Package 'gamsel'

April 1, 2018

Type Package

Title Fit Regularization Path for Generalized Additive Models

Version 1.8-1

Date 2018-03-31

Author Alexandra Chouldechova [aut, cre],
Trevor Hastie [aut, cre],
Vitalie Spinu [ctb]

Maintainer Trevor Hastie <hastie@stanford.edu>

Description Using overlap grouped-lasso penalties, 'gamsel' selects whether a term in a 'gam' is nonzero, linear, or a non-linear spline (up to a specified max df per variable). It fits the entire regularization path on a grid of values for the overall penalty lambda, both for gaussian and binomial families.

License GPL-2

Depends foreach, mda

URL http://arxiv.org/abs/1506.03850

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-04-01 03:23:38 UTC

R topics documented:

gamsel-package *gamsel*

Description

Using overlap grouped lasso penalties, gamsel selects whether a term in a gam is nonzero, linear, or a non-linear spline (up to a specified max df per variable). It fits the entire regularization path on a grid of values for the overall penalty lambda, both for gaussian and binomial families.

Details

Package:	gamsel
Type:	Package
Version:	1.0
Date:	2015-06-05
License:	What license is it under?

Accepts `x, y` data and complexity/tuning parameters. Key functions: `gamsel`
`predict.gamsel`
`plot.gamsel`
`print.gamsel`
`summary.gamsel`
`cv.gamsel`
`plot.cv.gamsel`

Author(s)

Alexandra Chouldechova, Trevor Hastie Maintainer: Trevor Hastie <hastie@stanford.edu>

---

basis.gen *Generate pseudo-spline bases*

Description

Generate an approximation to the Demmler-Reinsch orthonormal bases for smoothing splines, using orthogonal polynomials. `basis.gen` generates a basis for a single `x`, and `pseudo.bases` generates a list of bases for each column of the matrix `x`.

Usage

```
basis.gen(x, df = 6, thresh = 0.01, degree = 8, parms = NULL,...)
pseudo.bases(x, degree = 8, df = 6, parallel=FALSE, ...)
```

Arguments

x	A vector of values for basis.gen, or a matrix for pseudo.bases
df	The degrees of freedom of the smoothing spline.
thresh	If the next eigenvector improves the approximation by less than threshold, a truncated bases is returned. For pseudo.bases this can be a single value or a vector of values, which are recycled sequentially for each column of x
degree	The nominal number of basis elements. The basis returned has no more than degree columns. For pseudo.bases this can be a single value or a vector of values, which are recycled sequentially for each column of x
parms	A parameter set. If included in the call, these are used to define the basis. This is used for prediction.
parallel	For pseudo.bases, allows for parallel bases computation in multiple cores.
...	other arguments for basis.gen can be passed through pseudo.bases

Details

basis.gen starts with a basis of orthogonal polynomials of total degree. These are each smoothed using a smoothing spline, which allows for a one-step approximation to the Demmler-Reinsch basis for a smoothing spline of rank equal to the degree. See the reference for details. The function also approximates the appropriate diagonal penalty matrix for this basis, so that the a approximate smoothing spline (generalized ridge regression) has the target df.

Value

An orthonormal basis is returned (a list for pseudo.bases). This has an attribute parms, which has elements coefsCoefficients needed to generate the orthogonal polynomials rotateTransformation matrix for transforming the polynomial basis dpenalty values for the diagonal penalty dfdf used degreenumber of columns

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

T. Hastie *Pseudosplines*. (1996) JRSSB 58(2), 379-396.
Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
## Not run:
     require(doMC)
     registerDoMC(cores=4)
     bases=pseudo.bases(X,degree=10,df=6,parallel=TRUE)
```

```
## End(Not run)
```

`cv.gamsel` *Cross-validation Routine for Gamsel*

Description

A routine for performing K-fold cross-validation for gamsel.

Usage

