Description

Package for Bayesian Model Averaging in linear models using stochastic or deterministic sampling without replacement from posterior distributions. Prior distributions on coefficients are of the form of Zellner’s g-prior or mixtures of g-priors. Options include the Zellner-Siow Cauchy Priors, the Liang et al hyper-g priors, Local and Global Empirical Bayes estimates of g, and other default model selection criteria such as AIC and BIC. Sampling probabilities may be updated based on the sampled models.

Details

Package: BAS
Version: 1.0
Date: 2005-12-21
Depends: R (>= 2.1)
License: GPL version 2 or newer
URL: http://www.isds.duke.edu/~clyde
Built: R 2.2.0; i686-redhat-linux-gnu; 2005-12-27 14:20:12; unix

Author(s)

Merlise Clyde and Michael Littman,
Maintainer: Merlise Clyde <clyde@stat.duke.edu>

References


http://www.stat.duke.edu/05-12.pdf

See Also

bas
Examples

demo(BAS.USCrime)
demo(BAS.hald)

---

**EB.global**  
*Finds the global Empirical Bayes estimates for BMA*

---

**Description**

Finds the global Empirical Bayes estimates of g in Zellner’s g-prior and model probabilities

**Usage**

```r
EB.global.bma(object, tol = .1, g.0=NULL, max.iterations=100)
```

**Arguments**

- `object`: A 'bma' object created by `bas`
- `tol`: tolerance for estimating g
- `g.0`: initial value for g
- `max.iterations`: Maximum number of iterations for the EM algorithm

**Details**

Uses the EM algorithm in Liang et al to estimate the type II MLE of g in Zellner’s g prior

**Value**

An object of class 'bma' using Zellner’s g prior with an estimate of g based on all models

**Author(s)**

Merlise Clyde (clyde@stat.duke.edu)

**References**


**See Also**

- `bas`, `update`
Examples

```r
library(MASS)
data(UScrime)
UScrime[,-2] = log(UScrime[,-2])
# EB local uses a different g within each model
crime.EBL = bas.lm(y ~ ., data=UScrime, n.models=2^15,
  prior="EB-local", initprobs= "eplogp")
# use a common (global) estimate of g
crime.EBG = EB.global.bma(crime.EBL)
```

---

**as.matrix.bma**

*Coerce a BMA list object into a matrix*

---

**Description**

Models, coefficients, and standard errors in objects of class `bma` are represented as a list of lists to reduce storage by omitting the zero entries. These functions coerce the list object to a matrix and fill in the zeros to facilitate other computations.

**Usage**

```r
## S3 method for class 'bma':
as.matrix(x, what, which.models=NULL)
```

```r
## S3 method for class 'which':
as.matrix(x, which.models=NULL)
```

```r
which.matrix(which, n.vars)
```

**Arguments**

- `x` a 'bma' object
- `what` name of bma list to coerce
- `which.models` a vector of indices use to extract a subset
- `which` `x$which` a list of lists of model indicators
- `n.vars` the total number of predictors, `x$n.vars`

**Details**

`as.matrix.bma(x, which)` is equivalent to `as.matrix.which(x)`, however, the latter uses `sapply` rather than a loop. `as.matrix.which` and `which.matrix` both coerce `x$which` into a matrix.
Value

A matrix representation of `x$what`, with number of rows equal to the length of `which.models` or total number of models and number of columns `x$n.vars`

Author(s)

Merlise Clyde (clyde@stat.duke.edu)

See Also

bas

Examples

```r
library(MASS)
data(UScrime)
UScrime[,2] = log(UScrime[,2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC",
                   initprobs = "eplogp")
coef = as.matrix.bma(crime.bic, "ols")  # extract all ols coefficients
se = as.matrix.bma(crime.bic, "ols.se")
models = as.matrix.which(crime.bic)      # matrix of model indicators
models = which.matrix(crime.bic$which, crime.bic$n.vars)  # matrix of model indicators
```

bas.lm

Bayesian Adaptive Sampling Without Replacement for Variable Selection in Linear Models

