

Package ‘BAS’

February 14, 2012

Version 0.92

Date 2010-5-1

Title Bayesian Model Averaging using Bayesian Adaptive Sampling

Author Merlise Clyde <clyde@stat.duke.edu> with contributions from
Michael Littman and Joyee Ghosh

Maintainer Merlise Clyde <clyde@stat.duke.edu>

Depends R (>= 2.8), stats

LinkingTo stats

Imports stats

SUGGESTS MASS

Description Package for Bayesian Model Averaging in linear models using stochastic or deterministic sampling without replacement from posterior distributions. Prior distributions on coefficients are from Zellner’s g-prior or mixtures of g-priors corresponding to the Zellner-Siow Cauchy Priors or the Liang et al hyper-g priors (JASA 2008). Other model selection criterion include AIC and BIC. Sampling probabilities may be updated based on the sampled models. Allows uniform or beta-binomial prior distributions on models.

License GPL (>= 2)

URL <http://www.r-project.org>, <http://www.stat.duke.edu/~clyde/BAS>

Repository CRAN

Date/Publication 2010-10-01 08:22:34

R topics documented:

BAS-package	2
bas.lm	3
bayesglm.fit	7
Bernoulli	8
beta.binomial	9
bin2int	10
coef.bma	11
cv.summary.bma	12
EB.global	13
eplogprob	14
fitted.bma	15
g.prior	16
Hald	17
hypergeometric2F1	18
IC.prior	19
image.bma	20
list2matrix.bma	21
plot.bma	23
plot.coef.bma	24
predict.bma	25
protein	27
summary.bma	27
uniform	28
update.bma	29

Index	31
--------------	-----------

BAS-package	<i>Bayesian Model Averaging using Bayesian Adaptive Sampling</i>
-------------	--

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:	0.9
Date:	2010-5-1
Depends:	R (>= 2.8)
License:	GPL-2

URL: <http://www.stat.duke.edu/~clyde>

Index:

Author(s)

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

References

Clyde, M. Ghosh, J. and Littman, M. (2010) Bayesian Adaptive Sampling for Variable Selection and Model Averaging. *Journal of Computational and Graphical Statistics*. (to appear) Department of Statistical Science Discussion Paper 2009-16. Duke University.

Clyde, M. and George, E. I. (2004) Model uncertainty. *Statist. Sci.*, 19, 81-94.
<http://www.isds.duke.edu/~clyde/papers/statsci.pdf>

Clyde, M. (1999) Bayesian Model Averaging and Model Search Strategies (with discussion). In *Bayesian Statistics 6*. J.M. Bernardo, A.P. Dawid, J.O. Berger, and A.F.M. Smith eds. Oxford University Press, pages 157-185.

Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2008) Mixtures of g-priors for Bayesian Variable Selection. *Journal of the American Statistical Association*. 103:410-423.
<http://dx.doi.org/10.1198/016214507000001337>

See Also

[bas](#)

Examples

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

bas.lm

Bayesian Adaptive Sampling Without Replacement for Variable Selection in Linear Models

Description

Sample without replacement from a posterior distribution on models

Usage

```
bas.lm(formula, data, n.models=NULL, prior="ZS-null", alpha=NULL,
       modelprior=uniform(),
       initprobs="Uniform", method="BAS", update=NULL,
       bestmodel = NULL, bestmarg = NULL, prob.local = 0.0, prob.rw=0.5,
       Burnin.iterations = NULL, MCMC.iterations = NULL,
       lambda = NULL, delta = 0.025)
```

