Package ‘BsMD’

February 19, 2015

Version 2013.0718
Date 2013/07/18
Title Bayes Screening and Model Discrimination
Author Ernesto Barrios based on Daniel Meyer's code.
Maintainer Ernesto Barrios <ebarrios@itam.mx>
Description Bayes screening and model discrimination follow-up designs.
Depends R(>= 2.14.0)
License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
Date/Publication 2013-07-19 22:17:05

R topics documented:

BsMD-package .................................................. 2
BM86.data ....................................................... 2
BM93.e1.data ................................................... 4
BM93.e2.data ................................................... 5
BM93.e3.data ................................................... 6
BsProb ........................................................... 7
DanielPlot ....................................................... 9
LenthPlot ....................................................... 11
MD .............................................................. 13
PB12Des ........................................................ 16
plot.BsProb .................................................. 17
print.BsProb ................................................ 18
print.MD ...................................................... 20
Reactor.data .................................................. 22
summary.BsProb ............................................ 22
summary.MD .................................................. 24

Index 26
**BsMD-package**  
*Bayes screening and model discrimination follow-up designs*

**Description**

Bayes screening and model discrimination follow-up designs

**Details**

- **Package:** BsMD
- **Type:** Package
- **Version:** 2013.0718
- **Date:** 2013-07-18
- **License:** GPL version 2 or later

The packages allows you to perform the calculations and analyses described in Mayer, Stainberg and Box paper in Technometrics, 1996.

**Author(s)**

Author: Ernesto Barrios based on Daniel Meyer’s code. Maintainer: Ernesto Barrios <ebarrios@itam.mx>

**References**

Box and Mayer, 1986; Box and Mayer, 1993; Mayer, Steinberg and Box, 1996.

**Examples**

```r
data(BM86.data)
```

---

**BM86.data**  
*Data sets in Box and Meyer (1986)*

**Description**

Design factors and responses used in the examples of Box and Meyer (1986)

**Usage**

```r
data(BM86.data)
```
Format

A data frame with 16 observations on the following 19 variables.

X1 numeric vector. Contrast factor.
X2 numeric vector. Contrast factor.
X3 numeric vector. Contrast factor.
X4 numeric vector. Contrast factor.
X5 numeric vector. Contrast factor.
X6 numeric vector. Contrast factor.
X7 numeric vector. Contrast factor.
X8 numeric vector. Contrast factor.
X9 numeric vector. Contrast factor.
X10 numeric vector. Contrast factor.
X11 numeric vector. Contrast factor.
X12 numeric vector. Contrast factor.
X13 numeric vector. Contrast factor.
X14 numeric vector. Contrast factor.
X15 numeric vector. Contrast factor.
y1 numeric vector. Log drill advance response.
y2 numeric vector. Tensile strength response.
y3 numeric vector. Shrinkage response.
y4 numeric vector. Yield of isatin response.

Source


Examples

library(BsMD)
data(BM86.data,package="BsMD")
print(BM86.data)
Example 1 data in Box and Meyer (1993)

Description

12-run Plackett-Burman design from the $2^5$ reactor example from Box, Hunter and Hunter (1977).

Usage

data(BM93.e1.data)

Format

A data frame with 12 observations on the following 7 variables.

Run a numeric vector. Run number from a $2^5$ factorial design in standard order.
A a numeric vector. Feed rate factor.
B a numeric vector. Catalyst factor.
C a numeric vector. Agitation factor.
D a numeric vector. Temperature factor.
E a numeric vector. Concentration factor.
y a numeric vector. Percent reacted response.

Source


Examples

library(BsMD)
data(BM93.e1.data, package="BsMD")
print(BM93.e1.data)
BM93.e2.data

Example 2 data in Box and Meyer (1993)

Description

12-run Plackett-Burman design for the study of fatigue life of weld repaired castings.

Usage

data(BM93.e2.data)

Format

A data frame with 12 observations on the following 8 variables.

A  a numeric vector. Initial structure factor.
B  a numeric vector. Bead size factor.
C  a numeric vector. Pressure treat factor.
D  a numeric vector. Heat treat factor.
E  a numeric vector. Cooling rate factor.
F  a numeric vector. Polish factor.
G  a numeric vector. Final treat factor.
y  a numeric vector. Natural log of fatigue life response.

Source


Examples

library(BsMD)
data(BM93.e2.data,package="BsMD")
print(BM93.e2.data)
BM93.e3.data

Example 3 data in Box and Meyer (1993)

Description

$2^{8-4}$ Fractional factorial design in the injection molding example from Box, Hunter and Hunter (1978).

Usage

data(BM93.e3.data)

Format

A data frame with 20 observations on the following 10 variables.

