Package ‘vdg’

April 23, 2024

Type Package

Title Variance Dispersion Graphs and Fraction of Design Space Plots

Version 1.2.3

Date 2024-04-23

Author Pieter Schoonees [aut, cre, cph],
Niel le Roux [ctb]

Maintainer Pieter Schoonees <schoonees@gmail.com>

Depends R (>= 4.3.0), parallel, ggplot2, quantreg

Imports proxy, splines, gridExtra, grDevices, methods, stats, utils

Suggests rsm, AlgDesign, knitr, lhs, tinytex

Description Facilities for constructing variance dispersion graphs, fraction-of-design-space plots and similar graphics for exploring the properties of experimental designs. The design region is explored via random sampling, which allows for more flexibility than traditional variance dispersion graphs. A formula interface is leveraged to provide access to complex model formulae. Graphics can be constructed simultaneously for multiple experimental designs and/or multiple model formulae. Instead of using pointwise optimization to find the minimum and maximum scaled prediction variance curves, which can be inaccurate and time consuming, this package uses quantile regression as an alternative.

VignetteBuilder knitr

License GPL (>= 2)

LazyData yes

NeedsCompilation yes

Encoding UTF-8

RoxygenNote 7.3.1

Repository CRAN

Date/Publication 2024-04-23 13:00:02 UTC

R topics documented:

`vdg-package` *Variance Dispersion Graphs, Fraction-of-Design-Space Plots and Variants*

Description

This package provides functionality for producing variance dispersion graphs (VDGs), fraction-of-design (FDS) plots and related graphics for assessing the prediction variance properties of experimental designs. Random sampling is used to assess the distribution of the prediction variance throughout the design region. Multiple design and/or model formulae can be assessed at the same time. Graphics are produced by the **ggplot2** package.

Details

The workhorse function in the package is `spv`, which takes lists of experimental designs and / or model formulae and produces samples throughout the design region at which the prediction variance is evaluated. Depending on the type of input for the `design` and `formula` arguments, `spv` creates output objects of S3 classes `spv`, `spvlist`, `spvforlist` or `spvlistforlist`. The graphical output are obtained with the `plot` methods of these classes, and the `which` argument can be used to control the type of plots produced.

The design regions allowed for are typically spherical or cuboidal in nature, but the `keepfun` argument to `spv` can be used for rejection sampling. In this way nonstandard design regions can be allowed for. See also the `type` argument of `spv`. The output from the `plot` methods for objects created by `spv` are typically named lists of graphical objects created by **ggplot2**. These are best stored in an object and recreated by printing the required plot. Storing such graphical objects also enables post-hoc manipulation of the plots, such as changing the background colour by using **ggplot2**'s `theme` function.

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

See Also

`spv`, `plot.spv`, and `vignette(topic = "vdg")`.

`D310` *Design from the Vdgraph package*

Description

This data frame is taken verbatim from the (now archived) **Vdgraph** package. See that package for the original reference.

Usage

```
D310
```

Format

a data frame of 10 runs in three variables.

References

Lawson J, Vining G (2014). *Vdgraph: Variance dispersion graphs and Fraction of design space plots for response surface designs*. R package version 2.2-2, https://CRAN.R-project.org/package=Vdgraph.

`D416B` *Design from the Vdgraph package*

Description

This data frame is taken verbatim from the (now archived) **Vdgraph** package. See that package for the original reference.

Usage

```
D416B
```

Format

a data frame of 16 runs in four variables.

References

Lawson J, Vining G (2014). *Vdgraph: Variance dispersion graphs and Fraction of design space plots for response surface designs*. R package version 2.2-2, https://CRAN.R-project.org/package=Vdgraph.

D416C *Design from the Vdgraph package*

Description

This data frame is taken verbatim from the (now archived) **Vdgraph** package. See that package for the original reference.

Usage

```
D416C
```

Format

a data frame of 16 runs in four variables.

References

Lawson J, Vining G (2014). *Vdgraph: Variance dispersion graphs and Fraction of design space plots for response surface designs*. R package version 2.2-2, https://CRAN.R-project.org/package=Vdgraph.

GJ54 *Design from Goos & Jones*

Description

This data frame contains the design of Table 5.4 in Goos & Jones (2011).

Usage

```
GJ54
```

Format

a data frame of 15 runs in two variables: Time (seconds) and Temperature (Kelvin)

References

Goos, P., & Jones, B. (2011). Optimal design of experiments: a case study approach. John Wiley & Sons.