Arguments

formula	linear model formula for the full model with all predictors, $Y \sim X$. All code assumes that an intercept will be included in each model and that the X's will be centered.
data	data frame
n.models	number of models to sample. If NULL, BAS will enumerate unless $p > 25$
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"
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 α are between (2, 4), with $\alpha = 3$ recommended.
modelprior	Family of prior distribution on the models. Choices include uniform Bernoulli or beta.binomial
initprobs	vector of length p with the initial inclusion probabilities used for sampling without replacement (the intercept should be included with probability one) or a character string giving the method used to construct the sampling probabilities if "Uniform" each predictor variable is equally likely to be sampled (equivalent to random sampling without replacement). If "eplogp", use the eplogprob function to approximate the Bayes factor to find initial marginal inclusion probabilities and sample without replacement using these inclusion probabilities. For variables that should always be included set the corresponding <code>initprobs</code> to 1. To run a Markov Chain to provide initial estimates of marginal inclusion probabilities, use <code>method="MCMC+BAS"</code> below.
method	A character variable indicating which sampling method to use: <code>method="BAS"</code> uses Bayesian Adaptive Sampling (without replacement) using the sampling probabilities given in <code>initprobs</code> ; <code>method="MCMC+BAS"</code> runs an initial MCMC to calculate marginal inclusion probabilities and then samples without replacement as in BAS; <code>method="AMCMC"</code> runs an Adaptive MCMC (experimental). For both BAS and AMCMC, the sampling probabilities can be updated as more models are sampled. (see <code>update</code> below). We recommend "MCMC+BAS" for high dimensional problems.
update	number of iterations between potential updates of the sampling probabilities. If NULL do not update, otherwise the algorithm will update using the marginal inclusion probabilities as they change while sampling takes place. For large model spaces, updating is recommended. If the model space will be enumerated, leave at the default.

bestmodel	optional binary vector representing a model to initialize the sampling. If NULL sampling starts with the null model
bestmarg	optional value for the log marginal associated with the bestmodel
prob.local	A future option to allow sampling of models "near" the median probability model. Not used at this time.
prob.rw	For any of the MCMC methods, probability of using the random-walk proposal; otherwise use a random "flip" move to propose a new model.
Burnin.iterations	Number of iterations to discard when using any of the MCMC options
MCMC.iterations	Number of iterations to run MCMC when MCMC options are used
lambda	Parameter in the AMCMC algorithm.
delta	truncation parameter to prevent sampling probabilities to degenerate to 0 or 1.

Details

BAS provides several search algorithms to find high probability models for use in Bayesian Model Averaging or Bayesian model selection. For p less than 20-25, BAS can enumerate all models depending on memory availability, for larger p , BAS samples without replacement using random or deterministic sampling. The Bayesian Adaptive Sampling algorithm of Clyde, Ghosh, Littman (2010) samples models without replacement using the initial sampling probabilities, and will optionally update the sampling probabilities every "update" models using the estimated marginal inclusion probabilities. BAS uses different methods to obtain the `ini tprobs`, which may impact the results in high-dimensional problems. The deterministic sampler provides a list of the top models in order of an approximation of independence using the provided `ini tprobs`. This may be effective after running the other algorithms to identify high probability models and works well if the correlations of variables are small to modest. The priors on coefficients include Zellner's g-prior, the Hyper-g prior (Liang et al 2008, the Zellner-Siow Cauchy prior, Empirical Bayes (local and global) g-priors. AIC and BIC are also included.

Value

`bas` returns an object of class `BMA`

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

<code>postprob</code>	the posterior probabilities of the models selected
<code>priorprobs</code>	the prior probabilities of the models selected
<code>namesx</code>	the names of the variables
<code>R2</code>	R2 values for the models
<code>logmarg</code>	values of the log of the marginal likelihood for the models
<code>n.vars</code>	total number of independent variables in the full model, including the intercept
<code>size</code>	the number of independent variables in each of the models, includes the intercept
<code>which</code>	a list of lists with one list per model with variables that are included in the model
<code>probne0</code>	the posterior probability that each variable is non-zero

<code>ols</code>	list of lists with one list per model giving the OLS estimate of each (nonzero) coefficient for each model. The intercept is the mean of Y as each column of X has been centered by subtracting its mean.
<code>ols.se</code>	list of lists with one list per model giving the OLS standard error of each coefficient for each model
<code>prior</code>	the name of the prior that created the BMA object
<code>alpha</code>	value of hyperparameter in prior used to create the BMA object.
<code>modelprior</code>	the prior distribution on models that created the BMA object
<code>Y</code>	response
<code>X</code>	matrix of predictors
<code>mean.x</code>	vector of means for each column of X (used in <code>predict.bma</code>)

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`.

