDoParallel. Default is NULL.
- **...**: other arguments that can be supplied to smog.

## Details

When the type is “nloglike”, it requires doing nfolds cross-validations. For each cross-validation, evenly split the whole data into nfolds, and one fold of the observations are used as the testing data, and the remaining are used for model training. After calculating the AIC for each fold of testing data, return the average of the AICs. Note that we keep lambda2 = 0 during the greedy search for lambda’s.

## Value

- **cvfit**: the fitted model based on the optimal lambda’s.
- **lhat**: the optimal lambda’s which has the minimum model selection criterion.
- **profile**: a data frame contains the path of lambda’s and the corresponding model selection criterion, which is determined by the type.

## Author(s)

Chong Ma, <chongma8903@gmail.com>.

## References


See Also

smog.default, smog.formula, predict.smog, plot.smog.
Examples

# generate design matrix x
set.seed(2018)
n=50; p=100
s=10
x=matrix(0,n,1+2*p)
x[,1]=sample(c(0,1),n,replace = TRUE)
x[,seq(2,1+2*p,2)]=matrix(rnorm(n*p),n,p)
x[,seq(3,1+2*p,2)]=x[,seq(2,1+2*p,2)]x[,1]

# groups
v=c(0,rep(1, 2*p))
# penalization status
label=c("t",rep("prog","pred"),p)) # type of predictor variables

# generate beta
beta=c(rnorm(13,0,2),rep(0, ncol(x)-13))
beta[c(2,4,7,9)]=0

# generate y
data=x%*%beta
noise=rnorm(n)

snr=as.numeric(sqrt(var(data)/(s*var(noise))))
y=data+snr*noise

# generate y


glog

Generalized linear model constraint on hierarchical structure by using overlapped group penalty

Description

Generalized linear model constraint on hierarchical structure by using overlapped group penalty

Usage

glog(y, x, g, v, lambda, hierarchy, family = "gaussian", rho = 10, scale = TRUE, eabs = 0.001, erel = 0.001, LL = 1, eta = 1.25, maxitr = 1000L)

Arguments

y response variable, in the format of matrix. When family is “gaussian” or “binomial”, y ought to be a matrix of n by 1 for the observations; when family is “coxph”, y represents the survival objects, that is, a matrix of n by 2, containing the survival time and the censoring status. See Surv.
x a model matrix of dimensions n by p, in which the column represents the predictor variables.
g a numeric vector of group labels for the predictor variables.
v a numeric vector of binary values, represents whether or not the predictor variables are penalized. Note that 1 indicates penalization and 0 for not penalization.
lambda a numeric vector of three penalty parameters corresponding to L2 norm, squared L2 norm, and L1 norm, respectively.
hierarchy a factor value in levels 0, 1, 2, which represent different hierarchical structure within groups, respectively. When hierarchy=0, $\lambda_2$ and $\lambda_3$ are forced to be zeroes; when hierarchy=1, $\lambda_2$ is forced to be zero; when hierarchy=2, there is no constraint on $\lambda$’s. See smog.
family a description of the distribution family for the response variable variable. For continuous response variable, family is “gaussian”; for multinomial or binary response variable, family is “binomial”; for survival response variable, family is “coxph”, respectively.
rho the penalty parameter used in the alternating direction method of multipliers algorithm (ADMM). Default is 10.
scale whether or not scale the design matrix. Default is true.
eabs the absolute tolerance used in the ADMM algorithm. Default is 1e-3.
erel the relative tolerance used in the ADMM algorithm. Default is 1e-3.
LL initial value for the Lipschitz continuous constant for approximation to the objective function in the Majorization-Minimization (MM) (or iterative shrinkage-thresholding algorithm (ISTA)). Default is 1.
etagradient stepsize for the backtrack line search for the Lipschitz continuous constant. Default is 1.25.
maxitr the maximum iterations for convergence in the ADMM algorithm. Default is 500.

Value
A list of

coefficients A data frame of the variable name and the estimated coefficients
llikelihood The log likelihood value based on the ultimate estimated coefficients
loglike The sequence of log likelihood values since the algorithm starts
PrimalError The sequence of primal errors in the ADMM algorithm
DualError The sequence of dual errors in the ADMM algorithm
converge The integer of the iteration when the convergence occurs

Author(s)
Chong Ma, <chongma8903@gmail.com>.
References


See Also
cv.smog, smog.default, smog.formula, predict.smog, plot.smog.

