

Package ‘glmlep’

July 22, 2025

Type Package
Title Fit GLM with LEP-Based Penalized Maximum Likelihood
Version 0.2
Date 2019-03-06
Author Canhong Wen, Hao Lin, Xueqin Wang
Maintainer Canhong Wen <wencanhong@gmail.com>
Suggests mvtnorm
Description Efficient algorithms for fitting regularization paths for linear or logistic regression models penalized by LEP.
License GPL-2
NeedsCompilation yes
Repository CRAN
Date/Publication 2019-03-08 15:10:10 UTC

Contents

glmlep-package	2
cv.glmlep	3
glmlep	5
loglike	7
SetLambda	8
soft	9
Index	10

glmlep-package*Regularization paths for LEP-penalized regression models*

Description

Efficient algorithms for fitting regularization paths for linear or logistic regression models penalized by LEP.

Details

Package: glmlep
Type: Package
Version: 0.1
Date: 2013-06-05
License: GPL-2

Accepts a design matrix X and vector of responses y , produces the regularization path over a grid of values for the tuning parameter λ . Also provides methods for plotting and for determining locally convex regions of the coefficients paths.

Author(s)

Canhong Wen, Hao Lin, Shaoli Wang and Xueqin Wang.

Maintainer: Canhong Wen <wencanhong@gmail.com>

References

Wen, C., Wang, X., & Wang, S. (2013). Laplace Error Penalty based variable selection in ultra high-dimension. In press.

Examples

```
## generate data
require(mvtnorm)
n <- 100;
beta <- c(3,1.5,0,0,2,0,0,0)

set.seed(100)
p <- length(beta);
corr_data <- diag(rep(1,p));

x <- as.matrix(rmvnorm(n,rep(0,p),corr_data))
noise <- rnorm(n)

y <- tcrossprod(x,t(beta)) + noise;
fit <- glmlep(x,y,family="gaussian")
```

cv.glmlep

*Cross-validation for glmlep.***Description**

Does k-fold cross-validation for glmlep, produces a plot, and returns a value for lambda.

Usage

```
cv.glmlep(x, y, family = c("gaussian", "binomial"), lambda = NULL,
lambda.min = ifelse(n < p, 0.05, 0.001), nlambda = 100, lambda2 = 0,
kappa = ifelse(n < p, 0.1, 0.05), pen.fac = rep(1, p), tol = 1e-06,
max.ite = 1000, foldid, nfolds = 5, cv.seed = 100)
```

Arguments

x	The design matrix, without an intercept.
y	The response vector. Quantitative for family="gaussian". For family="binomial" should be a vector with two levels.
family	Response type (see above)
lambda	A user supplied lambda sequence. Typical usage is to have the program compute its own lambda sequence based on nlambda and lambda.min.ratio. Supplying a value of lambda overrides this. WARNING: use with care. Do not supply a single value for lambda. Supply instead a decreasing sequence of lambda values. glmnet relies on its warm starts for speed, and its often faster to fit a whole path than compute a single fit.
lambda.min	Smallest value for lambda, as a fraction of lambda.max, the (data derived) entry value (i.e. the smallest value for which all coefficients are zero). The default depends on the sample size nobs relative to the number of variables nvars. If nobs > nvars, the default is 0.001, close to zero. If nobs < nvars, the default is 0.05.
nlambda	The number of lambda values; default is 100.
lambda2	The tuning parameter for additional L ₂ penalty. Use for better grouping effect. The default is 0.
kappa	The scale tuning parameter of the LEP penalty. One can specify it to get the desired estimates because of the homotopy of LEP function to the L ₀ function. If nobs > nvars, the default is 0.05, close to zero. If nobs < nvars, the default is 0.1.
pen.fac	Separate penalty factors can be applied to each coefficient. This is a number that multiplies lambda to allow differential shrinkage. Can be 0 for some variables, which implies no shrinkage, and that variable is always included in the model. Default is 1 for all variables (and implicitly infinity for variables listed in exclude). Note: the penalty factors are internally rescaled to sum to nobs, and the lambda sequence will reflect this change.

tol	Convergence tolerance for MCD. Each inner MCD loop continues until the change in the estimates is less than tol. default is 1E-6.
max.ite	Maximum number of passes over the data for all lambda values; default is 10^3.
foldid	An optional vector of values between 1 and nfold identifying what fold each observation is in. If supplied, nfold can be missing.
nfolds	Number of folds - default is 5.
cv.seed	The seed for cross-validation. This could be used for simulation replicability.