Author(s)

Merlise Clyde (<clyde@stat.duke.edu>) and Michael Littman

References

- Clyde, M. Ghosh, J. and Littman, M. (2010) Bayesian Adaptive Sampling for Variable Selection and Model Averaging. *Journal of Computational Graphics and Statistics*. (to appear) Department of Statistical Science Discussion Paper 2009-16. Duke University.
- Clyde, M. and George, E. I. (2004) Model Uncertainty. *Statist. Sci.*, 19, 81-94.
<http://dx.doi.org/10.1214/088342304000000035>
- Clyde, M. (1999) Bayesian Model Averaging and Model Search Strategies (with discussion). In *Bayesian Statistics 6*. J.M. Bernardo, A.P. Dawid, J.O. Berger, and A.F.M. Smith eds. Oxford University Press, pages 157-185.
- Hoeting, J. A., Madigan, D., Raftery, A. E. and Volinsky, C. T. (1999) Bayesian model averaging: a tutorial (with discussion). *Statist. Sci.*, 14, 382-401.
<http://www.stat.washington.edu/www/research/online/hoeting1999.pdf>
- Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2008) Mixtures of g-priors for Bayesian Variable Selection. *Journal of the American Statistical Association*. 103:410-423.
<http://dx.doi.org/10.1198/016214507000001337>
- Zellner, A. (1986) On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*, pp. 233-243. North-Holland/Elsevier.
- Zellner, A. and Siow, A. (1980) Posterior odds ratios for selected regression hypotheses. In *Bayesian Statistics: Proceedings of the First International Meeting held in Valencia (Spain)*, pp. 585-603.

See Also

[summary.bma](#), [coefficients.bma](#), [print.bma](#), [predict.bma](#), [fitted.bma](#) [plot.bma](#), [image.bma](#), [eplogprob](#), [update.bma](#)

Examples

```
demo(BAS.hald)
## Not run: demo(BAS.USCrime)
```

bayesglm.fit	<i>Fitting Generalized Linear Models Bayesian marginal likelihood evaluation</i>
--------------	--

Description

A version of `glm.fit` rewritten in C; also returns marginal likelihoods for Bayesian model comparison

Usage

```
bayesglm.fit(x, y, weights = rep(1, nobs),
             start = NULL, etastart = NULL,
             mustart = NULL, offset = rep(0, nobs), family = binomial(),
             coefprior = bic.prior(nobs),
             control = glm.control(), intercept = TRUE)
```

Arguments

x	design matrix
y	response
weights	optional vector of weights to be used in the fitting process. Should be NULL or a numeric vector.
start	starting value for coefficients in the linear predictor
etastart	starting values for the linear predictor
mustart	starting values for the vectors of means
offset	a priori known component to be included in the linear predictor
family	a description of the error distribution and link function for exponential family; currently only <code>binomial()</code> is coded.
coefprior	function specifying prior distribution on coefficients with optional hyperparameters leading to marginal likelihood calculations; options include <code>bic.prior()</code> , <code>aic.prior()</code> , and <code>ic.prior()</code>
control	a list of parameters that control convergence in the fitting process. See the documentation for <code>glm.control()</code>
intercept	should an intercept be included in the null model?

Details

C version of glm-fit. For different prior choices returns, marginal likelihood of model using a Laplace approximation.