Examples

set.seed(2018)
# generate design matrix x
n=50; p=100
s=10
x=matrix(0,n,1+2*p)
x[,1]=sample(c(0,1),n,replace = TRUE)
x[,seq(2,1+2*p,2)]=matrix(rnorm(n*p),n,p)
x[,seq(3,1+2*p,2)]=x[,seq(2,1+2*p,2)]*x[,1]

g=c(p+1,rep(1:p,rep(2,p)))  # groups
v=c(0,rep(1,2*p))  # penalization status

# generate beta
beta=c(rnorm(13,0,2),rep(0,ncol(x)-13))
beta[c(2,4,7,9)]=0

# generate y
data1=x%*%beta
noise1=runif(n)

snr1=as.numeric(sqrt(var(data1)/(s*var(noise1))))
y1=data1+snr1*noise1
lambda = c(8,0,8)
hierarchy = 1


penalty

Penalty function on the composite L2, L2-Square, and L1 penalties

Description

Penalty function on the composite L2, L2-Square, and L1 penalties

Usage

penalty(x, lambda, hierarchy, d)
Arguments

x  A vector of two numeric values, in which \( x_1 \) represents the prognostic effect, and \( x_2 \) for the predictive effect, respectively.

lambda  a vector of three penalty parameters. \( \lambda_1 \) and \( \lambda_2 \) are L2 and L2-Square (ridge) penalties for \( x \) in a group level, and \( \lambda_3 \) is the L1 penalty for \( x_2 \), respectively.

hierarchy  a factor value in levels 0, 1, 2, which represent different hierarchical structure in \( x \), respectively. When \( \text{hierarchy}=0 \), \( \lambda_2 \) and \( \lambda_3 \) are forced to be zeroes; when \( \text{hierarchy}=1 \), \( \lambda_2 \) is forced to be zero; when \( \text{hierarchy}=2 \), there is no constraint on \( \lambda \)'s. See smog.

d  indices for overlapped variables in \( x \).

Value

A numeric value of the penalty function based on the composition of L2, L2-Square, and L2 penalties.

Author(s)

Chong Ma, <chongma8903@gmail.com>.

References


See Also

cv.smog, smog.default, smog.formula, predict.smog, plot.smog.

Examples

penalty(x = rnorm(6,2,1), lambda = c(0.5,0.3,0.1), hierarchy = 0, d = c(1,1,2,2,3,3))

Description

plot.smog can produce a panel of plots for the primal errors, dual errors, and the penalized log-likelihood values, based on the provided fitted model (\( x \)) in the S3method of smog.

Usage

## S3 method for class 'smog'
plot(x, type = "l", xlab = "iteration",
    caption = list("primal error", "dual error", "log-likelihood"), ...)

plot method for objects of the class smog
Arguments

- `x`: a fitted object of class inheriting from smog.
- `type`, `xlab`: default line types and x axis labels for the panel of plots.
- `caption`: a list of y axes labels for the panel of plots.
- `...`: additional arguments that could be supplied to `plot` and `par`.

Details

For the panel of three plots, the `xlab` is “iterations” and the `type` is “l”, by default. The `ylab` are “primal error”, “dual error”, “log-likelihood”, respectively. This panel of plots can reflect the convergence performance for the algorithm used in `smog`.

Author(s)

Chong Ma, <chongma8903@gmail.com>.

References


See Also

- `par`, `plot.default`, `predict.smog`, `smog.default`, `smog.formula`, `cv.smog`.

**predict.smog**

Predict method for objects of the class smog

Description

`predict.smog` can produce the prediction for user-given new data, based on the provided fitted model (`object`) in the S3 method of `smog`. If the `newdata` omitted, it would output the prediction for the fitted model itself. The yielded result should match with the family in the provided model. See `smog`.

Usage

```r
## S3 method for class 'smog'
predict(object, newdata = NULL, family = "gaussian", ...)
```

Arguments

- `object`: a fitted object of class inheriting from smog.
- `newdata`: a data frame containing the predictor variables, which are used to predict. If omitted, the fitted linear predictors are used.
- `family`: a description of distribution family for which the response variable is to be predicted.
- `...`: additional arguments affecting the predictions produced.
prox

Details
If newdata = NULL, the fitted.value based on the object is used for the prediction.

Value
If family = “gaussian”, a vector of prediction for the response is returned. For family = “coxph”, a vector of predicted survival probability is returned. When family = “binomial”, it outputs a data frame containing the predicted group labels and the corresponding probabilities.

Author(s)
Chong Ma, <chongma8903@gmail.com>.