LHS *Latin Hypercube Sampling*

Description

Different versions of latin hypercube sampling (LHS): ordinary LHS, midpoint LHS, symmetric LHS or randomized symmetric LHS. LHS is a method for constructing space-filling designs. They can be more efficient for hypercuboidal design regions than other sampling methods.

Usage

```
LHS(n, m = 3, lim = c(-1, 1))

MLHS(n, m = 3, lim = c(-1, 1))

SLHS(n, m = 3, lim = c(-1, 1))

RSLHS(n, m = 3, lim = c(-1, 1))
```

Arguments

n	number of design points to generate
m	number of design factors
lim	limits of the coordinates in all dimensions

Value

Matrix with samples as rows.

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

Examples

```
set.seed(1234)
pts <- seq(-1, 1, length = 11)

# Ordinary LHS
samp <- LHS(n = 10, m = 2)
plot(samp, main = "LHS")
```

```
abline(h = pts, v = pts, col = "lightgrey")

# Midpoint LHS
samp <- MLHS(n = 10, m = 2)
plot(samp, main = "MLHS")
abline(h = pts, v = pts, col = "lightgrey")

# Symmetric LHS
samp <- SLHS(n = 10, m = 2)
plot(samp, main = "SLHS")
abline(h = pts, v = pts, col = "lightgrey")

# Randomized Symmetric LHS
samp <- RSLHS(n = 10, m = 2)
plot(samp, main = "RSLHS")
abline(h = pts, v = pts, col = "lightgrey")
```

---

meanspv *Compute Mean Spherical SPV*

Description

Computes the matrix of spherical region moments for a given model formula and a vector of radii, and uses this to calculate the mean spherical SPV for each of the radii. The function `expmat` calculates the matrix containing the exponents of each model factor within each model term as columns. Only simple formulae are allowed for. Only products of terms should be included in calls to `I`. The power operator `^` should be used instead of `sqrt`. Models should contain only monomial terms.

Usage

```
meanspv(formula, radii, FtF.inv, n)

expmat(formula)
```

Arguments

`formula`	model formula
`radii`	numeric vector or radii at which to calculate the matrix of spherical region moments
`FtF.inv`	inverse of F'F, where F is the design matrix
`n`	integer giving the number of design runs

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

Examples

```
f1 <- formula(~ x1*x2)
expmat(f1)
f2 <- update(f1, ~ . + I(x1^2) + I(x2^2))
expmat(f2)
f3 <- update(f2, ~ . + I(x2^0.4))
expmat(f3)
f4 <- update(f3, ~ . + I(x1^2):I(x2^2))
expmat(f4)
f5 <- update(f4, ~ . + I(x1^3*x2^0.5))
expmat(f5)
```

---

plot.spv *Plot VDGs or FDS plots*

Description

Produce Variance Dispersion Graphs and/or Fraction of Design Space plots for experimental designs. There are methods for the S3 classes `spv`, `spvlist`, `spvforlist` and `spvlistforlist` – see `spv`.

Usage

```
## S3 method for class 'spv'
plot(
  x,
  which = c("fds", "vdgsim", "vdgquantile", "vdgboth", "boxplots"),
  np = 50,
  alpha = 7/sqrt(length(x$spv)),
  points.colour = "#39BEB1",
  points.size = 2,
  tau = c(0.05, 0.95),
  radii = 21,
  hexbin = FALSE,
  bins = 80,
  df = 10,
  lines.size = 1,
  origin = rep(0, ncol(x$sample)),
  method,
  arrange = FALSE,
  ...
```

```
)

## S3 method for class 'spvforlist'
plot(
  x,
  which = c("fds", "vdgsim", "vdgquantile", "vdgboth", "boxplots"),
  np = 50,
  alpha = 7/sqrt(length(x[[1]]$spv)),
  points.colour = "#39BEB1",
  points.size = 2,
  tau = c(0.05, 0.95),
  radii = 21,
  hexbin = FALSE,
  bins = 80,
  df = 10,
  lines.size = 1,
  origin = rep(0, ncol(x[[1]]$sample)),
  method,
  arrange = FALSE,
  ...
)

## S3 method for class 'spvlist'
plot(
  x,
  which = c("fds", "vdgsim", "vdgquantile", "vdgboth", "boxplots"),
  np = 50,
  alpha = 7/sqrt(length(x[[1]]$spv)),
  points.colour = "#39BEB1",
  points.size = 2,
  tau = c(0.05, 0.95),
  radii = 21,
  hexbin = FALSE,
  bins = 80,
  VRFDS = FALSE,
  df = 10,
  lines.size = 1,
  origin = rep(0, ncol(x[[1]]$sample)),
  method,
  arrange = FALSE,
  ...
)

## S3 method for class 'spvlistforlist'
plot(
  x,
  which = c("fds", "vdgsim", "vdgquantile", "vdgboth", "boxplots"),
  np = 50,
```

```
    alpha = 7/sqrt(length(x[[1]][[1]]$spv)),
    points.colour = "#39BEB1",
    points.size = 2,
    tau = c(0.05, 0.95),
    radii = 21,
    hexbin = FALSE,
    bins = 80,
    df = 10,
    lines.size = 1,
    origin = rep(0, ncol(x[[1]][[1]]$sample)),
    method,
    arrange = FALSE,
    ...
)
```