Value

coefficients	MLEs
se	Standard errors of coefficients based on the sqrt of the diagonal of the inverse information matrix
mu	fitted mean
rank	numeric rank of the fitted linear model
deviance	minus twice the log likelihood evaluated at the MLEs
g	value of g in g-priors
shrinkage	shrinkage factor for coefficients in linear predictor
RegSS	quadratic form $\beta'I(\beta)\beta$ used in shrinkage
logmarglik	the log marginal or integrated log likelihood (up to a constant)

Author(s)

Merlise Clyde translated the `glm.fit` from R base into C using the `.Call` interface

References

[glm](#)

See Also

[bic.prior](#)

Bernoulli

Independent Bernoulli Prior Distribution for Models

Description

Creates an object representing the prior distribution on models for BAS.

Usage

```
Bernoulli(probs=.5)
```

Arguments

`probs` a scalar or vector of prior inclusion probabilities. If a scalar, the values is replicated for all variables and a 1 is added for the intercept. BAS checks to see if the length is equal to the dimension of the parameter vector for the full model and adds a 1 to include the intercept.

Details

The independent Bernoulli prior distribution is a commonly used prior in BMA, with the Uniform distribution a special case with $\text{probs}=.5$. If all indicator variables have a independent Bernoulli distributions with common probability probs , the distribution on model size $\text{binomial}(p, \text{probs})$ distribution.

Value

returns an object of class "prior", with the family and hyperparameters.

Author(s)

Merlise Clyde

See Also

[bas.lm](#), [beta.binomial](#), [uniform](#)

Examples

```
Bernoulli(.9)
```

beta.binomial

Beta-Binomial Prior Distribution for Models

Description

Creates an object representing the prior distribution on models for BAS.

Usage

```
beta.binomial(alpha=1.0, beta=1.0)
```

Arguments

alpha parameter in the beta prior distribution

beta parameter in the beta prior distribution

Details

The beta-binomial distribution on model size is obtained by assigning each variable inclusion indicator independent Bernoulli distributions with probability w , and then giving w a $\text{beta}(\alpha, \beta)$ distribution. Marginalizing over w leads to the distribution on model size having the beta-binomial distribution. The default hyperparameters lead to a uniform distribution over model size.

Value

returns an object of class "prior", with the family and hyperparameters.

Author(s)

Merlise Clyde

See Also

[bas.lm](#), [Bernoulli](#), [uniform](#)

Examples

```
beta.binomial(1,10)
```

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 `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	<i>Coefficients of a Bayesian Model Average object</i>
----------	--

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

```
## 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	a matrix with conditional posterior means for each model
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 [plot.coef.bma](#) to view distributions.

Author(s)

Merlise Clyde <clyde@stat.duke.edu>

References

Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2005) Mixtures of g -priors for Bayesian Variable Selection.
<http://www.stat.duke.edu/05-12.pdf>

See Also

[bas](#)

Examples

```
data("Hald")
## Not run: 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(Not run)
```

cv.summary.bma

Summaries for Out of Sample Prediction

Description

Compute summaries from out of sample predictions for a BMA object

Usage

```
cv.summary.bma(object, pred, ytrue)
```

Arguments

object	an object of class 'bma'
pred	output from predict.bma
ytrue	vector of left out response values

Value

A matrix with the best models, posterior probabilities, R^2 , dimensions, Average Prediction Error from the HPM and Average prediction error for BMA prediction

Author(s)

Merlise Clyde <clyde@stat.duke.edu>

See Also

[predict.bma](#)

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

```
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

Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2005) Mixtures of g -priors for Bayesian Variable Selection.
<http://www.stat.duke.edu/05-12.pdf>

See Also

[bas](#), [update](#)

Examples

```
## Not run: 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)

## End(Not run)
```

eplogprob	<i>eplogprob - Compute approximate marginal inclusion probabilities from pvalues</i>
-----------	--

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 <code>bas.lm</code> 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 $H_0: \beta = 0$ versus $H_1: \beta \neq 0$, when the p-value p is less than $1/e$. Using equal prior odds on the hypotheses H_0 and H_1 , 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 base 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

Sellke, Thomas, Bayarri, M. J., and Berger, James O. (2001), "Calibration of p-values for testing precise null hypotheses", *The American Statistician*, 55, 62-71.