- **blk**  a numeric vector
- **A** a numeric vector. Mold temperature factor.
- **B** a numeric vector. Moisture content factor.
- **C** a numeric vector. Holding Pressure factor.
- **D** a numeric vector. Cavity thickness factor.
- **E** a numeric vector. Booster pressure factor.
- **F** a numeric vector. Cycle time factor.
- **G** a numeric vector. Gate size factor.
- **H** a numeric vector. Screw speed factor.
- **y** a numeric vector. Shrinkage response.

Source


Examples

```r
library(BsMD)
data(BM93.e3.data, package="BsMD")
print(BM93.e3.data)
```
**BsProb**  
*Posterior Probabilities from Bayesian Screening Experiments*

**Description**
Marginal factor posterior probabilities and model posterior probabilities from designed screening experiments are calculated according to Box and Meyer’s Bayesian procedure.

**Usage**
\[
\text{BsProb}(X, y, blk, mFac, mInt = 2, p = 0.25, g = 2, ng = 1, nMod = 10)
\]

**Arguments**
- **X**: Matrix. The design matrix.
- **y**: vector. The response vector.
- **blk**: integer. Number of blocking factors (>=0). These factors are accommodated in the first columns of matrix X. There are \( ncol(X) - blk \) design factors.
- **mFac**: integer. Maximum number of factors included in the models.
- **mInt**: integer <= 3. Maximum order of interactions considered in the models.
- **p**: numeric. Prior probability assigned to active factors.
- **g**: vector. Variance inflation factor(s) \( \gamma \) associated to active and interaction factors.
- **ng**: integer <=20. Number of different variance inflation factors (g) used in calculations.
- **nMod**: integer <=100. Number of models to keep with the highest posterior probability.

**Details**
Factor and model posterior probabilities are computed by Box and Meyer’s Bayesian procedure. The design factors are accommodated in the matrix X after blk columns of the blocking factors. So, \( ncol(X) - blk \) design factors are considered. If g, the variance inflation factor (VIF) \( \gamma \), is a vector of length 1, the same VIF is used for factor main effects and interactions. If the length of g is 2 and ng is 1, g[1] is used for factor main effects and g[2] for the interaction effects. If ng greater than 1, then ng values of VIFs between g[1] and g[2] are used for calculations with the same \textit{gamma} value for main effects and interactions. The function calls the FORTRAN subroutine ‘bm’ and captures summary results. The output is a list of class BsProb for which print, plot and summary methods are available.

**Value**
A list with all output parameters of the FORTRAN subroutine ‘bm’. The names of the list components are such that they match the original FORTRAN code. Small letters used for capturing program’s output.

- **X**: matrix. The design matrix.
Y vector. The response vector.
N integer. The number of runs.
COLS integer. The number of design factors.
BLKS integer. The number of blocking factors accommodated in the first columns of matrix X.
MXFAC integer. Maximum number of factors considered in the models.
MXINT integer. Maximum interaction order considered in the models.
PI numeric. Prior probability assigned to the active factors.
INDGAM integer. If 0, the same variance inflation factor (GAMMA) is used for main and interactions effects. If INDGAM == 1, then NGAM different values of GAMMA were used.
INDG2 integer. If 1, the variance inflation factor GAM2 was used for the interaction effects.
NGAM integer. Number of different VIFs used for computations.
GAMMA vector. Vector of variance inflation factors of length 1 or 2.
NTOP integer. Number of models with the highest posterior probability

mdcnt integer. Total number of models evaluated.
ptop vector. Vector of probabilities of the top ntop models.
sigtop vector. Vector of sigma-squared of the top ntop models.
nftop integer. Number of factors in each of the ntop models.
jtop matrix. Matrix of the number of factors and their labels of the top ntop models.
del numeric. Interval width of the GAMMA partition.
sprob vector. Vector of posterior probabilities. If ng>1 the probabilities are weighted averaged over GAMMA.
pgam vector. Vector of values of the unscaled posterior density of GAMMA.
prob matrix. Matrix of marginal factor posterior probabilities for each of the different values of GAMMA.
ind integer. Indicator variable. ind is 1 if the ‘bm’ subroutine exited properly. Any other number correspond to the format label number in the FORTRAN subroutine script.

Note
The function is a wrapper to call the FORTRAN subroutine ‘bm’, modification of Daniel Meyer’s original program, ‘mbcqp5.f’, for the application of Bayesian design and analysis of fractional factorial experiments, part of the mdopt bundle, available at StatLib.