Arguments

x	an object of type spv for a single experimental design or an object of type spvlist for multiple designs.
which	either a numeric vector of integers or a character vector indicating which plots to produce. The possible plots are: 1 **or** "fds" A (variance ratio) FDS plot 2 **or** "vdgsim" A VDG with only the simulated prediction variance points plotted 3 **or** "vdgquantile" A VDG with only the quantile regression lines corresponding to tau shown 4 **or** "vdgboth" A combination of 2 and 3 5 **or** "boxplots" Parallel boxplots of the prediction variance
np	scalar; the number of points to use for calculating the fraction of design space criterion.
alpha	the alpha transparency coefficient for the plots
points.colour	colour for points in scatterplot of SPV against the radius
points.size	size for points in scatterplot of SPV against the radius
tau	the tau parameter for rq (quantile regression)
radii	either a numeric vector containing the radii to use for calculating the mean spherical SPV over the spherical design space, or an integer (length one vector) giving the number of radii to use for calculationg the mean spherical SPV. If missing, the mean spherical SPV is not used.
hexbin	logical indicating whether hexagonal binning should be used to display density instead of colour transparency
bins	argument passed to stat_binhex to determine the number of hexagons used for binning.
df	degrees-of-freedom parameter passed to bs
lines.size	line size passed to geom_line

origin	numeric vector specifying the origin of the design space
method	optional; passed to dist to overwrite defaults of "Euclidean" for spherical regions or "supremum" for cubiodal regions
arrange	Logical indicating whether to return a single graphical object arranging the resulting plots in a single plot window via grid.arrange, or whether to return the list of graphical objects containing the plots.
...	additional arguments passed to dist
VRFDS	logical indicating whether to construct a variance ratio FDS plot or not (only for class spvlist). The first design is used as reference design in case of VRFDS is TRUE

Value

Returns a list of ggplot graphical objects (or grobs) with names corresponding to the character version of which. These plot objects can be manipulated by changing plot aesthetics and theme elements.

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

Examples

```
# Single design (class 'spv')
# Larger n should be used in actual cases
library(rsm)
bbd3 <- as.data.frame(bbd(3)[,3:5])
colnames(bbd3) <- paste0("x", 1:3)
quad.3f <- formula(~ x1*x2*x3 - x1:x2:x3 + I(x1^2) + I(x2^2) + I(x3^2))
set.seed(1234)
out <- spv(n = 1000, design = bbd3, type = "spherical", formula = quad.3f)
out
plot(out)

# List of designs (class 'spvlist')
## Not run:
data(SCDH5); data(SCDDL5)
des.list <- list(SCDH5 = SCDH5, SCDDL5 = SCDDL5)
quad.5f <- formula(~ x1 + x2 + x3 + x4 + x5 + x1:x2 + x1:x3 + x1:x4 + x1:x5
                    + x2:x3 + x2:x4 + x2:x5 + x3:x4 + x3:x5 + x4:x5
                   + I(x1^2) + I(x2^2) + I(x3^2) + I(x4^2) + I(x5^2))
out2 <- spv(n = 500, design = des.list, type = "spherical", formula = quad.5f)
out2
```

```
plot(out2)

## End(Not run)

# List of formulae (class 'spvforlist')
## Not run:
fact3 <- expand.grid(x1 = c(-1,1), x2 = c(-1, 1), x3 = c(-1,1))
lin.3f <- formula(~ x1 + x2 + x3)
int.3f <- formula(~ (x1+x2+x3)^2)
set.seed(4312)
out3 <- spv(n = 500, design = fact3, type = "cuboidal",
             formula = list(linear = lin.3f, interaction = int.3f))
out3
plot(out3)

## End(Not run)

# List of formulae and designs (class 'spvlistforlist')
## Not run:
fact3.n <- rbind(fact3, 0, 0, 0)
set.seed(4312)
out4 <- spv(n = 200, design = list(factorial = fact3, factorial.with.cntr = fact3.n),
             type = "cuboidal", formula = list(linear = lin.3f, interaction = int.3f))
out4
plot(out4)

## End(Not run)
```