Description

Sample without replacement from a posterior distribution on models

Usage

```r
bas.lm(formula, data, n.models, alpha=NULL,
       prior="ZS", initprobs="Uniform", random=TRUE, update=NULL,
       bestmodel = NULL, bestmarg = NULL, prob.local = 0)
```

Arguments

- **formula**: linear model formula for the full model with all predictors, `Y ~ X`. All code assumes that an intercept will be included in each model.
- **data**: data frame
- **n.models**: number of models to sample
initprobs  vector of initial marginal inclusion probabilities used for sampling without replacement or method, if "Uniform" each predictor variable is equally likely to be sampled (equivalent to random sampling without replacement). If "eplogp", use the function to approximate the Bayes factor to find initial marginal inclusion probabilities and sample without replacement the model probabilities using these inclusion probabilities.

alpha  optional hyperparameter in g-prior or hyper g-prior. For Zellner's g-prior, alpha = g, for the Liang et al hyper-g method, recommended choice is alpha = 3 or 4.

prior  prior distribution for regression coefficients. Choices include "AIC", "BIC", "g-prior", "ZS-null", "ZS-full", "hyper-g", "hyper-g-laplace", "EB-local", and "EB-global"

random  A logical variable indicating whether to use the stochastic (random=TRUE) or deterministic (random=FALSE) algorithm for sampling models without replacement

update  how often to update sampling probabilities

bestmodel  optional binary vector representing a model to initialize the sampling. If NULL sampling starts with the Full model

bestmarg  optional value for the log marginal associated with the bestmodel

prob.local  An experimental option to allow sampling of models "near" the median probability model. Not recommended for use at this time

Details

BAS provides two search algorithms to find high probability models for use in Bayesian Model Averaging or Bayesian model selection.

Value

bas returns an object of class BMA

An object of class BMA is a list containing at least the following components:

postprob  the posterior probabilities of the models selected

namesx  the names of the variables

R2  R2 values for the models

logmarg  values of the log of the marginal likelihood for the models

n.vars  total number of independent variables in the full model, including the intercept

size  the number of independent variables in each of the models, includes the intercept

which  a list of lists with one list per model with variables that are included in the model

probne0  the posterior probability that each variable is non-zero

ols  list of lists with one list per model giving the OLS estimate of each (nonzero) coefficient for each model
ols.se list of lists with one list per model giving the OLS standard error of each coefficient for each model
prior the name of the prior that created the BMA object
alpha value of hyperparameter in prior used to create the BMA object.
Y response
X matrix of predictors

The function summary.bma, is used to print a summary of the results. The function plot.bma is used to plot posterior distributions for the coefficients and image.bma provides an image of the distribution over models. Posterior summaries of coefficients can be extracted using coefficients.bma. Fitted values and predictions can be obtained using the functions fitted.bma and predict.bma. BMA objects may be updated to use a different prior (without rerunning the sampler) using the function update.bma.

Note
Uniform prior probabilities on models are the only option currently. A future update should allow alternative priors on models to be incorporated into the sampling and posterior inference. For now, users may manually reweight output using the log marginal likelihoods to update posterior model probabilities and probne0.

Author(s)
Merlise Clyde ((clyde@stat.duke.edu)) and Michael Littman

References
http://www.stat.duke.edu/05-12.pdf
bin2int  

Convert binary model representation into an integer  

Description  
Takes a binary string representation of a model and converts to an integer  

Usage  
bin2int(model)  

Arguments  
  model  
a Boolean/binary vector of length \( p \) representing a model  

Details  
Used in \texttt{fitted.bma} to determine if the median probability model is included in the sample. Not meant to be used directly by the user. On a 32 bit system, \( p \) must be less than or equal to 32.  

Value  
an integer  

Author(s)  
Merlise Clyde (clyde@stat.duke.edu)
Coef.bma

Coefficients of a Bayesian Model Average object

Description

Extract conditional posterior means and standard deviations, marginal posterior means and standard deviations, posterior probabilities, and marginal inclusions probabilities under Bayesian Model Averaging from an object of class BMA.