See Also

[bas](#)

Examples

```
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

```
## 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

Barbieri, M. and Berger, J.O. (2004) Optimal predictive model selection. *Annals of Statistics*. 32, 870-897. <http://projecteuclid.org/Dienst/UI/1.0/Summarize/euclid.aos/1085408489>

See Also

[predict.bma](#)

Examples

```
data(Hald)
hald.gprior = bas.lm(Y~ ., data=Hald, 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"))
```

Description

Creates an object representing the g-prior distribution on coefficients for BAS.

Usage

```
g.prior(g)
```

Arguments

`g` a scalar used in the covariance of Zellner's g-prior, $\text{Cov}(\beta) = \sigma^2 g (X'X)^{-1}$

Details

Creates a structure used for BAS.

Value

returns an object of class "prior", with the family and hyperparameters.

Author(s)

Merlise Clyde

See Also

[IC.prior](#)

Examples

```
g.prior(100)
```

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, X1: *tricalcium aluminate*, X2: *tricalcium silicate*, X3: *tetracalcium alumino ferrite*, X4: *dicalcium silicate*. An important feature of these data is that the variables X1 and X3 are highly correlated, as well as the variables X2 and X4. Thus we should expect any subset of (X1,X2,X3,X4) that includes one variable from highly correlated pair to do as any subset that also includes the other member.

Usage

```
data(Hald)
```

Format

hald is a dataframe with 13 observations and 5 variables (columns),

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

Source

Wood, H., Steinour, H.H., and Starke, H.R. (1932). "Effect of Composition of Portland cement on Heat Evolved During Hardening", *Industrial and Engineering Chemistry*, 24, 1207-1214.

hypergeometric2F1 *Gaussian hypergeometric2F1 function*

Description

Compute the Gaussian Hypergeometric2F1 function: ${}_2F_1(a,b,c,z) = \frac{\Gamma(b-c)}{\Gamma(b)\Gamma(c-b)} \int_0^1 t^{b-1} (1-t)^{c-b-1} (1-tz)^{-a} dt$

Usage

```
hypergeometric2F1(a,b,c,z, method="Cephes", log=TRUE)
```

Arguments

a	arbitrary
b	Must be greater 0
c	Must be greater than b if $ z < 1$, and $c > b + a$ if $z = 1$
z	$ z \leq 1$
method	The default is to use the Cephes library routine. This sometimes is unstable for large a or z near one returning Inf or negative values. In this case, try method="Laplace", which use a Laplace approximation for $\tau = \exp(t/(1-t))$.
log	if TRUE, return log(2F1)

Details

The default is to use the routine hyp2f1.c from the Cephes library. If that return a negative value or Inf, one should try method="Laplace" which is based on the Laplace approximation as described in Liang et al JASA 2008. This is used in the hyper-g prior to calculate marginal likelihoods.

Value

if log=T returns the log of the 2F1 function; otherwise the 2F1 function.

Author(s)

Merlise Clyde (<clyde@stat.duke.edu>)

References

Cephes library hyp2f1.c

Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2005) Mixtures of g-priors for Bayesian Variable Selection. Journal of the American Statistical Association
<http://www.stat.duke.edu/05-12.pdf>

Examples

```
hypergeometric2F1(12,1,2, .65)
```

IC.prior

Families of Prior Distribution for Coefficients in BMA Models

Description

Creates an object representing the prior distribution on coefficients for BAS.

Usage

```
IC.prior(penalty)
```

Arguments

penalty a scalar used in the penalized loglikelihood of the form penalty*dimension

Details

The log marginal likelihood is approximated as $-2*(\text{deviance} + \text{penalty}*\text{dimension})$. Allows alternatives to AIC (penalty = 2) and BIC (penalty = $\log(n)$)

Value

returns an object of class "prior", with the family and hyperparameters.

Author(s)

Merlise Clyde

See Also

[g.prior](#)

Examples

```
IC.prior(2)
aic.prior()
bic.prior(100)
```

 image.bma

Images of models used in Bayesian model averaging

Description

Creates an image of the models selected using `bas`.