```
cv.gamsel(x, y, lambda, family, degrees, dfs, bases,
          type.measure =c("mse", "mae", "deviance", "class"),
          nfolds = 10, foldid, keep = FALSE, parallel = FALSE, ...)
```

Arguments

`x`	x matrix as in `gamsel`
`y`	response y as in `gamsel`
`lambda`	Optional use-supplied lambda sequence. If `NULL`, default behaviour is for `gamsel` routine to automatically select a good lambda sequence.
`family`	`family` as in `gamsel`
`degrees`	`degrees` as in `gamsel`
`dfs`	`dfs` as in `gamsel`
`bases`	`bases` as in `gamsel`
`type.measure`	Loss function for cross-validated error calculation. Currently there are four options: `mse` (mean squared error), `mae` (mean absolute error), `deviance` (deviance, same as `mse` for `family="gaussian"`), `class` (misclassification error, for use with `family="binomial"`).
`nfolds`	Numer of folds (default is 10). Maximum value is `nobs`. Small values of `nfolds` are recommended for large data sets.
`foldid`	Optional vector of length `nobs` with values between 1 and `nfolds` specifying what fold each observation is in.
`keep`	If `keep=TRUE`, a *prevalidated* array is returned containing fitted values for each observation and each value of `lambda`. This means these fits are computed with this observation and the rest of its fold omitted. The `folid` vector is also returned. Default is keep=FALSE
`parallel`	If `TRUE`, use parallel `foreach` to fit each fold. See the example below for usage details.
`...`	Other arguments that can be passed to `gamsel`.

Details

This function has the effect of running gamsel nfolds+1 times. The initial run uses all the data and gets the lambda sequence. The remaining runs fit the data with each of the folds omitted in turn. The error is accumulated, and the average error and standard deviation over the folds is computed. Note that cv.gamsel does NOT search for values for alpha. A specific value should be supplied, else alpha=1 is assumed by default. If users would like to cross-validate alpha as well, they should call cv.gamsel with a pre-computed vector foldid, and then use this same fold vector in separate calls to cv.gamsel with different values of alpha. Note also that the results of cv.gamsel are random, since the folds are selected at random. Users can reduce this randomness by running cv.gamsel many times, and averaging the error curves.

Value

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

lambda	the values of lambda used in the fits.
cvm	The mean cross-validated error - a vector of length length(lambda).
cvsd	estimate of standard error of cvm.
cvup	upper curve = cvm+cvsd.
cvlo	lower curve = cvm-cvsd.
nzero	number of non-zero coefficients at each lambda.
name	a text string indicating type of measure (for plotting purposes).
gamsel.fit	a fitted gamsel object for the full data.
lambda.min	value of lambda that gives minimum cvm.
lambda.1se	largest value of lambda such that error is within 1 standard error of the minimum.
fit.preval	if keep=TRUE, this is the array of prevalidated fits. Some entries can be NA, if that and subsequent values of lambda are not reached for that fold
foldid	if keep=TRUE, the fold assignments used
index.min	the sequence number of the minimum lambda.
index.1se	the sequence number of the 1se lambda value.

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

See Also

gamsel, plot function for cv.gamsel object.

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
par(mfrow=c(1,2),mar=c(5,4,3,1))
summary(gamsel.out)
gamsel.cv=cv.gamsel(X,y,bases=bases)
par(mfrow=c(1,1))
plot(gamsel.cv)
par(mfrow=c(3,4))
plot(gamsel.out,newx=X,index=20)
```

---

gamsel *Fit Regularization Path for Gaussian or Binomial Generalized Additive Model*

Description

Using overlap grouped lasso penalties, gamsel selects whether a term in a gam is nonzero, linear, or a non-linear spline (up to a specified max df per variable). It fits the entire regularization path on a grid of values for the overall penalty lambda, both for gaussian and binomial families.

Usage

```
gamsel(x, y, num_lambda = 50, lambda = NULL, family = c("gaussian",
"binomial"), degrees = rep(10, p), gamma = 0.4, dfs = rep(5, p),
bases = pseudo.bases(x, degrees, dfs, parallel=parallel, ...),
 tol = 1e-04, max_iter = 2000, traceit = FALSE, parallel=FALSE, ...)
```