Author(s)
R. Daniel Meyer. Adapted for R by Ernesto Barrios.
References


See Also


Examples

library(BsMD)
data(BM86.data, package="BsMD")
X <- as.matrix(BM86.data[,1:15])
y <- BM86.data["y1"]
# Using prior probability of p = 0.20, and k = 10 (gamma = 2.49)
drillAdvance.BsProb <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
p = 0.20, g = 2.49, ng = 1, nMod = 10)
plot(drillAdvance.BsProb)
summary(drillAdvance.BsProb)

# Using prior probability of p = 0.20, and a 5 <= k <= 15 (1.22 <= gamma <= 3.74)
drillAdvance.BsProbG <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
p = 0.25, g = c(1.22, 3.74), ng = 3, nMod = 10)
plot(drillAdvance.BsProbG, code = FALSE, prt = TRUE)

DanielPlot

Normal Plot of Effects

Description

Normal plot of effects from a two level factorial experiment.

Usage

DanielPlot(fit, code = FALSE, faclab = NULL, block = FALSE,
datax = TRUE, half = FALSE, pch = "*", cex.fac = par("cex.lab"),
cex.lab = par("cex.lab"), cex.pch = par("cex.axis"), ...)

Arguments

fit object of class lm. Fitted model from lm or aov.
code logical. If TRUE labels "A","B", etc are used instead of the names of the coefficients (factors).
faclab list. If NULL points are labelled accordingly to code, otherwise faclab should be a list with idx (integer vector) and lab (character vector) components. See Details.
block logical. If TRUE, the first factor is labelled as "BK" (block).
datax logical. If TRUE, the x-axis is used for the factor effects the y-axis for the normal scores. The opposite otherwise.
half logical. If TRUE, half-normal plot of effects is display.
pch numeric or character. Points character.
cex.fac numeric. Factors’ labels character size.
cex.lab numeric. Labels character size.
cex.pch numeric. Points character size.

... graphical parameters passed to plot.

Details
The two levels design are assumed -1 and 1. Factor effects assumed 2*coef(obj) ((Intercept) removed) are displayed in a qqnorm plot with the effects in the x-axis by default. If half=TRUE the half-normal plots of effects is plotted as the normal quantiles of 0.5*(rank(abs(effects))-0.5)/length(effects)+1 versus abs(effects).

Value
The function returns invisible data frame with columns: x, y and no, for the coordinates and the enumeration of plotted points. Names of the factor effects (coefficients) are the row names of the data frame.

Author(s)
Ernesto Barrios.

References

See Also
qqnorm, LenthPlot

Examples
### Injection Molding Experiment. Box et al. 1978.
library(BsMD)
# Data
data(BM86.data,package="BsMD") # Design matrix and response
print(BM86.data) # from Box and Meyer (1986)

# Model Fitting. Box and Meyer (1986) example 3.
injectionMolding.lm <- lm(y3 ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 + X10 + X11 + X12 + X13 + X14 + X15, data = BM86.data)
print(coef(injectionMolding.lm)) # Model coefficients
# Daniel Plots
par(mfrow=c(1,3),oma=c(0,0,1,0),pty="s")
DanielPlot(injectionMolding.lm, half = TRUE, main = "Half-Normal Plot")
DanielPlot(injectionMolding.lm, main = "Normal Plot of Effects")
DanielPlot(injectionMolding.lm,
   faclab = list(idx = c(12,4,13), lab = c(" -H"," VG"," -B")),
   main = "Active Contrasts")

LenthPlot

Lenth's Plot of Effects

Description

Plot of the factor effects with significance levels based on robust estimation of contrast standard errors.

Usage

LenthPlot(obj, alpha = 0.05, plt = TRUE, limits = TRUE,
   xlab = "factors", ylab = "effects", faclab = NULL, cex.fac = par("cex.lab"),
   cex.axis=par("cex.axis"), adj = 1, ...)

Arguments

obj object of class lm or vector with the factor effects.
alpha numeric. Significance level used for the margin of error (ME) and simultaneous margin of error (SME). See Lenth(1989).
plt logical. If TRUE, a spikes plot with the factor effects is displayed. Otherwise, no plot is produced.
limits logical. If TRUE ME and SME limits are displayed and labeled.
xlab character string. Used to label the x-axis. "factors" as default.
ylab character string. Used to label the y-axis. "effects" as default.
faclab list with components idx (numeric vector) and lab (character vector). The idx entries of effects vector (taken from obj) are labelled as lab. The rest of the effect names are blanked. If NULL all factors are labelled using the coefficients' name.
cex.fac numeric. Character size used for the factor labels.
cex.axis numeric. Character size used for the axis.
adj numeric between 0 and 1. Determines where to place the "ME" (margin of error) and the "SME" (simultaneous margin of error) labels (character size of 0.9*cex.axis). 0 for extreme left hand side, 1 for extreme right hand side.
... extra parameters passed to plot.
Details

If `obj` is of class `lm`, `2*coef(obj)` is used as factor effect with the intercept term removed. Otherwise, `obj` should be a vector with the factor effects. Robust estimate of the contrasts standard error is used to calculate marginal (ME) and simultaneous margin of error (SME) for the provided significance \((1 - \alpha)\) level. See Lenth(1989). Spikes are used to display the factor effects. If `faclab` is NULL, factors are labelled with the effects or coefficient names. Otherwise, those `faclab$idx` factors are labelled as `faclab$lab`. The rest of the factors are blanked.