Details

The function runs glmlep nfolds+1 times; the first to get the lambda sequence and the final estimate, and then the remainder to compute the fit with each of the folds omitted. The loss is accumulated, and the average loss over the folds is computed. Note that cv.glmlep does NOT search for values for kappa. A specific value should be supplied, else kappa=0.05 is assumed by default. If users would like to cross-validate kappa as well, they should call cv.glmlep with a pre-computed vector foldid, and then use this same fold vector in separate calls to cv.glmlep with different values of kappa. Note that n is the sample size and p is the dimension of variables.

Value

An object of class "cv.glmlep" is returned, which is a list with the ingredients of the cross-validation fit.

beta	A nrow(x) x length(lambda) matrix of estimated coefficient.
lambda	The sequence of regularization parameter values used
df	The degree of freedom for each value of lambda.
loss	The -2*log-likelihood value for each value of lambda.
lambda.min	The value of lambda with the minimum EBIC.
beta.min	The coefficient with the minimum EBIC.
call	The call that produces this object

Author(s)

Canhong Wen, Hao Lin, Shaoli Wang and Xueqin Wang.

Maintainer: Canhong Wen <wencanhong@gmail.com>

References

Wen, C., Wang, X., & Wang, S. (2013). Laplace Error Penalty based variable selection in ultra high-dimension. In press.

Examples

```
## generate data from multivariate normal distribution
require(mvtnorm)
n = 100;
beta <- c(3,1.5,0,0,2,0,0,0)

set.seed(100)
p=length(beta);
corr_data=diag(rep(1,p));
x=as.matrix(rmvnorm(n,rep(0,p),corr_data))
noise=rnorm(n);

## Gaussian
y <- tcrossprod(x,t(beta)) + noise;
fit <- cv.glmlep(x,y,family="gaussian")
```

glmlep

Fit a GLM wit LEP regularization

Description

Fit a generalized linear model via penalized maximum likelihood. The regularization path is computed for the LEP penalty at a grid of values for the regularization parameter lambda. Fits linear, logistic and Cox regression models.

Usage

```
glmlep(x, y, family = c("gaussian", "binomial"), lambda = NULL,
lambda.min = ifelse(n < p, 0.05, 0.001), nlambda = 100, lambda2 = 0,
kappa = ifelse(n < p, 0.1, 0.05), pen.fac = rep(1, p), tol = 1e-06,
max.ite = 1000)
```

Arguments

x	The design matrix, without an intercept.
y	The response vector. Quantitative for family="gaussian". For family="binomial" should be a vector with two levels.
family	Response type (see above)
lambda	A user supplied lambda sequence. Typical usage is to have the program compute its own lambda sequence based on nlambda and lambda.min.ratio. Supplying a value of lambda overrides this. WARNING: use with care. Do not supply a single value for lambda. Supply instead a decreasing sequence of lambda values. glmnet relies on its warm starts for speed, and its often faster to fit a whole path than compute a single fit.

lambda.min	Smallest value for lambda, as a fraction of lambda.max, the (data derived) entry value (i.e. the smallest value for which all coefficients are zero). The default depends on the sample size nobs relative to the number of variables nvars. If nobs > nvars, the default is 0.001, close to zero. If nobs < nvars, the default is 0.05.
nlambda	The number of lambda values; default is 100.
lambda2	The tuning parameter for additional L ₂ penalty. Use for better grouping effect. The default is 0.
kappa	The scale tuning parameter of the LEP penalty. One can specify it to get the desired estimates because of the homotopy of LEP function to the L ₀ function. If nobs > nvars, the default is 0.05, close to zero. If nobs < nvars, the default is 0.1.
pen.fac	Separate penalty factors can be applied to each coefficient. This is a number that multiplies lambda to allow differential shrinkage. Can be 0 for some variables, which implies no shrinkage, and that variable is always included in the model. Default is 1 for all variables (and implicitly infinity for variables listed in exclude). Note: the penalty factors are internally rescaled to sum to nobs, and the lambda sequence will reflect this change.
tol	Convergence tolerance for MCD. Each inner MCD loop continues until the change in the estimates is less than tol. default is 1E-6.
max.ite	Maximum number of passes over the data for all lambda values; default is 10^3.

