Package 'OptimaRegion'

January 20, 2025

Type Package

Title Confidence Regions for Optima of Response Surfaces

Version 1.2

Author Enrique del Castillo, Peng Chen, Adam Meyers, John Hunt, and James Rapkin

Maintainer Enrique del Castillo <exd13@psu.edu>

Depends R (>= 3.3.0)

Imports boot, DepthProc, fields, magrittr, nloptr, rsm, Rdpack, Rdsdp, spam, stringr

Suggests knitr, lhs, rmarkdown

RdMacros Rdpack

Description Computes confidence regions on the location of response surface optima. Response surface models can be up to cubic polynomial models in up to 5 controllable factors, or Thin Plate Spline models in 2 controllable factors.

License GPL-3

LazyData true

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation no

Repository CRAN

Date/Publication 2023-02-18 00:20:02 UTC

Contents

OptimaRegion-package   *OptimaRegion package description*

Description

OptimaRegion is a package for the computation of confidence regions on the location of the optima of response surface models (Del Castillo et al. 2020). Both parametric (quadratic polynomial) and nonparametric (thin plate spline) models are supported. The confidence regions obtained do not rely on any distributional assumption, such as Normality of the response, and are obtained by bootstrapping. The resulting regions are both valid and unbiased, and have a size that rapidly decreases as the sample size increases. Regions are obtained both numerically (as a set of points) and graphically, as the convex hull of the points. Functionality for the computation of a bootstrap confidence interval on the distance between the optima of two different response surfaces is included.

Details

The OptimaRegion package provides five main functions: OptRegionQuad, OptRegionTps, GloptiPolyRegion, CRcompare and GloptipolyR.

OptRegionQuad computes distribution-free bootstrapped confidence regions for the location of the optima of a quadratic polynomial model in 2 regressors. OptRegionTps computes distribution-free bootstrapped confidence regions for the location of the optima of a Thin Plate Spline model in 2 regressors. GloptiPolyRegion computes distribution-free bootstrapped CRs for the location of global optima for polynomial models up to cubic order in up to 5 regressors. CRcompare computes bootstrapped confidence intervals for the distance between the optima of two different response surface models, either quadratic polynomials or thin plate spline models. GloptipolyR is an R implementation of the "Gloptipoly" algorithm (Lasserre 2001) for global optimization of polynomial equations subject to bounds.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Lasserre JB (2001). "Global optimization with polynomials and the problem of moments." *SIAM Journal on optimization*, **11**(3), 796–817.

CRcompare   *Confidence interval for the distance between two response surface op-*
                             tima (2 regressors)

Description

Computes bootstrapped confidence intervals for the mean and median distance between the optima of two response surface models. Models can be thin plate splines or quadratic polynomials (Del Castillo et al. 2020).

Usage

```
CRcompare(
  X1,
  y1,
  X2,
  y2,
  responseType = "TPS",
  lambda = 0.04,
  nosim1and2 = 200,
  alpha = 0.05,
  LB1,
  LB2,
  UB1,
  UB2,
  triangularRegion1 = FALSE,
  vertex11 = NULL,
  vertex21 = NULL,
  triangularRegion2 = FALSE,
  vertex12 = NULL,
  vertex22 = NULL,
  maximization1 = TRUE,
  maximization2 = TRUE,
  xlab1and2 = "Protein eaten (mg)",
  ylab1and2 = "Carbohydrates eaten (mg)",
  outputPDFFile1 = "CR_plot1.pdf",
  outputOptimaFile1 = "Optima1.txt",
  outputPDFFile2 = "CR_plot2.pdf",
  outputOptimaFile2 = "Optima2.txt"
)
```

Arguments

X1	nx2 matrix with the values of the 2 regressors (experimental factors) corresponding to the first response. Note: can have replicates. They will be eliminated by the program and the corresponding y-values averaged
y1	nx1 vector of values for the first response corresponding to X1