Value

The function is called mainly for its side effect. It returns a vector with the value of \(\alpha\) used, the estimated PSE, ME and SME.

Author(s)


References


See Also

`DanielPlot`, `BsProb` and `plot.BsProb`

Examples

```r
### Tensile Strength Experiment. Taguchi and Wu. 1980
library(BsMD)
# Data
data(BM86.data, package="BsMD")  # Design matrix and responses
print(BM86.data)                 # from Box and Meyer (1986)

# Model Fitting. Box and Meyer (1986) example 2.
tensileStrength.lm <- lm(y2 ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 +
                         X10 + X11 + X12 + X13 + X14 + X15, data = BM86.data)
print(coef(tensileStrength.lm))  # Model coefficients

par(mfrow=c(1,2), pty="s")
DanielPlot(tensileStrength.lm, main = "Daniel Plot")
LenthPlot(tensileStrength.lm, main = "Lenth's Plot")
```

Best Model Discrimination (MD) Follow-Up Experiments

Description

Best follow-up experiments based on the MD criterion are suggested to discriminate between competing models.

Usage

```
MD(X, y, nFac, nBlk = 0, mInt = 3, g = 2, nMod, p, s2, nf, facs, nFDes = 4,
Xcand, mIter = 20, nStart = 5, startDes = NULL, top = 20, eps = 1e-05)
```

Arguments

- `X` matrix. Design matrix of the initial experiment.
- `y` vector. Response vector of the initial experiment.
- `nFac` integer. Number of factors in the initial experiment.
- `nBlk` integer >=1. The number of blocking factors in the initial experiment. They are accommodated in the first columns of matrix `X`.
- `mInt` integer. Maximum order of the interactions in the models.
- `g` vector. Variance inflation factor for main effects (g[1]) and interactions effects (g[2]). If vector length is 1 the same inflation factor is used for main and interactions effects.
- `nMod` integer. Number of competing models.
- `p` vector. Posterior probabilities of the competing models.
- `s2` vector. Competing model variances.
- `nf` vector. Factors considered in each of the models.
- `facs` matrix. Matrix [nMod x max(nf)] of factor numbers in the design matrix.
- `nFDes` integer. Number of runs to consider in the follow-up experiment.
- `Xcand` matrix. Candidate runs to be chosen for the follow-up design.
- `mIter` integer. If 0, then user-entered designs startDes are evaluated, otherwise the maximum number of iterations for each Wynn search.
- `nStart` integer. Number of starting designs.
- `startDes` matrix. Matrix [nStart x nFDes]. Each row has the row numbers of the user-supplied starting design.
- `top` integer. Highest MD follow-up designs recorded.
- `eps` numeric. A small number (1e-5 by default) used for computations.
Details

The MD criterion, proposed by Meyer, Steinberg and Box is used to discriminate among competing models. Random starting runs chosen from Xcand are used for the Wynn search of best MD follow-up designs. nStart starting points are tried in the search limited to mIter iterations. If mIter=0 then startDes user-provided designs are used. Posterior probabilities and variances of the competing models are obtained from BsProb. The function calls the FORTRAN subroutine ‘md’ and captures summary results.