Arguments

x	Input (predictor) matrix of dimension nobs x nvars. Each observation is a row.
y	Response variable. Quantitative for family="gaussian" and with values in {0,1} for family="binomial"
num_lambda	Number of lambda values to use. (Length of lambda sequence.)
lambda	User-supplied lambda sequence. For best performance, leave as NULL and allow the routine to automatically select lambda. Otherwise, supply a (preferably gradually) decreasing sequence.
family	Response type. "gaussian" for linear model (default). "binomial" for logistic model.
degrees	An integer vector of length nvars specifying the maximum number of spline basis functions to use for each variable.
gamma	Penalty mixing parameter $0 \leq \gamma \leq 1$. Values $\gamma < 0.5$ penalize linear fit less than non-linear fit. The default is $\gamma = 0.4$, which encourages a linear term over a nonlinear term.

dfs	Numeric vector of length nvars specifying the maximum (end-of-path) degrees of freedom for each variable.
bases	A list of orthonormal bases for the non-linear terms for each variable. The function pseudo.bases generates these, using the parameters dfs and degrees. See the documentation for pseudo.bases.
tol	Convergence threshold for coordinate descent. The coordinate descent loop continues until the total change in objective after a pass over all variables is less than tol. Default is 1e-4.
max_iter	Maximum number of coordinate descent iterations over all the variables for each lambda value. Default is 2000.
traceit	If TRUE, various information is printed during the fitting process.
parallel	passed on to the pseudo.bases() function. Uses multiple process if available.
...	additional arguments passed on to pseudo.bases()

Details

The sequence of models along the lambda path is fit by (block) cordinate descent. In the case of logistic regression the fitting routine may terminate before all num_lambda values of lambda have been used. This occurs when the fraction of null deviance explained by the model gets too close to 1, at which point the fit becomes numerically unstable. Each of the smooth terms is computed using an approximation to the Demmler-Reinsch smoothing spline basis for that variable, and the accompanying diagonal pernalty matrix.

Value

An object with S3 class gamsel.

intercept	Intercept sequence of length num_lambda
alphas	nvars x num_lambda matrix of linear coefficient estimates
betas	sum(degrees) x num_lambda matrix of non-linear coefficient estimates
lambdas	The sequence of lambda values used
degrees	Number of basis functions used for each variable
parms	A set of parameters that capture the bases used. This allows for efficient generation of the bases elements for predict.gamsel

, the predict method for this class.

family	"gaussian" or "binomial"
nulldev	Null deviance (deviance of the intercept model)
dev.ratio	Vector of length num_lambda giving fraction of (null) deviance explained by each model along the lambda sequence
call	The call that produced this object

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*, http://arxiv.org/abs/1506.03850

See Also

predict.gamsel, cv.gamsel, plot.gamsel, summary.gamsel, basis.gen, gendata,

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
par(mfrow=c(1,2),mar=c(5,4,3,1))
summary(gamsel.out)
gamsel.cv=cv.gamsel(X,y,bases=bases)
par(mfrow=c(1,1))
plot(gamsel.cv)
par(mfrow=c(3,4))
plot(gamsel.out,newx=X,index=20)
# Binomial model
gamsel.out=gamsel(X,yb,family="binomial")
par(mfrow=c(1,2),mar=c(5,4,3,1))
summary(gamsel.out)
par(mfrow=c(3,4))
plot(gamsel.out,newx=X,index=30)
```

---

getActive *Returns active variables*

Description

Extract active variables of different kinds from a gamsel object

Usage

```
getActive(object, index, type = , EPS = 0)
```

Arguments

object	gamsel object
index	index or vector of indices at which to obtain active information. NULL returns all.
type	type of active variables to report. One of c("nonzero", "linear", "nonlinear")
EPS	threshold for what is nonzero; default is 0

Details

Returns a vector of variables indices of variables having the desired properties.

Value

vector of indices

---

plot.cv.gamsel *Plotting Routine for Gamsel Cross-Validation Object*

Description

Produces a cross-validation curve with standard errors for a fitted gamsel objecty.

Usage

```
## S3 method for class 'cv.gamsel'
plot(x, sign.lambda = 1, ...)
```

Arguments

x	cv.gamsel object
sign.lambda	Either plot against log(lambda) (default) against -lambda if sign.lambda=-1.
...	Optional graphical parameters to plot.

Details

A plot showing cross-validation error is produced. Nothing is returned.

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
gamsel.cv=cv.gamsel(X,y,bases=bases)
par(mfrow=c(1,1))
plot(gamsel.cv)
```

plot.gamsel *Plotting Routine* gamsel *Object*

Description

Produces plots of the estimated functions for specified variables at a given value of lambda.