Details

The sequence of models implied by lambda is fit by a modified version of coordinate descent (MCD), see reference below. Note that n is the sample size and p is the dimension of variables.

Value

An object of class "glmlep", "*", where "*" is "gaulep" or "binlep" for the two types of models.

beta	A nrow(x) x length(lambda) matrix of estimated coefficient.
lambda	The sequence of regularization parameter values used
df	The degree of freedom for each value of lambda.
loss	The -2*log-likelihood value for each value of lambda.
EBIC	The EBIC value for each value of lambda. Note tha the EBIC value is defined as
lambda.min	The value of lambda with the minimum EBIC.
beta.min	The coefficient with the minimum EBIC.
call	The call that produces this object

Author(s)

Canhong Wen, Hao Lin. <wencanhong@gmail.com>

Maintainer: Canhong Wen <wencanhong@gmail.com>

References

Wen, C., Wang, X., & Wang, S. (2013). Laplace Error Penalty based variable selection in ultra high-dimension. In press.

Examples

```
## generate data
require(mvtnorm)
n = 100;
beta <- c(3,1.5,0,0,2,0,0,0)

set.seed(100)
p=length(beta);
corr_data=diag(rep(1,p));

x=as.matrix(rmvnorm(n,rep(0,p),corr_data))
noise=rnorm(n);

## Gaussian
y <- tcrossprod(x,t(beta)) + noise;
fit <- glmlep(x,y,family="gaussian")
```

loglike

Internal glmlep functions

Description

Internal glmlep functions

Usage

```
loglike(x, y, beta, family = c("gaussian", "binomial"))
```

Arguments

x	The design matrix, without an intercept.
y	The response vector. Quantitative for family="gaussian". For family="binomial" should be a vector with two levels.
beta	The estimated coefficients.
family	Response type (see above)

Details

These are not intended for use by users.

Author(s)

Canhong Wen, Hao Lin, Shaoli Wang and Xueqin Wang.

Maintainer: Canhong Wen <wencanhong@gmail.com>

SetLambda

Internal glmlep functions

Description

Internal glmlep functions

Usage

```
SetLambda(x, y, lambda.min, nlambda, penalty.factor)
```

Arguments

x	The design matrix, without an intercept.
y	The response vector. Quantitative for family="gaussian". For family="binomial" should be a vector with two levels.
lambda.min	Smallest value for lambda, as a fraction of lambda.max, the (data derived) entry value (i.e. the smallest value for which all coefficients are zero). The default depends on the sample size nobs relative to the number of variables nvars. If nobs > nvars, the default is 0.001, close to zero. If nobs < nvars, the default is 0.05.
nlambda	The number of lambda values; default is 100.
penalty.factor	Separate penalty factors can be applied to each coefficient. This is a number that multiplies lambda to allow differential shrinkage.

Details

These are not intended for use by users.

Author(s)

Canhong Wen, Hao Lin, Shaoli Wang and Xueqin Wang.

Maintainer: Canhong Wen <wencanhong@gmail.com>

`soft`*Internal glmlep functions*

Description

Internal glmlep functions

Usage

```
soft(z, lambda)
```

Arguments

<code>z</code>	The partial least square estimate.
<code>lambda</code>	The tuning parameter.

Details

These are not intended for use by users.

Author(s)

Canhong Wen, Hao Lin, Shaoli Wang and Xueqin Wang.
Maintainer: Canhong Wen <wencanhong@gmail.com>

Index

- * **Cross-validation**

- cv.glmlep, [3](#)

- * **EBIC**

- glmlep, [5](#)

- * **GLM**

- cv.glmlep, [3](#)

- glmlep, [5](#)

- * **LEP**

- cv.glmlep, [3](#)

- glmlep, [5](#)

cv.glmlep, [3](#)

glmlep, [5](#)

glmlep-package, [2](#)

loglike, [7](#)

SetLambda, [8](#)

soft, [9](#)