Usage

```
## 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

<code>x</code>	An object of type 'bma' created by BAS
<code>top.models</code>	Number of the top ranked models to plot
<code>intensity</code>	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.
<code>prob</code>	Logical variable for whether the area in the image for each model should be proportional to the posterior probability (or log probability) of the model (TRUE) or with equal area (FALSE).
<code>log</code>	Logical variable indicating whether the intensities should be based on log posterior odds (TRUE) or posterior probabilities (FALSE). The log of the posterior odds is for comparing the each model to the worst model in the top.models.
<code>rotate</code>	Should the image of models be rotated so that models are on the y-axis and variables are on the x-axis (TRUE)
<code>color</code>	The color scheme for image intensities. The value "rainbow" uses the rainbow palette. The value "blackandwhite" produces a black and white image (greyscale image)
<code>subset</code>	indices of variables to include in plot; 1 is the intercept
<code>offset</code>	numeric value to add to intensity
<code>digits</code>	number of digits in posterior probabilities to keep
<code>vlas</code>	las parameter for placing variable names; see <code>par</code>
<code>plas</code>	las parameter for posterior probability axis
<code>rlas</code>	las parameter for model ranks
<code>...</code>	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

Clyde, M. (1999) Bayesian Model Averaging and Model Search Strategies (with discussion). In Bayesian Statistics 6. J.M. Bernardo, A.P. Dawid, J.O. Berger, and A.F.M. Smith eds. Oxford University Press, pages 157-185.

See Also

[bas](#)

Examples

```
data("Hald")
hald.ZSprior = bas.lm(Y~ ., data=Hald, prior="ZS-null")
image(hald.ZSprior, subset=-1)
```

list2matrix.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

```
## S3 method for class 'bma'
list2matrix(x, what, which.models=NULL)

## S3 method for class 'which'
list2matrix(x, which.models=NULL)

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

list2matrix.bma(x, which) is equivalent to list2matrix.which(x), however, the latter uses sapply rather than a loop. list2matrix.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

```
## Not run: 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 = list2matrix.bma(crime.bic, "ols") # extract all ols coefficients
se = list2matrix.bma(crime.bic, "ols.se")
models = list2matrix.which(crime.bic) #matrix of model indicators
models = which.matrix(crime.bic$which, crime.bic$n.vars) #matrix of model indicators
## End(Not run)
```

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	bma object result of 'bas'
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 <code>deparse(x\$call)</code> 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

This provides a panel of 4 plots: the first is a plot of the residuals versus fitted values under BMA. The second is a plot of the cumulative marginal likelihoods of models; if the model space cannot be enumerated then this provides some indication of whether the probabilities are leveling off. The third is a plot of log marginal likelihood versus model dimension and the fourth plot show the posterior marginal inclusion probabilities.

Author(s)

Merlise Clyde, based on plot.lm by John Maindonald and Martin Maechler

See Also

[plot.coef.bma](#) and [image.bma](#).

Examples

```
data(Hald)
hald.gprior = bas.lm(Y~ ., data=Hald, prior="g-prior", alpha=13,
                    modelprior=beta.binomial(1,1),
                    initprobs="eplog")

plot(hald.gprior)
```

plot.coef.bma	<i>Plots the posterior distributions of coefficients derived from Bayesian model averaging</i>
---------------	--

Description

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

Usage

```
## 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 <c.clyde@stat.duke.edu>

References

Hoeting, J.A., Raftery, A.E. and Madigan, D. (1996). A method for simultaneous variable selection and outlier identification in linear regression. *Computational Statistics and Data Analysis*, 22, 251-270.

See Also

[coef.bma](#)

Examples

```
## Not run: 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)