References

See Also
smog.default, smog.formula, cv.smog, plot.smog.

prox
Composite proximal operator based on L2, L2-Square, and L1 penalties

Description
Composite proximal operator based on L2, L2-Square, and L1 penalties

Usage
prox(x, lambda, hierarchy, d)

Arguments
x A numeric vector of two.
lambda a vector of three penalty parameters. \( \lambda_1 \) and \( \lambda_2 \) are L2 and L2-Square (ridge) penalties for \( x \) in a group level, and \( \lambda_3 \) is the L1 penalty for \( x_2 \), respectively.
hierarchy a factor value in levels 0, 1, 2, which represent different hierarchical structure in \( x \), respectively. When hierarchy=0, \( \lambda_2 \) and \( \lambda_3 \) are forced to be zeroes; when hierarchy=1, \( \lambda_2 \) is forced to be zero; when hierarchy=2, there is no constraint on \( \lambda \)'s. See smog.
d indices for overlapped variables in \( x \).

Value
A two-dimensional numerical vector, soft-thresholded based on a composition of \( \lambda_1 \), \( \lambda_2 \), and \( \lambda_3 \).
Author(s)

Chong Ma, <chongma8903@gmail.com>.

References


See Also

cv.smog, smog.default, smog.formula, predict.smog, plot.smog.

Examples

```r
prox(x = rnorm(6,2,1), lambda = c(0.5,0.3,0.1), hierarchy = 0, d = c(1,1,2,2,3,3))
```

```
proxL1
L1 proximal operator
```

Description

L1 proximal operator

Usage

```r
proxL1(x, lambda)
```

Arguments

- **x**: numeric value.
- **lambda**: numeric value for the L1 penalty parameter.

Value

A numeric value soft-thresholded by \( \lambda \), which is \( \text{sign}(x)(|x| - \lambda)_+ \).

Author(s)

Chong Ma, <chongma8903@gmail.com>.

References


Examples

```r
proxL1(2.0,0.5)
```
proxL2

L2 proximal operator

Description

L2 proximal operator

Usage

proxL2(x, lambda)

Arguments

x
A vector of p numerical values.

lambda
decimal value for the L2 penalty parameter.

Value

A numeric vector soft-thresholded by \( \lambda \) as a group, which is \( (1 - \frac{\lambda \sqrt{\pi}}{\sqrt{x_1^2 + \cdots + x_p^2}}) \cdot x \).

Author(s)

Chong Ma, <chongma8903@gmail.com>

References


Examples

proxL2(rnorm(6,2,1),0.5)

sim_rct_biomarker

Simulate a randomized clinical trial with biomarkers

Description

sim_rct_biomarker is used to simulate clinical trial data with specified treatment, prognostic, and predictive effect sizes.

Usage

sim_rct_biomarker(n = 50, p = 100, p_prog = 5, p_pred = 5, p_both = 5,

v_trt = 0.4, v_prog = 0.2, v_pred = 0.2, v_err = 0.2, corr = NULL,

family = "gaussian", ...)


Arguments

- \( n \) Number of subjects.
- \( p \) Number of biomarkers.
- \( p_{\text{prog}} \) Number of biomarkers with prognostic effects only.
- \( p_{\text{pred}} \) Number of biomarkers with predictive effects only.
- \( p_{\text{both}} \) Number of biomarkers with both prognostic and predictive effects.
- \( v_{\text{trt}} \) Variance of response due to treatment.
- \( v_{\text{prog}} \) Variance of response due to prognostic effects.
- \( v_{\text{pred}} \) Variance of response due to predictive effects.
- \( v_{\text{err}} \) Variance of response due to random noise.
- \( \text{corr} \) Autocorrelation parameter between biomarkers, default is NULL.
- \( \text{family} \) The distribution family for response variable, can be “gaussian”, or “binomial”. Default is “gaussian”.
- \( \ldots \) further arguments passed to or from other methods.

Value

A list containing several variables.

- \( T \) Treatment status in 1 or -1 values.
- \( X \) Biomarkers.
- \( W \) Hadamard product of treatment and biomarkers.
- \( M \) Model matrix - binding of \( T \), \( X \), and \( W \).
- \( Y \) Response.
- \( Y_0 \) Response without error.
- \( \tau \) Treatment effect.
- \( \beta \) Prognostic effects.
- \( \gamma \) Predictive effects.
- \( \theta \) All effects corresponding to \( M \).

Author(s)

Chong Ma <chong.ma@yale.edu>, Kevin Galinsky <Kevin.Galinsky@takeda.com>.