Value

A list with all input and output parameters of the FORTRAN subroutine MD. Most of the variable names kept to match FORTRAN code.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSTART</td>
<td>Number of starting designs.</td>
</tr>
<tr>
<td>NRUNS</td>
<td>Number of runs used in follow-up designs.</td>
</tr>
<tr>
<td>ITMAX</td>
<td>Maximum number of iterations for each Wynn search.</td>
</tr>
<tr>
<td>INITDES</td>
<td>Number of starting points.</td>
</tr>
<tr>
<td>NO</td>
<td>Numbers of runs already completed before follow-up.</td>
</tr>
<tr>
<td>IND</td>
<td>Indicator; 0 indicates the user supplied starting designs.</td>
</tr>
<tr>
<td>X</td>
<td>Matrix for initial data (nrow(X)=N0; ncol(X)=COLS+BL).</td>
</tr>
<tr>
<td>Y</td>
<td>Response values from initial experiment (length(Y)=N0).</td>
</tr>
<tr>
<td>GAMMA</td>
<td>Variance inflation factor.</td>
</tr>
<tr>
<td>GAM2</td>
<td>If IND=1, GAM2 was used for interaction factors.</td>
</tr>
<tr>
<td>BL</td>
<td>Number of blocks (&gt;=1) accommodated in first columns of X and Xcand.</td>
</tr>
<tr>
<td>COLS</td>
<td>Number of factors.</td>
</tr>
<tr>
<td>N</td>
<td>Number of candidate runs.</td>
</tr>
<tr>
<td>Xcand</td>
<td>Matrix of candidate runs. (nrow(Xcand)=N, ncol(Xcand)=ncol(X)).</td>
</tr>
<tr>
<td>NM</td>
<td>Number of models considered.</td>
</tr>
<tr>
<td>P</td>
<td>Models posterior probability.</td>
</tr>
<tr>
<td>SIGMA2</td>
<td>Models variances.</td>
</tr>
<tr>
<td>NF</td>
<td>Number of factors per model.</td>
</tr>
<tr>
<td>MNF</td>
<td>Maximum number of factor in models. (MNF=max(NF)).</td>
</tr>
<tr>
<td>JFAC</td>
<td>Matrix with the factor numbers for each of the models.</td>
</tr>
<tr>
<td>CUT</td>
<td>Maximum interaction order considered.</td>
</tr>
<tr>
<td>MBEST</td>
<td>If INITDES=0, the first row of the MBEST[1,] matrix has the first user-supplied starting design. The last row the NSTART-th user-supplied starting design.</td>
</tr>
<tr>
<td>NTOP</td>
<td>Number of the top best designs.</td>
</tr>
<tr>
<td>TOPD</td>
<td>The D value for the best NTOP designs.</td>
</tr>
<tr>
<td>TOPDES</td>
<td>Top NTOP design factors.</td>
</tr>
<tr>
<td>ESP</td>
<td>&quot;Small number&quot; provided to the ‘md’ FORTRAN subroutine. 1e-5 by default.</td>
</tr>
<tr>
<td>flag</td>
<td>Indicator = 1, if the ‘md’ subroutine finished properly, -1 otherwise.</td>
</tr>
</tbody>
</table>
Note

The function is a wrapper to call the FORTAN subroutine ‘md’, modification of Daniel Meyer’s original program, ‘md.f’, part of the mdopt bundle for Bayesian model discrimination of multifactor experiments.

Author(s)

R. Daniel Meyer. Adapted for R by Ernesto Barrios.

References


See Also

print.MD, BsProb

Examples

```R
### Injection Molding Experiment. Meyer et al. 1996, example 2.
library(BsMD)
data(BM93.e3.data, package="BsMD")
X <- as.matrix(BM93.e3.data[1:16,c(1,2,4,6,9)])
y <- BM93.e3.data[1:16,10]
p <- c(0.2356,0.2356,0.2356,0.2356,0.0566)
s2 <- c(0.5815,0.5815,0.5815,0.5815,0.4412)
nf <- c(3,3,3,3,4)
facs <- matrix(c(2,1,1,1,3,3,2,2,2,4,4,3,4,3,0,0,0,0,4),nrow=5,
dimnames=list(1:5,c("f1","f2","f3","f4")))
nFDes <- 4
Xcand <- matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1
),nrow=16, dimnames=list(1:16,c("blk","f1","f2","f3","f4")))
injectionMolding.MD <- MD(X = X, y = y, nFac = 4, nBlk = 1, mInt = 3,
g = 2, nMod = 5, p = p, s2 = s2, nf = nf, facs = facs,
nFDes = 4, Xcand = Xcand, mIter = 20, nStart = 25, top = 10)
summary(injectionMolding.MD)
```

```R
### Reactor Experiment. Meyer et al. 1996, example 3.
par(mfrow=c(1,2), pty="s")
data(Reactor.data, package="BsMD")
```
# Posterior probabilities based on first 8 runs
X <- as.matrix(cbind(blk = rep(-1,8), Reactor.data[c(25,2,19,12,13,22,7,32), 1:5]))
y <- Reactor.data[c(25,2,19,12,13,22,7,32), 6]
reactor8.BsProb <- BsProb(X = X, y = y, blk = 1, mFac = 5, mInt = 3,
p =0.25, g =0.40, ng = 1, nMod = 32)
plot(reactor8.BsProb,prt=TRUE,,main="(8 runs)"

# MD optimal 4-run design
p <- reactor8.BsProb$p\top
s2 <- reactor8.BsProb$s\top
nf <- reactor8.BsProb$n\top
facs <- reactor8.BsProb$f\top
nFDes <- 4
Xcand <- as.matrix(cbind(blk = rep(+1,32), Reactor.data[,1:5]))
reactor.MD <- MD(X = X, y = y, nFac = 5, nBlk = 1, mInt = 3, g =0.40, nMod = 32,
p = p, s2 = s2, nf = nf, facs = facs, nFDes = 4, Xcand = Xcand,
mIter = 20, nStart = 25, top = 5)
s\top

summary(reactor.MD)

# Posterior probabilities based on all 12 runs
X <- rbind(X, Xcand[c(4,10,11,26),])
y <- c(y, Reactor.data[c(4,10,11,26)],6]
reactor12.BsProb <- BsProb(X = X, y = y, blk = 1, mFac = 5, mInt = 3,
p = 0.25, g =1.20, ng = 1, nMod = 5)
plot(reactor12.BsProb,prt=TRUE,main="(12 runs)"

---

**PB12Des**

**12-run Plackett-Burman Design Matrix**

**Description**

12-run Plackett-Burman design matrix.