---

print.spv *Print Method for S3* spv *classes*

Description

Simple print methods for S3 classes spv, spvlist, spvforlist and spvlistforlist. See plot.spv for examples.

Usage

```
## S3 method for class 'spv'
print(x, ...)

## S3 method for class 'spvforlist'
print(x, ...)

## S3 method for class 'spvlist'
print(x, ...)

## S3 method for class 'spvlistforlist'
print(x, ...)
```

Arguments

`x`	Object of class `spv` or `spvlist`
`...`	Unimplemented

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

`runif_cube` *Sampling for hyperspheres/hypercubes*

Description

Sample uniformly in or on a hyperspheres or hypercubes.

Usage

```
runif_cube(n, m = 2, max.dist = 1, at = FALSE, nr.dist = 21)

runif_sphere(n, m = 2, max.radius = sqrt(m), at = FALSE, nr.rad = 21)
```

Arguments

`n`	number of points to sample
`m`	number of design factors
`max.dist`	maximum distance from origin (L-infinity norm/supremum distance) for the hypercuboidal design region (enveloping hypercube)
`at`	logical indicating whether to sample on concentric hyperspheres/hypercubes or not. With this option `n` is distributed proportionally across radii / supremum distances so that the density of samples on each concentric hypercube / hypersphere are uniform across the different hyperspheres / hypercubes..
`nr.dist`	the number of concentric hypercubes to use in case at is `TRUE`
`max.radius`	maximum radius of the hyperspherical design region (enveloping hypersphere)
`nr.rad`	number of concentric hyperspheres to sample on in case of `at` being `TRUE`

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

Examples

```
set.seed(1234)
runif_sphere(n = 10)

set.seed(1234)
samp <- runif_sphere(n = 500, at = TRUE)
plot(samp, asp = 1)
```

`sampler` *Sampler Function*

Description

This is a wrapper for the sampling funcions of the **vdg** package. It extracts design properties from the design passed to it to take appropriate samples.

Usage

```
sampler(
  n,
  design,
  type = c("spherical", "cuboidal", "lhs", "mlhs", "slhs", "rslhs", "custom"),
  at = FALSE,
  custom.fun = NULL,
  ...
)
```

Arguments

`n`	number of points to sample
`design`	design for which the sample is required (either a matrix or data frame)
`type`	type of design region/sampling method. One of "spherical", "cuboidal", "lhs", "mlhs", "slhs", "rslhs" or "custom". Option "custom" requires `custom.fun` to be non-`NULL`.
`at`	logical; should sampling be done on the surface of hyperspheres or hypercubes? Not used for LHS methods.
`custom.fun`	A custom sampling function, used in conjunction with `type = "custom"`. The first and second arguments must be the sample size and dimension respectively.
`...`	other arguments passed to the underlying sampling functions.

Value

Matrix with samples as rows, with S3 class smpl

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

See Also

runif_sphere, runif_cube, LHS, MLHS, SLHS, RSLHS

Examples

```
## Default spherical design region
set.seed(1896)
samp1 <- sampler(n = 100, design = expand.grid(x = -1:1, y = -1:1))
plot(samp1)

## Supplying a custom sampling function based on lhs::improvedLHS()
library("lhs")
sfun <- function(n, k, dup = 1) 2 * improvedLHS(n, k, dup = dup) - 1
samp2 <- sampler(n = 100, design = expand.grid(x = -1:1, y = -1:1),
                 type = "custom", custom.fun = sfun)
plot(samp2)
```

SCDDL5 *Design from the Vdgraph package*

Description

This data frame is taken verbatim from the (now archived) **Vdgraph** package. See that package for the original reference.

Usage

SCDDL5

Format

a data frame of 23 runs in five variables.