X2	nx2 matrix with the values of the 2 regressors (experimental factors) corresponding to the second response. Note: can have replicates. They will be eliminated by the program and the corresponding y-values averaged
y2	nx1 vector of values for the second response corresponding to X2
responseType	use 'TPS' if fitting thin plate spline responses, 'Quad' if fitting quadratic polynomials
lambda	psmoothing penalty if a TPS model is selected (default=0.04)
nosim1and2	number of simulations(default = 200) used to find each of the two confidence regions of optima
alpha	confidence level ($0 < $ alpha $ < 1$; default = 0.05)
LB1	vector of lower bounds for x (2x1 vector) above which the optimum is sought for the first response
LB2	vector of lower bounds for x (2x1 vector) above which the optimum is sought for the second response
UB1	vector of upper bounds for x (2x1 vector) below which the optimum is sought for the first response
UB2	vector of upper bounds for x (2x1 vector) below which the optimum is sought for the second response
triangularRegion1	
	logical: if TRUE it will constrain the maximum points of response 1 to lie inside a triangle defined by the coordinates (0,0), and those in "vertex11", and "vertex21", see below (in addition to being constrained to lie inside the region defined by LB1 and UB1). NOTE: use TRUE when the treatments form a triangular experimental region in shape. If FALSE, optima will only be constrained to lie inside the rectangular region defined by LB1 and UB1. Default is FALSE.
vertex11	2 times 1 vector with coordinates defining one of the 3 vertices of the triangular region where the first response is being optimized. Must be provided if triangularRegion1 is TRUE (NOTE: vertices numbered clockwise, with vertex0 fixed to (0,0))
vertex21	2 times 1 vector with coordinates defining a second vertex of a triangular region where the first response is being optimized. Must be provided if triangularRegion1 is TRUE
triangularRegion2	
	logical: if TRUE it will constrain the maximum points of response 2 to lie inside a triangle defined by the coordinates (0,0), and those in "vertex12", and "vertex22", see below (in addition to being constrained to lie inside the region defined by LB2 and UB2).NOTE: use TRUE when the treatments form a triangular experimental region in shape. If FALSE, optima will only be constrained to lie inside the rectangular region defined by LB2 and UB2. Default is FALSE.
vertex12	2 times 1 vector with coordinates defining one of the 3 vertices of the triangular region where the second response is being optimized. Must be provided if triangularRegion2 is TRUE (NOTE: vertices numbered clockwise, with vertex0 fixed to (0,0))

vertex22	2 times 1 vector with coordinates defining a second vertex of a triangular region where the second response is being optimized. Must be provided if triangularRegion2 is TRUE
maximization1	logical: if TRUE (default) it maximizes response 1, if FALSE it minimizes it
maximization2	logical: if TRUE (default) it maximizes response 2, if FALSE it minimizes it
xlab1and2	text label for x axis in both confidence region plots (default: "Protein eaten (mg)")
ylab1and2	text label for y axis in both confidence region plots (default: "Carbohydrates eaten (mg)")
outputPDFFile1	name of the PDF file where the CR plot of the first response is saved (default: "CR_plot.pdf")
outputOptimaFile1	name of the text file containing the coordinates of all the simulated optima of the first response
outputPDFFile2	name of the PDF file where the CR plot of the second response is saved (default: "CR_plot.pdf")
outputOptimaFile2	name of the text file containing the coordinates of all the simulated optima of the second response

Details

Computes distribution-free bootstrapped confidence intervals on the mean and median distance between the optima of two different responses. The responses can be Thin Plate Spline models or Quadratic polynomial models. Program calls each response, next computes all pairwise distances between points in each CR, and finally bootstraps the distances to compute bca bootstrapped confidence intervals for the mean and median distance.

Value

Upon completion, two PDF files with the CR plots and two text files with the coordinates of each set of optima are created, and the function also returns a list consisting of the following 5 components:

dist vector of distances between pairs of points taken from each set of optima

mean mean of dist

median median of dist

ciMean 95 bca bootstrapping; it is a vector with 5 columns, containing the signicance level, the next two containing the indices of the order statistics used in the calculations and the final two the calculated endpoints of the CI's.

ciMEdian 95 using bca bootstrapping; it is a vector with 5 columns, containing the signicance level, the next two containing the indices of the order statistics used in the calculations and the final two the calculated endpoints of the CI's.