Usage

```r
## S3 method for class 'bma':
coef(object, ...)
## S3 method for class 'coef.bma':
print(x, n.models=5, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

- `object` object of class 'bma' created by BAS
- `x` object of class 'coef.bma' to print
- `n.models` Number of top models to report in the printed summary
- `digits` number of significant digits to print
- `...` other optional arguments

Details

Calculates posterior means and (approximate) standard deviations of the regression coefficients under Bayesian Model averaging using g-priors and mixtures of g-priors. Print returns overall summaries. For fully Bayesian methods that place a prior on g, the posterior standard deviations do not take into account full uncertainty regarding g. Will be updated in future releases.

Value

`coefficients` returns an object of class coef.bma with the following:

- `conditionalmeans`
- `conditionalsd` standard deviations for each model
- `postmean` marginal posterior means of each regression coefficient using BMA
- `postsd` marginal posterior standard deviations using BMA
- `postne0` vector of posterior inclusion probabilities, marginal probability that a coefficient is non-zero
Note
With highly correlated variables, marginal summaries may not be representative of the
distribution. Use \texttt{plot.coef.bma} to view distributions.

Author(s)
Merlise Clyde (clyde@stat.duke.edu)

References
for Bayesian Variable Selection.
\url{http://www.stat.duke.edu/05-12.pdf}

See Also
\texttt{bas}

Examples
\begin{verbatim}
data("Hald")
hald.gprior = bas.lm(Y ~ ., data=Hald, n.models=2^4, alpha=13,
                     prior="ZS-null", initprobs="Uniform", update=10)
coef.hald.gprior = coefficients(hald.gprior)
coef.hald.gprior
plot(coef.hald.gprior)
\end{verbatim}

\texttt{cv.summary.bma} \hspace{1cm} \textit{Summaries for Out of Sample Prediction}

Description
Compute summaries from out of sample predictions for a BMA object

Usage
\texttt{cv.summary.bma(object, pred, ytrue)}

Arguments
\begin{itemize}
  \item \texttt{object} \hspace{1cm} an object of class ’bma’
  \item \texttt{pred} \hspace{1cm} output from \texttt{predict.bma}
  \item \texttt{ytrue} \hspace{1cm} vector of left out response values
\end{itemize}

Value
A matrix with the best models, posterior probabilities, R2, dimensions, Average Prediction
Error from the HPM and Average prediction error for BMA prediction
eplogprob - Compute approximate marginal inclusion probabilities from p-values

Description

eplogprob calculates approximate marginal posterior inclusion probabilities from p-values computed from a linear model using a lower bound approximation to Bayes factors. Used to obtain initial inclusion probabilities for sampling using Bayesian Adaptive Sampling bas.lm

Usage

eplogprob(lm.obj, thresh=.5, max = 0.99, int=TRUE)

Arguments

lm.obj a linear model object
thresh the value of the inclusion probability when if the p-value > 1/exp(1), where the lower bound approximation is not valid.
max maximum value of the inclusion probability; used for the bas.lm function to keep initial inclusion probabilities away from 1.
int If the Intercept is included in the linear model, set the marginal inclusion probability corresponding to the intercept to 1

Details

Sellke, Bayarri and Berger (2001) provide a simple calibration of p-values

\[ BF(p) = -e p \log(p) \]

which provide a lower bound to a Bayes factor for comparing H0: beta = 0 versus H1: beta not equal to 0, when the p-value p is less than 1/e. Using equal prior odds on the hypotheses H0 and H1, the approximate marginal posterior inclusion probability

\[ p(\beta \neq 0 \mid \text{data}) = 1/(1 + BF(p)) \]

When p > 1/e, we set the marginal inclusion probability to 0.5 or the value given by thresh.
Value

`eplogprob` returns a vector of marginal posterior inclusion probabilities for each of the variables in the linear model. If `int = TRUE`, then the inclusion probability for the intercept is set to 1. If the model is not full rank, variables that are linearly dependent based on the QR factorization will have NA for their p-values. In `bas.lm`, where the probabilities are used for sampling, the inclusion probability is set to 0.

Author(s)

Merlise Clyde (clyde@stat.duke.edu)

References


See Also

`bas`

Examples