References

Lawson J, Vining G (2014). *Vdgraph: Variance dispersion graphs and Fraction of design space plots for response surface designs*. R package version 2.2-2, https://CRAN.R-project.org/package=Vdgraph.

SCDH5 — *Design from the Vdgraph package*

Description

This data frame is taken verbatim from the (now archived) **Vdgraph** package. See that package for the original reference.

Usage

```
SCDH5
```

Format

a data frame of 28 runs in four variables.

References

Lawson J, Vining G (2014). *Vdgraph: Variance dispersion graphs and Fraction of design space plots for response surface designs*. R package version 2.2-2, https://CRAN.R-project.org/package=Vdgraph.

---

spv — *Calculate the Scaled Prediction Variance (or SPV)*

Description

Calculates the SPV for a sample of points in a design region of specified type. Sampling is done by calling sampler.

Usage

```
spv(
  n,
  design,
  type = "spherical",
  formula,
  at = FALSE,
  keepfun,
  sample,
```

```
  unscaled = FALSE,
  ...
)

## S3 method for class 'data.frame'
spv(
  n,
  design,
  type = c("spherical", "cuboidal", "lhs", "mlhs", "slhs", "rslhs", "custom"),
  formula,
  at = FALSE,
  keepfun,
  sample,
  unscaled = FALSE,
  ...
)

## S3 method for class 'list'
spv(
  n,
  design,
  type = c("spherical", "cuboidal", "lhs", "mlhs", "slhs", "rslhs", "custom"),
  formula,
  at = FALSE,
  keepfun,
  sample,
  unscaled = FALSE,
  ...
)

## S3 method for class 'matrix'
spv(
  n,
  design,
  type = c("spherical", "cuboidal", "lhs", "mlhs", "slhs", "rslhs", "custom"),
  formula,
  at = FALSE,
  keepfun,
  sample,
  unscaled = FALSE,
  ...
)
```

Arguments

n	number of samples to take
design	a design or list of designs. Each design must be either a matrix or a data.frame or coercible to a data.frame.

type	type of sampling passed to sampler
formula	either a single model formula of a list of formulae
at	only used when type is 'spherical' or 'cuboidal'
keepfun	optional; function operating on the columns of a matrix with the same number of columns as design which return a logical value for including a specific point in the sample or not. Useful for rejection sampling for nonstandard design regions.
sample	optional; if not missing it should contain a matrix or data.frame containing points sampled over the required design region. If it is not missing, no further sampling will be done: the SPV is simply evaluated at these points.
unscaled	logical indicating whether to use the unscaled prediction variance (UPV) instead of the scale prediction variance (SPV)
...	additional arguments passed to sampler. This enables the used of user-specified sampling functions via the custom.fun argument to sampler.

Value

Object of class 'spv', 'spvlist', 'spvforlist' or 'spvlistforlist', depending on whether single designs/formulas are passed or lists of these.

Author(s)

Pieter C. Schoonees

References

Pieter C. Schoonees, Niel J. le Roux, Roelof L.J. Coetzer (2016). Flexible Graphical Assessment of Experimental Designs in R: The vdg Package. *Journal of Statistical Software*, 74(3), 1-22. doi:10.18637/jss.v074.i03.

See Also

plot.spv for more examples

Examples

```
# Single design (class 'spv')
library(rsm)
bbd3 <- as.data.frame(bbd(3)[,3:5])
colnames(bbd3) <- paste0("x", 1:3)
quad.3f <- formula(~(x1 + x2 + x3)^2 - x1:x2:x3)
out <- spv(n = 1000, design = bbd3, type = "spherical", formula = quad.3f)
out
```

`stdrange` *Standardize or Unstandarize the Column Range*

Description

Simple functions for rescaling a data matrix to a coded design and back. `stdrange` converts the design in actual measurements into a coded design, while `ustdrange` reverses the process (if the correct arguments are given).

Usage

```
stdrange(x, mins = apply(x, 2, min), maxs = apply(x, 2, max))

ustdrange(x, mins, maxs)
```

Arguments

`x`	matrix containing the design, or an object coercible to a matrix.
`mins`	vector of original values, one for each column, which should be recoded to the value `-1`; or which have alreadty been recoded to `-1`. This and the next argument are both recycled if not of the correct length.
`maxs`	vector of original values which should be recoded as 1, or which have already been recoded to 1.

Author(s)

Pieter C. Schoonees

Index