Author(s)

Enrique del Castillo <exd13@psu.edu>, Peng Chen <pfc5098@psu.edu>, Adam Meyers <akm5733@psu.edu>, John Hunt <J.Hunt@westernsydney.edu.au> and James Rapkin <jr297@exeter.ac.uk>.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Examples

```
## Not run:
# Example: two randomly generated data sets, quadratic polynomial responses.
X1 <- cbind(runif(100, -2, 2), runif(100, -2, 2))
y1 <- as.matrix(72 - 11.78 * X1[, 1] + 0.74 * X1[, 2] - 7.25 * X1[, 1]^2 -
  7.55 * X1[, 2]^2 - 4.85 * X1[, 1] * X1[, 2] + rnorm(100, 0, 8))
X2 <- cbind(runif(100, -2, 2), runif(100, -2, 2))
y2 <- as.matrix(72 - 11.78 * X2[, 1] + 0.74 * X2[, 2] - 7.25 * X2[, 1]^2 -
  7.55 * X2[, 2]^2 - 4.85 * X2[, 1] * X2[, 2] + rnorm(100, 0, 8))
out <- CRcompare(
  X1 = X1, y1 = y1, X2 = X2, y2 = y2, responseType = "Quad", nosim1and2 = 200,
  alpha = 0.05, LB1 = c(-2, -2), UB1 = c(2, 2), LB2 = c(-2, -2), UB2 = c(2, 2)
)

## End(Not run)
```

---

cubic_5D   *Simulated dataset based on a cubic function (5 factors)*

Description

The dataset is simulated from the following function

$$f(x) = 10 - (x_1 - 1.5)^2 - (x_2 - 2)^2 - (x_3 - 2.5)^2 - (x_4 - 3)^2 - (x_5 - 3.5)^2 + 0.1x_1^3 - 0.1x_2^3 - 0.1x_3^3 - 0.1x_4^3 - 0.1x_5^3 + x_2x_4 - x_3x_4$$

defined in the region R = { 0 <= x_i <= 5, for i = 1, ..., 5}, with its maximum at (2.28, 2.44, 1.02, 2.65, 2.54). The sample locations are generated via a Hyper Latin Square (HLS) design within R, and the noisy level is sigma = 2.

Usage

```
cubic_5D
```

Format

A list consisting of 2 components:

design_matrix 300 design points generated by a HLS design within R

responses 300 noisy responses simulated at the design points with sigma = 2

Drug *Mixture-amount experiment dataset (2 factors)*

Description

A pharmaceutical mixture-amount experiment in two components

Usage

```
Drug
```

Format

A data frame with 360 observations on the following 3 variables.

Component_1 Component 1 amount (mg)

Component_2 Component 2 amount (mg)

Percent Percent of cells killed (response)

Examples

```
plot(Drug[,1:2])
```

GloptiPolyR *Global optimization of up to cubic polynomial functions (up to 5 variables)*

Description

Optimize a quadratic or cubic polynomial functionin 2 ~ 5 variables with bound constraints (Del Castillo et al. 2020).

Usage

```
GloptiPolyR(P)
```

Arguments

P   A list of list; Each sub-list has 2 components: 1. a multi-dimensional array corresponding to a objective or constraint function 2. an attribute of the objective or constraint function

Details

GloptipolyR is an R implementation of the "Gloptipoly" algorithm (Lasserre 2001)

Value

Returns the optimal solution and its corresponding objective value

Author(s)

Enrique del Castillo <exd13@psu.edu>, Peng Chen <pfc5098@psu.edu>, Adam Meyers <akm5733@psu.edu>, John Hunt <J.Hunt@westernsydney.edu.au> and James Rapkin <jr297@exeter.ac.uk>.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Lasserre JB (2001). "Global optimization with polynomials and the problem of moments." *SIAM Journal on optimization*, **11**(3), 796–817.