References


Examples

```r
sim <- sim_rct_biomarker(n = 1e5)
var(sim$T * sim$tau)
var(sim$X %*% sim$beta)
var(sim$W %*% sim$gamma)
```
Description

smog fits a linear non-penalized phenotype (demographic) variables such as age, gender, treatment, etc. and penalized groups of prognostic effect (main effect) and predictive effect (interaction effect), by satisfying the hierarchy structure: if a predictive effect exists, its prognostic effect must be in the model. It can deal with continuous, binomial or multinomial, and survival response variables, underlying the assumption of Gaussian, binomial (multinomial), and Cox proportional hazard models, respectively. It can accept formula, and output coefficients table, fitted.values, and convergence information produced in the algorithm iterations.

Usage

## Default S3 method:
smog(x, y, g, v, label, lambda1, lambda2, lambda3,
     family = "gaussian", subset = NULL, rho = 10, scale = TRUE,
     eabs = 0.001, erel = 0.001, LL = 1, eta = 1.25, maxitr = 1000, ...)

## S3 method for class 'formula'
smog(formula, data = list(), g, v, label, lambda1, lambda2,
     lambda3, ...)

Arguments

x  a model matrix, or a data frame of dimensions n by p, in which the columns represents the predictor variables.

y  response variable, corresponds to the family description. When family is “gaussian” or “binomial”, y ought to be a numeric vector of observations of length n; when family is “coxph”, y represents the survival objects, containing the survival time and the censoring status. See Surv.

g  a vector of group labels for the predictor variables.

v  a vector of binary values, represents whether or not the predictor variables are penalized. Note that 1 indicates penalization and 0 for not penalization.

label a character vector, represents the type of predictors in terms of treatment, prognostic, and predictive effects by using “t”, “prog”, and “pred”, respectively.

lambda1 penalty parameter for the L2 norm of each group of prognostic and predictive effects.

lambda2 ridge penalty parameter for the squared L2 norm of each group of prognostic and predictive effects.

lambda3 penalty parameter for the L1 norm of predictive effects.
family  a description of the distribution family for the response variable. For continuous response variable, family is “gaussian”; for multinominal or binary response variable, family is “binomial”; for survival response variable, family is “coxph”, respectively.

subset  an optional vector specifying a subset of observations to be used in the model fitting. Default is NULL.

rho  the penalty parameter used in the alternating direction method of multipliers (ADMM) algorithm. Default is 10.

scale  whether or not scale the design matrix. Default is TRUE.

eabs  the absolute tolerance used in the ADMM algorithm. Default is 1e-3.

erel  the relative tolerance used in the ADMM algorithm. Default is 1e-3.

LL  initial value for the Lipschitz continuous constant for approximation to the objective function in the Majorization-Minimization (MM) (or iterative shrinkage-thresholding algorithm (ISTA)). Default is 1.

et  gradient stepsize for the backtrack line search for the Lipschitz continuous constant. Default is 1.25.

maxitr  the maximum iterations for convergence in the ADMM algorithm. Default is 1000.

...  other relevant arguments that can be supplied to smog.

formula  an object of class “formula”: a symbolic description of the model to be fitted. Should not include the intercept.

data  an optional data frame, containing the variables in the model.

Details

The formula has the form response ~ 0 + terms where terms is a series of predictor variables to be fitted for response. For gaussian family, the response is a continuous vector. For binomial family, the response is a factor vector, in which the last level denotes the “pivot”. For coxph family, the response is a Surv object, containing the survival time and censoring status.

The terms contains the non-penalized predictor variables, and many groups of prognostic and predictive terms, where in each group the prognostic term comes first, followed by the predictive term.

Different hierarchical structures within groups can result from adjusting the penalty parameters in the penalty function:

$$\Omega(\beta) = \lambda_1 \|\beta\| + \lambda_2 \|\beta\|^2 + \lambda_3 |\beta_2|$$

Where $\beta = (\beta_1, \beta_2)$. Note that $\beta_1$ denotes the prognostic effect (main effect), and $\beta_2$ for the predictive effect (interactive effect), respectively. When $\lambda_2 = 0$ and $\lambda_3 = 0$, it indicates no structure within groups. When $\lambda_2 \neq 0$, the penalty function honors the structure within groups such that: predictive effect $\neq 0 \implies$ prognostic effect $\neq 0$.

rho, eabs, erel, LL, eta are the corresponding parameters used in the iterative shrinkage-thresholding algorithm (ISTA) and the alternating direction method of multipliers algorithm (ADMM).

Note that the missing values in the data are supposed to be dealt with in the data preprocessing, before applying the method.
**Value**

`smog` returns an object of class inhering from “smog”. The generic accessor functions `coef`, `coefficients`, `fitted.value`, and `predict` can be used to extract various useful features of the value returned by `smog`.