Usage

```
## S3 method for class 'gamsel'
plot(x, newx, index, which = 1:p, rugplot = TRUE, ylims, ...)
```

Arguments

x	Fitted gamsel object.
newx	nobs_new x p matrix giving values of each predictor at which to plot.
index	Index of lambda value (i.e., model) for which plotting is desired.
which	Which values to plot. Default is all variables, i.e. {1,2,...,nvars}. Besides indices, which can take two special values: "nonzero" will plot only the nonzero functions, and "nonlinear" only the nonlinear functions.
rugplot	If TRUE, a rugplot showing values of x is shown at the bottom of each fitted function plot.
ylims	ylim argument for plotting each curve, which overides the default which is the range of all the functions.
...	Optional graphical parameters to plot.

Details

A plot of the specified fitted functions is produced. Nothing is returned.

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

See Also

gamsel, and print.gamsel, summary.gamsel

Examples

```
set.seed(1211)
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.8)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
par(mfrow=c(3,4))
plot(gamsel.out,newx=X,index=20)
```

`predict.gamsel` *Gamsel Prediction Routine*

Description

Make predictions from a `gamsel` object.

Usage

```
## S3 method for class 'gamsel'
predict(object, newdata, index = NULL,
       type = c("link", "response", "terms", "nonzero"), ...)
```

Arguments

`object`	Fitted `gamsel` object.
`newdata`	`nobs_new` x `p` matrix of new data values at which to predict.
`index`	Index of model in the sequence for which plotting is desired. Note, this is NOT a lambda value.
`type`	Type of prediction desired. Type `link` gives the linear predictors for `"binomial"`, and fitted values for `"gaussian"`. Type `response` gives fitted probabilities for `"binomial"` and fitted values for `"gaussian"`. Type `"terms"` returns a matrix of fitted functions, with as many columns as there are variables. Type `nonzero` returns a list of the indices of nonzero coefficients at the given `lambda` index.
`...`	Not used

Value

Either a vector aor a matrix is returned, depending on `type`.

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

See Also

gamsel, cv.gamsel, summary.gamsel, basis.gen, gendata,

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
preds=predict(gamsel.out,X,index=20,type="terms")
```

---

print.gamsel *print a gamsel object*

Description

Print a summary of the gamsel path at each step along the path

Usage

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

Arguments

x	fitted gamsel object
digits	significant digits in printout
...	additional print arguments

Details

The call that produced the object x is printed, followed by a five-column matrix with columns NonZero, Lin, NonLin, %Dev and Lambda. The first three columns say how many nonzero, linear and nonlinear terms there are. %Dev is the percent deviance explained (relative to the null deviance).

Value

The matrix above is silently returned

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

See Also

predict.gamsel, cv.gamsel, plot.gamsel, summary.gamsel, basis.gen, gendata,

---

summary.gamsel *Gamsel summary routine*

Description

This makes a two-panel plot of the gamsel object.

Usage

```
## S3 method for class 'gamsel'
summary(object, label, ...)
```

Arguments

object	gamsel object
label	if TRUE, annotate the plot with variable labels. Default is FALSE
...	additional arguments to summary

Details

A two panel plot is produced, that summarizes the linear components and the nonlinear components, as a function of lambda. For the linear components, it is the coefficient for each variable. For the nonlinear, we see the norm of the nonlinear coefficients.

Value

Nothing is returned.

Author(s)

Alexandra Chouldechova and Trevor Hastie
Maintainer: Trevor Hastie <hastie@stanford.edu>

References

Chouldechova, A. and Hastie, T. (2015) *Generalized Additive Model Selection*

See Also

gamsel, and methods plot, print and predict for cv.gamsel object.

Examples

```
data=gendata(n=500,p=12,k.lin=3,k.nonlin=3,deg=8,sigma=0.5)
attach(data)
bases=pseudo.bases(X,degree=10,df=6)
# Gaussian gam
gamsel.out=gamsel(X,y,bases=bases)
par(mfrow=c(1,2),mar=c(5,4,3,1))
summary(gamsel.out)
```

Index