Examples

```
# Optimize the following quadratic function in 3 variables
# f(x) = -1.5 x_1 + 2.13 x_2 - 1.81 x_3 + 7.13 x_1 x_2 +
#        3.27 x_1 x_3  + 2.73 x_2 x_3 +
#        4.69 x_1^2 + 6.27 x_2^2 + 5.21 x_3^2.

# The input for GloptiPolyR is a list of 7 sub-lists,
# each of which corresponds to the objective function or a constraint
# function, respectively. See del Castillo et al. (2019) for details.
P <- list()
p_f <- list()
p_g_1 <- list()
p_g_2 <- list()
p_g_3 <- list()
p_g_4 <- list()
p_g_5 <- list()
p_g_6 <- list()

p_f$c <- array(0, dim = c(3, 3, 3))
p_f$c[2, 1, 1] <- -1.5
p_f$c[1, 2, 1] <- 2.13
p_f$c[1, 1, 2] <- -1.81
p_f$c[2, 2, 1] <- 7.13
p_f$c[2, 1, 2] <- 3.27
p_f$c[1, 2, 2] <- 2.73
p_f$c[3, 1, 1] <- 4.69
p_f$c[1, 3, 1] <- 6.27
p_f$c[1, 1, 3] <- 5.21

p_g_1$c <- array(0, dim = c(3, 3, 3))
p_g_1$c[1, 1, 1] <- 2
p_g_1$c[2, 1, 1] <- 1
```

```
p_g_2$c <- array(0, dim = c(3, 3, 3))
p_g_2$c[1, 1, 1] <- -2
p_g_2$c[2, 1, 1] <- 1

p_g_3$c <- array(0, dim = c(3, 3, 3))
p_g_3$c[1, 1, 1] <- 2
p_g_3$c[1, 2, 1] <- 1

p_g_4$c <- array(0, dim = c(3, 3, 3))
p_g_4$c[1, 1, 1] <- -2
p_g_4$c[1, 2, 1] <- 1

p_g_5$c <- array(0, dim = c(3, 3, 3))
p_g_5$c[1, 1, 1] <- 2
p_g_5$c[1, 1, 2] <- 1

p_g_6$c <- array(0, dim = c(3, 3, 3))
p_g_6$c[1, 1, 1] <- -2
p_g_6$c[1, 1, 2] <- 1

# Set the attribute for the objective function as either ``min'' or ``max''.
p_f$t <- "min"

# Set the attributes for the constraint functions as either ``>='' or ``<=''.
p_g_1$t <- ">="
p_g_2$t <- "<="
p_g_3$t <- ">="
p_g_4$t <- "<="
p_g_5$t <- ">="
p_g_6$t <- "<="

# Now we put together the input P and use it to call GloptiPolyR
P <- list(p_f, p_g_1, p_g_2, p_g_3, p_g_4, p_g_5, p_g_6)
GloptiPolyR(P)
```

GloptiPolyRegion   *Confidence region for optima of higher order polynomial models in multiple factors*

Description

Computes and displays an approximated (1 - alpha)*100% confidence region (CR) for the bound-constrained optimum of a fitted polynomial regression model of up to cubic order with up to 5 controllable factors (Del Castillo et al. 2020).

Usage

```
GloptiPolyRegion(
  X,
  y,
```

```
  degree,
  lb,
  ub,
  B = 200,
  alpha = 0.05,
  maximization = TRUE,
  axes_labels = NULL,
  outputPDFFile = "CRplot.pdf",
  verbose = TRUE,
  local_plot = FALSE
)
```

Arguments

X	an N*k numeric matrix where N is the number of experiments and k is the number of variables/factors (an integer in [2,5]). X specifies the design matrix
y	N*1 numeric vector that specifies the responses
degree	integer scalar; degree specifies the order of the polynomial model, which can be 2 or 3
lb	numeric vector of dimension (1, k); lb specifies the lower bounds for the k variables
ub	1*k numeric vector that specifies the upper bounds for the k variables
B	integer scalar; B specifies the number of bootstrap operations
alpha	numeric scalar between 0 and 1; alpha specifies the nominal confidence level, (1 - alpha)*100%, of the confidence region
maximization	boolean scalar; specifies whether the algorithm computes the confidence region for a maximum or a minimun
axes_labels	vector of strings; specifies the name of each variable or experimental factor to be displayed on the CR plot; the default value is NULL, which sets labels as "x1", "x2", etc.
outputPDFFile	name of the PDF file where the CR plot is saved (default: "CR_plot.pdf")
verbose	boolean scalar; specifies whether or not to display status of the bootstrapping/optimization iterations
local_plot	boolean scalar; specifies whether or to display the confidence region on the screen