## End(Not run)
```

predict.bma

Prediction Method for an object of class BMA

Description

Predictions under model averaging from a BMA object

Usage

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

Arguments

object	An object of class BMA, created by bas
newdata	new matrix or vector of data for predictions. May include a column for the intercept or just the predictor variables
top	Use only the top M models, based on posterior probabilities
...	optional extra arguments

Details

Use BMA to form predictions using the top highest probability models. Currently newdata must be in the form of a matrix or vector with variables in the same order as in the model matrix used to obtain the BMA object (see object\$X). Future versions will allow newdata to be a dataframe.

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

See Also

[bas](#), [fitted.bma](#)

Examples

```
## Not run: data("Hald")  
hald.gprior = bas.lm(Y~ ., data=Hald, alpha=13, prior="g-prior")  
predict(hald.gprior, hald.gprior$X[,-1], top=5)  
  
## End(Not run)
```

protein	<i>Protein Activity Data</i>
---------	------------------------------

Description

This data sets includes several predictors of protein activity from an experiment run at Glaxo.

Usage

```
data(protein)
```

Format

protein is a dataframe with 96 observations and 8 predictor variables of protein activity:

[,1]	buf	factor	Buffer
[,2]	pH	numeric	
[,3]	NaCl	numeric	
[,4]	con	numeric	protein concentration
[,5]	ra	factor	reducing agent
[,6]	det	factor	detergent
[,7]	MgCl2	numeric	
[,8]	temp	numeric	(temerature)
[,9]	prot.act1	numeric	
[,10]	prot.act2	numeric	
[,11]	prot.act3	numeric	
[,12]	prot.act4	numeric	protein activity

Source

Clyde, M. A. and Parmigiani, G. (1998), Protein Construct Storage: Bayesian Variable Selection and Prediction with Mixtures, *Journal of Biopharmaceutical Statistics*, 8, 431-443

summary.bma	<i>Summaries of Bayesian Model Averaging objects</i>
-------------	--

Description

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

Usage

```
## 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 <code>print.default</code>

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

```
## Not run: library(MASS)
data(UScrime)
UScrime[, -2] = log(UScrime[, -2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC", initprobs= "eplog")
print(crime.bic)
summary(crime.bic)

## End(Not run)
```

uniform

Uniform Prior Distribution for Models

Description

Creates an object representing the prior distribution on models for BAS.

Usage

```
uniform()
```

Details

The Uniform prior distribution is a commonly used prior in BMA, and is a special case of the independent Bernoulli prior with `probs=.5`. The implied prior distribution on model size is `binomial(p, .5)`.

Value

returns an object of class "prior", with the family name Uniform.

Author(s)

Merlise Clyde

See Also

[bas.lm](#), [beta.binomial](#), [Bernoulli](#),

Examples

```
uniform()
```

```
update.bma
```

Update BMA object using a new prior

Description

Update a BMA object using a new prior distribution on the coefficients.

Usage

```
## S3 method for class 'bma'
update(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

Clyde, M. and George, E. I. (2004) Model uncertainty. *Statist. Sci.*, 19, 81-94.
<http://www.isds.duke.edu/~clyde/papers/statsci.pdf>

Clyde, M. (1999) Bayesian Model Averaging and Model Search Strategies (with discussion). In *Bayesian Statistics 6*. J.M. Bernardo, A.P. Dawid, J.O. Berger, and A.F.M. Smith eds. Oxford University Press, pages 157-185.

Hoeting, J. A., Madigan, D., Raftery, A. E. and Volinsky, C. T. (1999) Bayesian model averaging: a tutorial (with discussion). *Statist. Sci.*, 14, 382-401.
<http://www.stat.washington.edu/www/research/online/hoeting1999.pdf>

Liang, F., Paulo, R., Molina, G., Clyde, M. and Berger, J.O. (2008) Mixtures of g-priors for Bayesian Variable Selection. *JASA*
<http://www.stat.duke.edu/05-12.pdf>

Zellner, A. (1986) On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*, pp. 233-243. North-Holland/Elsevier.

Zellner, A. and Siow, A. (1980) Posterior odds ratios for selected regression hypotheses. In *Bayesian Statistics: Proceedings of the First International Meeting held in Valencia (Spain)*, pp. 585-603.

See Also

[bas](#) for available methods and choices of alpha

Examples