An object of “smog” is a list containing at least the following components:

- **coefficients**: a data frame containing the nonzero predictor variables’ indexes, names, and estimates. When family is “binomial”, the estimates have K-1 columns, each column representing the weights for the corresponding group. The last group behaves the “pivot”.
- **fitted.values**: the fitted mean values for the response variable, for family is “gaussian”. When family is “binomial”, the fitted.values are the probabilities for each class; when family is “coxph”, the fitted.values are survival probabilities.
- **model**: a list of estimates for the intercept, treatment effect, and prognostic and predictive effects for the selectd biomarkers.
- **weight**: the weight of predictors resulted from the penalty function, is used to calculate the degrees of freedom.
- **DF**: the degrees of freedom. When family = “gaussian”, \( DF = \text{tr}(x'_\lambda x_\lambda + W)x_\lambda) \). For other families, DF is approximated by \( \text{diag}(1/(1 + W)) \).
- **criteria**: model selection criteria, including the correction Akaike’s Information Criterion (AIC), AIC, Bayesian Information Criterion (BIC), and the generalized cross-validation score (GCV), respectively. See also `cv.smog`.

\[
\text{cAIC} = \frac{n}{2}\log(2\times \text{log-likelihood}) + \frac{n}{2}\left(\frac{1+k/n}{1-k+2/n}\right).
\]

- **AIC**: \( \log(2\times \text{log-likelihood})/n + 2\frac{k}{n} \).
- **BIC**: \( \log(2\times \text{log-likelihood})/n + 2\frac{k}{n}\log(n) \).
- **GCV**: \( 2\times \text{log-likelihood}/(n(1-k/n)^2) \).

Where k is the degrees of freedom DF, which is related to the penalty parameters \( \lambda \)’s.

- **llikelihood**: the log-likelihood value for the converged model.
- **loglike**: the penalized log-likelihood values for each iteration in the algorithm.
- **PrimalError**: the averaged norms \( ||\beta - Z||/\sqrt{p} \) for each iteration, in the ADMM algorithm.
- **DualError**: the averaged norms \( ||Z^{t+1} - Z^t||/\sqrt{p} \) for each iteration, in the ADMM algorithm.
- **converge**: the number of iterations processed in the ADMM algorithm.
- **call**: the matched call.
- **formula**: the formula supplied.

**Author(s)**

Chong Ma, <chongma8903@gmail.com>.
References


See Also

cv.smog, predict.smog, plot.smog.

Examples

```r
require(coxed)

n=50; p=100
set.seed(2018)
# generate design matrix x
s=10
x=matrix(0,n,1+2*p)
x[,1]=sample(c(0,1),n,replace = TRUE)
x[,seq(2,1+2*p,2)]=matrix(rnorm(n*p),n,p)
x[,seq(3,1+2*p,2)]=x[,seq(2,1+2*p,2)]*x[,1]

g=c(p+1,rep(1:p,rep(2,p))) # groups
v=c(0,rep(1,2*p)) # penalization status
label=c("t",rep(c("prog","pred"),p)) # type of predictor variables

# generate beta
beta=c(rnorm(13,0,2),rep(0,ncol(x)-13))
beta[c(2,4,7,9)]=0

# generate y
data1=x%*%beta
noise1=rnorm(n)
snr1=as.numeric(sqrt(var(data1)/(s*var(noise1))))
y1=data1+snr1*noise1
lfit1=smog(x,y1,g,v,label,lambda1=8,lambda2=0,lambda3=8,family = "gaussian")

## generate binomial data
prob=exp(as.matrix(x)%*%as.matrix(beta))/(1+exp(as.matrix(x)%*%as.matrix(beta)))
y2=ifelse(prob<0.5,0,1)
lfit2=smog(x,y2,g,v,label,lambda1=0.03,lambda2=0,lambda3=0.03,family = "binomial")

## generate survival data
data3=sim.survdata(N=n,T=100,X=x,beta=beta)
y3=data3$data[,c("y","failed")]
y3$failed=ifelse(y3$failed,1,0)
colnames(y3)=c("time","status")
lfit3=smog(x,y3,g,v,label,lambda1=0.2,lambda2=0,lambda3=0.2,family = "coxph")
```
Index

cv.smog, 2, 6–10, 15, 16

foreach, 3
formula, 13

glog, 4

makeCluster, 3

par, 8
penalty, 6
plot, 8
plot.default, 8
plot.smog, 3, 6, 7, 9, 10, 16
predict.smog, 3, 6–8, 8, 10, 16
prox, 9
proxL1, 10
proxL2, 11

registerDoParallel, 3

sim_rct_biomarker, 11
smog, 5, 7–9
smog.default, 3, 6–10, 13
smog.formula, 3, 6–10
smog.formula(smog.default), 13
Surv, 2, 4, 13, 14