Value

Upon completion, a pdf file with the plot displaying the confidence region of the global optimum projected onto each pairwise-variable planes. If local_plot = TRUE, the plot will also be created on the screen. The function also returns a list consisting of 2 components:

boot_optima numeric matrix of dimension ((1 - alpha)*B, k); it contains the (1 - alpha)*B bootstrap optima

bagged_optimum numeric vector of dimension (1, k) containing the bagged optimum, computed by taking the column average of boot_optima

Author(s)

Enrique del Castillo <exd13@psu.edu>, Peng Chen <pfc5098@psu.edu>, Adam Meyers <akm5733@psu.edu>, John Hunt <J.Hunt@westernsydney.edu.au> and James Rapkin <jr297@exeter.ac.uk>.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Examples

```
## Not run:
# Example 1: run GloptiPolyRegion on a quadratic, 3 vars example
out <- GloptiPolyRegion(
  X = quad_3D[, 1:3], y = quad_3D[, 4], degree = 2,
  lb = c(-2, -2, -2), ub = c(2, 2, 2), B = 500, alpha = 0.1,
  maximization = TRUE,
  outputPDFFile = "CR_quad_3D.pdf", verbose = TRUE
)
# check result
str(out)

# Example 2: run GloptiPolyRegion on a cubic, 5 vars example
out <- GloptiPolyRegion(
  X = cubic_5D$design_matrix, y = cubic_5D$response,
  degree = 3, lb = rep(0, 5), ub = rep(5, 5), B = 200,
  alpha = 0.05, maximization = TRUE,
  outputPDFFile = "CR_cubic_5D.pdf", verbose = TRUE
)
# check result
str(out)

## End(Not run)
```

OptRegionQuad *Confidence region for optima of quadratic polynomial models (2 regressors)*

Description

Computes and displays an approximated (1 - alpha)*100 the linear-constrained maximum of a quadratic polynomial regression model in 2 controllable factors (Del Castillo et al. 2020). Grey region on output plot is the approximate CR. The CR is computed as the convex hull of the coordinates of the optima found from simulating and optimizing nosim quadratic polynomial regressions from the data (therefore, it is an approximate CR). The mean value of the optimum is shown as a red point, and a smoothed contour plot of the X,y data obtained via thin plate splines is shown as well.

Usage

```
OptRegionQuad(
  X,
  y,
  nosim = 200,
  alpha = 0.05,
  LB,
  UB,
  triangularRegion = FALSE,
  vertex1 = NULL,
  vertex2 = NULL,
  maximization = TRUE,
  xlab = "Protein eaten, mg",
  ylab = "Carbohydrates eaten, mg",
  outputPDFFile = "CRplot.pdf"
)
```

Arguments

X	n*2 matrix with the values of the 2 regressors (experimental factors) in the n observations. Note: this can have replicates. They will be eliminated by the program and the corresponding y-values averaged
y	n*1 vector of response value observations, in the same order corresponding to the rows of X
nosim	number of simulations (default = 200)
alpha	confidence level (0 < alpha < 1; default = 0.05)
LB	vector of lower bounds for x (2*1 vector) above which the maximum is sought
UB	vector of upper bounds for x (2*1 vector) below which the maximum is sought
triangularRegion	
	logical: if TRUE it will constrain the maximum points to lie inside a triangle defined by the coordinates (0,0), and those in 'vertex1', and 'vertex2', see below (in addition to being constrained to lie inside the region defined by LB and UB). NOTE: use TRUE when the treatments form a triangular experimental region in shape. If FALSE, maxima will only be constrained to lie inside the rectangular region defined by LB and UB. Default is FALSE.
vertex1	2 times 1 vector with coordinates defining one of the 3 vertices of a triangular region. Must be provided if triangularRegion is TRUE (NOTE: vertices numbered clockwise)
vertex2	2 times 1 vector with coordinates defining a second vertex of a triangular region (third vertex is (0,0) by default). Must be provided if triangularRegion is TRUE (NOTE: vertices numbered clockwise)
maximization	logical: if TRUE (default) it maximizes it FALSE it minimizes
xlab	text label for x axis in confidence region plot (default: "Protein eaten (mg)")
ylab	text label for y axis in confidence region plot (default: "Carbohydrates eaten (mg)")
outputPDFFile	name of the PDF file where the CR plot is saved (default: "CR_plot.pdf")