**Usage**

`data(PB12Des)`

**Format**

A data frame with 12 observations on the following 11 variables.

- **x1** numeric vectors. Contrast factor.
- **x2** numeric vectors. Contrast factor.
- **x3** numeric vectors. Contrast factor.
- **x4** numeric vectors. Contrast factor.
- **x5** numeric vectors. Contrast factor.
- **x6** numeric vectors. Contrast factor.
- **x7** numeric vectors. Contrast factor.
\texttt{plot.BsProb}

- \texttt{x8} numeric vectors. Contrast factor.
- \texttt{x9} numeric vectors. Contrast factor.
- \texttt{x10} numeric vectors. Contrast factor.
- \texttt{x11} numeric vectors. Contrast factor.

**Source**


**Examples**

```r
library(BsMD)
data(PB12Des, package="BsMD")
str(PB12Des)
X <- as.matrix(PB12Des)
print(t(X)%*%X)
```

---

\textbf{plot.BsProb} \hspace{1cm} \textit{Plotting of Posterior Probabilities from Bayesian Screening}

**Description**

Method function for plotting marginal factor posterior probabilities for Bayesian screening.

**Usage**

```r
## S3 method for class 'BsProb'
plot(x, code = TRUE, prt = FALSE, cex.axis=par("cex.axis"), ...)
```

**Arguments**

- \texttt{x} \hspace{1cm} list. List of class \texttt{BsProb} output from the \texttt{BsProb} function.
- \texttt{code} \hspace{1cm} logical. If TRUE coded factor names are used.
- \texttt{prt} \hspace{1cm} logical. If TRUE, summary of the posterior probabilities calculation is printed.
- \texttt{cex.axis} \hspace{1cm} Magnification used for the axis annotation. See \texttt{par}.
- \texttt{...} \hspace{1cm} additional graphical parameters passed to \texttt{plot}.

**Details**

A spike plot, similar to barplots, is produced with a spike for each factor. Marginal posterior probabilities are used for the vertical axis. If code=TRUE, X1, X2, ... are used to label the factors otherwise the original factor names are used. If prt=TRUE, the \texttt{print.BsProb} function is called and the posterior probabilities are displayed. When \texttt{BsProb} is called for more than one value of gamma (g), the spikes for each factor probability are overlapped to show the resulting range of each marginal probability.
Value

The function is called for its side effects. It returns an invisible NULL.

Author(s)

Ernesto Barrios.

References


See Also


Examples

library(BsMD)
data(BM86.data, package="BsMD")
X <- as.matrix(BM86.data[,1:15])
y <- BM86.data["y1"]
# Using prior probability of p = 0.20, and k = 10 (gamma = 2.49)
drillAdvance.BsProb <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
    p = 0.20, g = 2.49, ng = 1, nMod = 10)
plot(drillAdvance.BsProb)
summary(drillAdvance.BsProb)

# Using prior probability of p = 0.20, and 5 <= k <= 15 (1.22 <= gamma <= 3.74)
drillAdvance.BsProbG <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
    p = 0.25, g = c(1.22, 3.74), ng = 3, nMod = 10)
plot(drillAdvance.BsProbG, code = FALSE, prt = TRUE)

print.BsProb

Printing Posterior Probabilities from Bayesian Screening

Description

Printing method for lists of class BsProb. Prints the posterior probabilities of factors and models from the Bayesian screening procedure.

Usage

## S3 method for class 'BsProb'
print(x, X = TRUE, resp = TRUE, factors = TRUE, models = TRUE,
    nMod = 10, digits = 3, plt = FALSE, verbose = FALSE, ...)
print.BsProb

Arguments

x list. Object of BsProb class, output from the BsProb function.
X logical. If TRUE, the design matrix is printed.
resp logical. If TRUE, the response vector is printed.
factors logical. Marginal posterior probabilities are printed if TRUE.
models logical. If TRUE models posterior probabilities are printed.
nMod integer. Number of the top ranked models to print.
digits integer. Significant digits to use for printing.
plt logical. Factor marginal probabilities are plotted if TRUE.
verbose logical. If TRUE, the unclass-ed list x is displayed.
... additional arguments passed to print function.