```r
library(MASS)
data(UScrime)
UScrime[,2] = log(UScrime[,2])
eplogprob(lm(y ~ ., data=UScrime))
```

---

`fitted.bma`  
*Fitted values for a BMA objects*

Description

Calculate fitted values for a BMA object

Usage

```r
## S3 method for class 'bma':
fitted(object, type="HPM", top=NULL, ...)
```

Arguments

- `object`  
  An object of class ‘bma’ as created by `bas`

- `type`  
  type of fitted value to return. Options include  
  'HPM' the highest probability model  
  'BMA' Bayesian model averaging, using optionally only the 'top' models  
  'MPM' the median probability model of Barbieri and Berger.
top

optional argument specifying that the 'top' models will be used in constructing the BMA prediction, if NULL all models will be used. If top=1, then this is equivalent to 'HPM'

... optional arguments, not used currently

Details

Calculates fitted values at observed design matrix using either the highest probability model, 'HPM', the posterior mean (under BMA) 'BMA', or the median probability model 'MPM'. The median probability model is defined by including variable where the marginal inclusion probability is greater than or equal to 1/2. For type="BMA", the weighted average may be based on using a subset of the highest probability models if an optional argument is given for top. By default BMA uses all sampled models, which may take a while to compute if the number of variables or number of models is large.

Value

A vector of length n of fitted values.

Author(s)

Merlise Clyde (clyde@AT@stat.duke.edu)

References


See Also

 predict.bma

Examples

data(Hald)
hald.gprior = bas.lm(Y~ ., data=Hald, n.models=2^4, alpha=13, prior="ZS-null", initprobs="Uniform") plot(Hald$Y, fitted(hald.gprior, type="HPM"))
plot(Hald$Y, fitted(hald.gprior, type="BMA"))
plot(Hald$Y, fitted(hald.gprior, type="MPM"))

Hald

Hald Data
Description

The Hald data have been used in many books and papers to illustrate variable selection. The data relate to an engineering application that was concerned with the effect of the composition of cement on heat evolved during hardening. The response variable \( Y \) is the heat evolved in a cement mix. The four explanatory variables are ingredients of the mix, \( X_1 \): tricalcium aluminate, \( X_2 \): tricalcium silicate, \( X_3 \): tetracalcium alumino ferrite, \( X_4 \): dicalcium silicate. An important feature of these data is that the variables \( X_1 \) and \( X_3 \) are highly correlated, as well as the variables \( X_2 \) and \( X_4 \). Thus we should expect any subset of \((X_1,X_2,X_3,X_4)\) that includes one variable from the highly correlated pair to do as any subset that also includes the other member.

Usage

```r
data(Hald)
```

Format

\texttt{hald} is a dataframe with 13 observations and 5 variables (columns),

- \texttt{Y}: Heat evolved per gram of cement (in calories)
- \texttt{X1}: Amount of tricalcium aluminate
- \texttt{X2}: Amount of tricalcium silicate
- \texttt{X3}: Amount of tetracalcium alumino ferrite
- \texttt{X4}: Amount of dicalcium silicate

Source


---

**image.bma**

Images of models used in Bayesian model averaging

Description

Creates an image of the models selected using \texttt{bas}.

Usage

