EXAMPLE

```
# Generate data as in first example from paper

library(mvtnorm)
n = 30
p = 4
n.true = 2
beta = c(rep(1,n.true),rep(0,p-n.true))
times = 1:p
rho = 0.9
sigma = 1
H = abs(outer(times, times, "-"))
V = sigma * rho^H
X = rmvnorm(n,rep(0,p),V)
y = X%*%beta+rnorm(n)

# Estimate both lambda and pi (called weight) via marginal likelihood

optimized.values = full.marginal.opt(X, y)

# Compute and display posterior probabilities for each model using the optimal pair.

post.probs(X, y, lambda=optimized.values$opt.lambda,
weight=optimized.values$opt.weight)

# Note that this full optimization can take a long time for many problems
# May wish to only search over a small set of possible lambdas by using
# the function "fixed.lambda.opt" as demonstrated below.

# Fix lambda at two values (1.5 and -1) and find optimal weight for each.
# Compute and display posterior probabilities for these choices

optimized.weight.1.5 = fixed.lambda.opt(X, y, lambda=1.5)
optimized.weight.neg.1 = fixed.lambda.opt(X, y, lambda=-1)
post.probs(X, y, lambda=1.5, weight=optimized.weight.1.5$opt.weight)
post.probs(X, y, lambda=-1, weight=optimized.weight.neg.1$opt.weight)

# Can compare marginal log-likelihoods by using "optimized.weight.1.5$opt.value"
# Note larger (less negative) is better.
```

REFERENCE

Krishna, A., Bondell, H. D., and Ghosh, S. K. (2009). Bayesian variable selection using an adaptive powered correlation prior. *Journal of Statistical Planning and Inference* **139**, 2665-2674.

post.probs *Function to compute the posterior probabilities for all models, for a given set of power parameter and prior weight.*

USAGE

post.probs (X, y, lambda, weight, vo=.01)

ARGUMENTS

X	Design matrix, not including intercept. The columns should have mean zero and sum of squares 1. If not, the code will standardize the design and the output is based on this standardized design.
y	Vector of responses. Should have mean zero, so that intercept is zero.
lambda	Specified value of power parameter.
weight	Specified value of prior inclusion weights (denoted by pi in the paper).
vo	Specified value for inverse gamma prior. Default is 0.01.

VALUE

post.prob	Exact posterior probabilities for each of the models.
models	List of the models using vector of 0/1 to indicate whether each predictor is out/in the model. A 1 denotes that the variable in that position is included in that model.
log.marginal	Value of the marginal log-likelihood for the specified values of the power parameter and prior weight.
lambda	Power parameter used.
weight	Prior weight used.

fixed.lambda.opt *Function to obtain the optimal prior weight for a fixed value of the power parameter via empirical Bayes. Also allows for comparisons across lambdas to find best lambda over a chosen grid.*

USAGE

fixed.lambda.opt (X, y, lambda, vo=.01)

ARGUMENTS

X	Design matrix, not including intercept. The columns should have mean zero and sum of squares 1. If not, the code will standardize the design and the output is based on this standardized design.
y	Vector of responses. Should have mean zero, so that intercept is zero.
lambda	Specified value of power parameter.
vo	Specified value for inverse gamma prior. Default is 0.01.

VALUE

opt.weight	Optimal choice of prior weight via maximizing log-marginal for this choice of lambda.
opt.value	Value of log-marginal obtained. Can be used to compare various values for lambda (larger/less-negative is better).

full.marginal.opt *Function to obtain the optimal pair of power parameter and prior weight via empirical Bayes. This can take quite some time depending on the problem.*

USAGE

```
full.marginal.opt (X, y, lambda.bounds=c(-3,3), vo=.01)
```

ARGUMENTS

X	Design matrix, not including intercept. The columns should have mean zero and sum of squares 1. If not, the code will standardize the design and the output is based on this standardized design.
y	Vector of responses. Should have mean zero, so that intercept is zero.
lambda.bounds	Lower and upper bounds for choice of lambda. Default is -3 to 3.
vo	Specified value for inverse gamma prior. Default is 0.01.

VALUE

opt.lambda	Optimal choice of lambda via jointly maximizing log-marginal.
opt.weight	Optimal choice of prior weight via jointly maximizing log-marginal.
opt.value	Value of log-marginal obtained.