Value

The function prints out marginal factors and models posterior probabilities. Returns invisible list with the components:

calc numeric vector with general calculation information.
probabilities Data frame with the marginal posterior factor probabilities.
models Data frame with model the posterior probabilities.

Author(s)

Ernesto Barrios.

References


See Also


Examples

library(BsMD)
data(BM86.data, package="BsMD")
X <- as.matrix(BM86.data[,1:15])
y <- BM86.data["y1"]
# Using prior probability of p = 0.20, and k = 10 (gamma = 2.49)
drillAdvance.BsProb <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
                              p = 0.20, g = 2.49, ng = 1, nMod = 10)
print(drillAdvance.BsProb)
plot(drillAdvance.BsProb)

# Using prior probability of $p = 0.20$, and a $5 \leq k \leq 15$ ($1.22 \leq \gamma \leq 3.74$)
drillAdvance.BsProbG <- BsProb(X = x, y = y, blk = 0, mFac = 15, mint = 1, 
  p = 0.25, g = c(1.22, 3.74), ng = 3, nMod = 10)
print(drillAdvance.BsProbG, X = FALSE, resp = FALSE)
plot(drillAdvance.BsProbG)

---

**Print Best MD Follow-Up Experiments**

**Description**

Printing method for lists of class *MD*. Displays the best MD criterion set of runs and their MD for follow-up experiments.

**Usage**

```r
## S3 method for class 'MD'
print(x, X = FALSE, resp = FALSE, Xcand = TRUE, models = TRUE, nMod = x$nMod,
    digits = 3, verbose = FALSE, ...)
```

**Arguments**

- `x` list of class *MD*. Output list of the *MD* function.
- `X` logical. If TRUE, the initial design matrix is printed.
- `resp` logical If TRUE, the response vector of initial design is printed.
- `Xcand` logical. Prints the candidate runs if TRUE.
- `models` logical. Competing models are printed if TRUE.
- `nMod` integer. Top models to print.
- `digits` integer. Significant digits to use in the print out.
- `verbose` logical. If TRUE, the unclass-ed x is displayed.
- `...` additional arguments passed to `print` generic function.

**Value**

The function is mainly called for its side effects. Prints out the selected components of the class *MD* objects, output of the *MD* function. For example the marginal factors and models posterior probabilities and the top MD follow-up experiments with their corresponding MD statistic. It returns invisible list with the components:

- `calc` Numeric vector with basic calculation information.
- `models` Data frame with the competing models posterior probabilities.
- `follow-up` Data frame with the runs for follow-up experiments and their corresponding MD statistic.
Author(s)

Ernesto Barrios.

References


See Also

`md`, `BsProb`

Examples

```r
# MD for one extra experiment.
library(BsMD)
data(BM93.e3.data,package="BsMD")
X <- as.matrix(BM93.e3.data[1:16,c(1,2,4,6,9)])
y <- BM93.e3.data[1:16,10]
nBlk <- 1
nFac <- 4
mInt <- 3
g <- 2
nMod <- 5
p <- c(0.2356,0.2356,0.2356,0.2356,0.0566)
s2 <- c(0.5815,0.5815,0.5815,0.5815,0.4412)
nf <- c(3,3,3,3,4)
facs <- matrix(c(2,1,1,1,1,3,3,2,2,2,4,4,3,3,3,3,0,0,0,0,4),nrow=5,
dimnames=list(1:5,c("f1","f2","f3","f4")))
nFDes <- 1
Xcand <- matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),
                 nrow=16,dimnames=list(1:16,c("blk","f1","f2","f3","f4")))

mIter <- 0
startDes <- matrix(c(9,11,12,15),nrow=4)
top <- 10
injectionMolding.MD <- MD(X=X,y=y,nFac=nFac,nBlk=nBlk,mInt=mInt,g=g,
nMod=nMod,p=p,s2=s2,nf=nf,facs=facs,
nFDes=nFDes,Xcand=Xcand,mIter=mIter,startDes=startDes,top=top)

print(injectionMolding.MD)
summary(injectionMolding.MD)
```
**Reactor.data**  
*Reactor Experiment Data*

**Description**

Data of the Reactor Experiment from Box, Hunter and Hunter (1978).

**Usage**

```r
data(Reactor.data)
```

**Format**

A data frame with 32 observations on the following 6 variables.

- **A** numeric vector. Feed rate factor.
- **B** numeric vector. Catalyst factor.
- **C** numeric vector. Agitation rate factor.
- **D** numeric vector. Temperature factor.
- **E** numeric vector. Concentration factor.
- **y** numeric vector. Percentage reacted response.

**Source**


**Examples**

```r
library(BsMD)
data(Reactor.data, package="BsMD")
print(Reactor.data)
```

---

**summary.BsProb**  
*Summary of Posterior Probabilities from Bayesian Screening*

**Description**

Reduced printing method for class BsProb lists. Prints posterior probabilities of factors and models from Bayesian screening procedure.