```r
## S3 method for class 'bma':
image(x, top.models=20, intensity=TRUE, prob=TRUE, log=TRUE,
      rotate=TRUE, color="rainbow", subset=NULL, offset=.75, digits=3,
      vlas=2,plas=0,rlas=0, ...)
```
Arguments

- **x**: An object of type ‘bma’ created by BAS
- **top.models**: Number of the top ranked models to plot
- **intensity**: Logical variable, when TRUE image intensity is proportional to the probability or log(probability) of the model, when FALSE, intensity is binary indicating just presence (light) or absence (dark) of a variable.
- **prob**: Logical variable indicating whether the intensities should be based on log Bayes Factors (TRUE) or posterior probabilities (FALSE). The log of the Bayes factor is for comparing the each model to the worst model in the set.
- **rotate**: Should the image of models be rotated so that models are on the y-axis and variables are on the x-axis (TRUE)
- **color**: The color scheme for image intensities. The value "rainbow" uses the rainbow palette. The value "blackandwhite" produces a black and white image (greyscale image)
- **subset**: indices of variables to include in plot; 1 is the intercept
- **offset**: numeric value to add to intensity
- **digits**: number of digits in posterior probabilities to keep
- **vlas**: las parameter for placing variable names; see par
- **plas**: las parameter for posterior probability axis
- **rlas**: las parameter for model ranks
- **...**: Other parameters to be passed to the `image` and `axis` functions.

Details

Creates an image of the model space sampled using `bas`. If a subset of the top models are plotted, then probabilities are renormalized over the subset.

Note

Suggestion to allow area of models be proportional to posterior probability due to Thomas Lumley

Author(s)

Merlise Clyde (clyde@stat.duke.edu)

References

See Also

bas

Examples

data("Hald")
hald.gprior = bas.lm(Y ~ ., data=Hald, n.models=2^4, alpha=13, prior="ZS-null", initprobs="Uniform", update=10)
image(hald.gprior, subset=-1)

plot.bma  

Plot Diagnostics for an blm Object

Description

Four plots (selectable by 'which') are currently available: a plot of residuals against fitted values, Cumulative Model Probabilities, log marginal likelihoods versus model dimension, and marginal inclusion probabilities.

Usage

## S3 method for class 'bma':
plot(x, which=c(1:4),caption = c("Residuals vs Fitted", "Model Probabilities", "Model Complexity", "Inclusion Probabilities"), panel = if (add.smooth) panel.smooth else points, sub.caption = NULL, main = "", ask = prod(par("mfcol")) < length(which) && dev.interactive(), ..., id.n = 3, labels.id = names(residuals(x)), cex.id = 0.75, add.smooth = getOption("add.smooth"), label.pos = c(4, 2))

Arguments

x blm object, typically result of 'blm'
which if a subset of the plots is required, specify a subset of the numbers '1:4'
caption captions to appear above the plots
panel panel function. The useful alternative to 'points', 'panel.smooth' can be chosen by 'add.smooth = TRUE'
sub.caption common title-above figures if there are multiple; used as 'sub' (s.'title') otherwise. If 'NULL', as by default, a possible shortened version of deparse(x$call) is used
main title to each plot-in addition to the above 'caption'
ask logical; if 'TRUE', the user is asked before each plot, see 'par(ask=.)'
... other parameters to be passed through to plotting functions
id.n
number of points to be labelled in each plot, starting with the most extreme

labels.id
vector of labels, from which the labels for extreme points will be chosen. 'NULL' uses observation numbers

cex.id
magnification of point labels.

add.smooth
logical indicating if a smoother should be added to most plots; see also 'panel' above

label.pos
positioning of labels, for the left half and right half of the graph respectively, for plots 1-3

Details

Author(s)
Merlise Clyde, based on plot.lm by John Maindonald and Martin Maechler

References

See Also

plot.coef.bma and image.bma.

Examples

plot.coef.bma
*Plots the posterior distributions of coefficients derived from Bayesian model averaging*

Description

Displays plots of the posterior distributions of the coefficients generated by Bayesian model averaging over linear regression.

Usage