```
## Not run:
library(MASS)
data(UScrime)
UScrime[,-2] = log(UScrime[,-2])
crime.bic = bas.lm(y ~ ., data=UScrime, n.models=2^15, prior="BIC", initprobs= "eplog")
crime.zs = update(crime.bic, newprior="ZS-null")

## End(Not run)
```

Index

- *Topic **datasets**
 - Hald, [17](#)
 - protein, [27](#)
- *Topic **math**
 - hypergeometric2F1, [18](#)
- *Topic **package**
 - BAS-package, [2](#)
- *Topic **print**
 - summary.bma, [27](#)
- *Topic **regression**
 - BAS-package, [2](#)
 - bas.lm, [3](#)
 - bayesglm.fit, [7](#)
 - bin2int, [10](#)
 - coef.bma, [11](#)
 - cv.summary.bma, [12](#)
 - EB.global, [13](#)
 - eplogprob, [14](#)
 - fitted.bma, [15](#)
 - image.bma, [20](#)
 - list2matrix.bma, [21](#)
 - plot.bma, [23](#)
 - plot.coef.bma, [24](#)
 - predict.bma, [25](#)
 - summary.bma, [27](#)
 - update.bma, [29](#)
- AIC.prior (IC.prior), [19](#)
- aic.prior (IC.prior), [19](#)
- BAS (BAS-package), [2](#)
- bas, [3](#), [12](#), [13](#), [15](#), [20–22](#), [26](#), [29](#), [30](#)
- bas (bas.lm), [3](#)
- BAS-package, [2](#)
- bas.lm, [3](#), [9](#), [10](#), [29](#)
- bayesglm.fit, [7](#)
- Bernoulli, [4](#), [8](#), [10](#), [29](#)
- bernoulli (Bernoulli), [8](#)
- Beta.Binomial (beta.binomial), [9](#)
- beta.binomial, [4](#), [9](#), [9](#), [29](#)
- BIC.prior (IC.prior), [19](#)
- bic.prior, [8](#)
- bic.prior (IC.prior), [19](#)
- bin2int, [10](#)
- coef (coef.bma), [11](#)
- coef.bma, [11](#), [25](#)
- coefficients (coef.bma), [11](#)
- coefficients.bma, [6](#), [7](#), [28](#)
- cv.summary.bma, [12](#)
- EB.global, [13](#)
- eplogprob, [4](#), [7](#), [14](#)
- fitted (fitted.bma), [15](#)
- fitted.bma, [6](#), [7](#), [15](#), [26](#)
- g.prior, [16](#), [19](#)
- glm, [8](#)
- glm.fit, [8](#)
- Hald, [17](#)
- hald (Hald), [17](#)
- hypergeometric2F1, [18](#)
- IC.prior, [17](#), [19](#)
- image (image.bma), [20](#)
- image.bma, [6](#), [7](#), [20](#), [24](#)
- list2matrix.bma, [21](#)
- list2matrix.which (list2matrix.bma), [21](#)
- plot (plot.coef.bma), [24](#)
- plot.bma, [6](#), [7](#), [23](#)
- plot.coef.bma, [11](#), [24](#), [24](#)
- predict (predict.bma), [25](#)
- predict.bma, [6](#), [7](#), [12](#), [13](#), [16](#), [25](#)
- print (summary.bma), [27](#)
- print.bma, [7](#)
- print.coef.bma (coef.bma), [11](#)
- protein, [27](#)

summary (summary.bma), [27](#)
summary.bma, [6](#), [7](#), [27](#)

Uniform (uniform), [28](#)

uniform, [4](#), [9](#), [10](#), [28](#)

update, [13](#)

update (update.bma), [29](#)

update.bma, [6](#), [7](#), [29](#)

which.matrix (list2matrix.bma), [21](#)