**Usage**

```r
## S3 method for class 'BsProb'
summary(object, nMod = 10, digits = 3, ...)
```
**summary.BsProb**

**Arguments**

- **object** list. BsProb class list. Output list of BsProb function.
- **nMod** integer. Number of the top ranked models to print.
- **digits** integer. Significant digits to use.
- **...** additional arguments passed to summary generic function.

**Value**

The function prints out the marginal factors and models posterior probabilities. Returns invisible list with the components:

- **calc** Numeric vector with basic calculation information.
- **probabilities** Data frame with the marginal posterior factor probabilities.
- **models** Data frame with the models posterior probabilities.

**Author(s)**

Ernesto Barrios.

**References**


**See Also**


**Examples**

```r
library(BsMD)
data(BM86.data, package="BsMD")
X <- as.matrix(BM86.data[,1:15])
y <- BM86.data["y1"]
# Using prior probability of p = 0.20, and k = 10 (gamma = 2.49)
drillAdvance.BsProb <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
                             p = 0.20, g = 2.49, ng = 1, nMod = 10)
plot(drillAdvance.BsProb)
summary(drillAdvance.BsProb)

# Using prior probability of p = 0.20, and a 5 <= k <= 15 (1.22 <= gamma <= 3.74)
drillAdvance.BsProbG <- BsProb(X = X, y = y, blk = 0, mFac = 15, mInt = 1,
                             p = 0.25, g = c(1.22, 3.74), ng = 3, nMod = 10)
plot(drillAdvance.BsProbG)
summary(drillAdvance.BsProbG)
```


Summary of Best MD Follow-Up Experiments

Description

Reduced printing method for lists of class MD. Displays the best MD criterion set of runs and their MD for follow-up experiments.

Usage

```r
## S3 method for class 'MD'
summary(object, digits = 3, verbose=FALSE, ...)
```

Arguments

- `object`: list of MD class. Output list of MD function.
- `digits`: integer. Significant digits to use in the print out.
- `verbose`: logical. If TRUE, the unclass-ed object is displayed.
- `...`: additional arguments passed to summary generic function.

Value

It prints out the marginal factors and models posterior probabilities and the top MD follow-up experiments with their corresponding MD statistic.

Author(s)

Ernesto Barrios.

References


See Also

`print.MD` and `MD`
Examples

```r
### Reactor Experiment. Meyer et al. 1996, example 3.
library(BsMD)
data(Reactor.data, package="BsMD")

# Posterior probabilities based on first 8 runs
X <- as.matrix(cbind(blk = rep(-1,8), Reactor.data[c(25,2,19,12,13,22,7,32), 1:5]))
y <- Reactor.data[c(25,2,19,12,13,22,7,32), 6]
reactor.BsProb <- BsProb(X = X, y = y, blk = 1, mFac = 5, mInt = 3,
                         p = 0.25, g = 0.40, ng = 1, nMod = 32)

# MD optimal 4-run design
p <- reactor.BsProb$top
s2 <- reactor.BsProb$sigtop
nf <- reactor.BsProb$nftop
facs <- reactor.BsProb$fjtop
nFDes <- 4
Xcand <- as.matrix(cbind(blk = rep(+1,32), Reactor.data[,1:5]))
reactor.MD <- MD(X = X, y = y, nFac = 5, nBlk = 1, mInt = 3, g = 0.40, nMod = 32,
                  p = p, s2 = s2, nf = nf, facs = facs, nFDes = 4, Xcand = Xcand,
                  mIter = 20, nStart = 25, top = 5)
print(reactor.MD)
summary(reactor.MD)
```
Index

*Topic datasets
   BM86.data, 2
   BM93.e1.data, 4
   BM93.e2.data, 5
   BM93.e3.data, 6
   PB12Des, 16
   Reactor.data, 22

*Topic design
   BsProb, 7
   DanielPlot, 9
   LenthPlot, 11
   MD, 13
   plot.BsProb, 17
   print.BsProb, 18
   print.MD, 20
   summary.BsProb, 22
   summary.MD, 24

*Topic hplot
   plot.BsProb, 17

*Topic package
   BsMD-package, 2
   BsMD(BsMD-package), 2
   BsMD-package, 2
   BsProb, 7, 12, 14, 15, 17–19, 21, 23
   DanielPlot, 9, 12
   LenthPlot, 10, 11
   MD, 13, 20, 21, 24
   par, 17
   PB12Des, 16
   plot.BsProb, 12, 17, 19, 23
   print.BsProb, 9, 17, 18, 18, 23
   print.MD, 15, 20, 24

qqnorm, 10
Reactor.data, 22
summary.BsProb, 9, 18, 19, 22
summary.MD, 24