```r
## S3 method for class 'coef.bma':
plot(x, e = 1e-04, subset = 1:x$n.vars, ask=TRUE,...)
```
Arguments

- **x**: object of class `coefficients.bma`
- **e**: optional numeric value specifying the range over which the distributions are to be graphed.
- **subset**: optional numerical vector specifying which variables to graph (including the intercept)
- **ask**: Prompt for next plot
- **...**: other parameters to be passed to `plot` and `lines`

Details

Produces plots of the posterior distributions of the coefficients under model averaging. The posterior probability that the coefficient is zero is represented by a solid line at zero, with height equal to the probability. The nonzero part of the distribution is scaled so that the maximum height is equal to the probability that the coefficient is nonzero.

The parameter **e** specifies the range over which the distributions are to be graphed by specifying the tail probabilities that dictate the range to plot over.

Note

For mixtures of g-priors, uncertainty in g is not incorporated at this time, thus results are approximate.

Author(s)

based on function `plot.bic` by Ian Painter in package BMA; adapted for 'bma’ class by Merlise Clyde (clyde@stat.duke.edu)

References


See Also

- `coef.bma`

Examples

```r
library(MASS)
data(UScrime)
UScrime[, -2] = log(UScrime[, -2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC")
plot(coefficients(crime.bic), ask=TRUE)
```
predict.bma  

Prediction Method for an object of class BMA

Description

Predictions under model averaging from a BMA object

Usage

```r
## S3 method for class 'bma': 
predict(object, newdata, top=NULL, ...)
```

Arguments

- `object`: An object of class BMA, created by `bas`
- `newdata`: new data for predictions
- `top`: Use only the top M models, based on posterior probabilities
- `...`: optional extra arguments

Details

Value

- a list of
  - `Ybma`: predictions using BMA
  - `Ypred`: matrix of predictions under each model
  - `best`: index of top models included

Author(s)

Merlise Clyde

References

See Also

`bas`, `fitted.bma`

Examples

```r
data("Hald")
hald.gprior = bas.lm(Y ~ ., data=Hald, n.models=2^4, alpha=13, prior="g-prior", initprobs="Uniform")
predict(hald.gprior, hald.gprior$X, top=5)
```
**summary.bma**

*Summaries of Bayesian Model Averaging objects*

**Description**

`summary` and `print` methods for Bayesian model averaging objects created by `bas` Bayesian Adaptive Sampling

**Usage**

```r
## S3 method for class 'bma'
summary(object, n.models = 5, ...)
## S3 method for class 'bma'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

**Arguments**

- `object`: object of class 'bma'
- `x`: object of class 'bma'
- `n.models`: optional number specifying the number of best models to display in summary
- `digits`: optional number specifying the number of digits to display
- `...`: other parameters to be passed to `print.default`

**Details**

The print methods display a view similar to `print.lm`. The summary methods display a view specific to Bayesian model averaging giving the top highest probability models.

**Author(s)**

Merlise Clyde (clyde@stat.duke.edu)

**See Also**

`coefficients.bma`

**Examples**

```r
library(MASS)
data(UScrime)
UScrime[,2] = log(UScrime[,2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC", initprobs= "explogp")
print(crime.bic)
summary(crime.bic)
```
update.bma  

Update BMA object using a new prior

Description

Usage

update.bma(object, newprior, alpha=NULL, ...)

Arguments

- **object**: BMA object to update
- **newprior**: Update posterior model probabilities, probne0, shrinkage, logmarg, etc, using prior based on newprior. See bas for available methods
- **alpha**: optional new value of hyperparameter in prior for method
- **...**: optional arguments

Details

Recomputes the marginal likelihoods for the new methods for models already sampled in current object.

Value

A new object of class BMA

Author(s)

Merlise Clyde (clyde@stat.duke.edu)

References


See Also

bas for available methods and choices of alpha

Examples

```r
library(MASS)
data(UScrime)
UScrime[,2] = log(UScrime[,2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC", initprobs= "eplogp")
crime.aic = update(crime.bic, newprior="AIC")
crime.zs = update(crime.bic, newprior="ZS-null")
crime.hg = update(crime.bic, newprior="hyper-g-laplace", alpha=3)
```