Details

This program approximates the confidence region (CR) of the location of the optimum of a regression function in 2 regressors constrained inside a rectangular region defined by LB and UB. If triangularRegion = TRUE it will also contrain the optimum to lie inside the experimental region (assumed to be well approximated by a triangle). The CR is generated pointwise by simulating from the posterior of the regression parameters and solving the corresponding constrained maximization problem. The confidence region is approximated by the convex hull of all the solutions found. The simulation approach is based on the "CS" bootstrapping approach for building a confidence set described in Woutersen and Ham (2013). This version of the program uses nonparamteric bootstrapping confidence regions to get the posteazrior of the parameters of the regression equation using the notion of data depth according to Yeh and Singh (1997). Hence, this version does not rely on any normality assumption on the data.

Value

Upon completion, a PDF file containing the CR plot with name as set in ouputPDFFile is created and the function also returns a list containing the following 2 components:

meanPoint a 2x1 vector with the coordinates of the mean optimum point (displayed as a red dot in the CR plot in output PDF file)

xin an mx2 matrix with the x,y coordinates of all simulated points that belong to the confidence region (dim(m) is (1-alpha)*nosim)

Author(s)

Enrique del Castillo <exd13@psu.edu>, Peng Chen <pfc5098@psu.edu>, Adam Meyers <akm5733@psu.edu>, John Hunt <J.Hunt@westernsydney.edu.au> and James Rapkin <jr297@exeter.ac.uk>.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Woutersen T, Ham JC (2013). "Confidence sets for Continuous and Discontinuous functions of parameters." *Technical report, University of Arizona*.

Yeh AB, Singh K (1997). "Balanced confidence regions based on Tukey's depth and the bootstrap." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **59**(3), 639–652.

Examples

```
## Not run:
# Example 1: randomly generated 2-variable response surface data
X <- cbind(runif(100, -2, 2), runif(100, -2, 2))
y <- as.matrix(72 - 11.78 * X[, 1] + 0.74 * X[, 2] - 7.25 * X[, 1]^2 - 7.55 * X[, 2]^2 -
  4.85 * X[, 1] * X[, 2] + rnorm(100, 0, 8))
# Find a 95 percent confidence region for the maximum of a quadratic polynomial
# fitted to these data
```

```
out <- OptRegionQuad(
  X = X, y = y, nosim = 200, LB = c(-2, -2), UB = c(2, 2),
  xlab = "X1", ylab = "X2"
)

# Example 2: a mixture-amount experiment in two components (Drug dataset) with
# non-normal data. Note triangular experimental region. Resulting 95%
# confidence region is pushed against the constraint and results in a
# "thin line"
out <- OptRegionQuad(
  X = Drug[, 1:2], y = Drug[, 3], nosim = 500,
  LB = c(0, 0), UB = c(0.08, 11), xlab = "Component 1 (mg.)",
  ylab = "Component 2 (mg.)", triangularRegion = TRUE,
  vertex1 = c(0.02, 11), vertex2 = c(0.08, 1.8), outputPDFFile = "Mixture_plot.pdf"
)

## End(Not run)
```

OptRegionTps   *Confidence region for optima of Thin Plate Spline Models (2 regressors)*

Description

Computes and displays an approximated (1 - alpha)*100 the linear-constrained maximum of a penalized Thin Plate Spline (TPS) model in 2 controllable factors (Del Castillo et al. 2020). Generates a PDF file with a graph displaying the CR. Grey region on output plot is the approximate CR. The mean coordinates (centroid) of the optima is shown as a red point.

Usage

```
OptRegionTps(
  X,
  y,
  lambda = 0.04,
  nosim = 1000,
  alpha = 0.05,
  LB,
  UB,
  triangularRegion = FALSE,
  vertex1 = NULL,
  vertex2 = NULL,
  maximization = TRUE,
  xlab = "Protein eaten, mg",
  ylab = "Carbohydrate eaten, mg",
  outputPDFFile = "CRplot.pdf",
  outputOptimaFile = "Optima.txt"
)
```

Arguments

X	n*2 matrix with the values of the 2 regressors (experimental factors) in the n observations. Note: this can have replicates. They will be eliminated by the program and the corresponding y-values averaged
y	n*1 vector of response value observations, in the same order corresponding to the rows of X
lambda	penalization parameter (larger values implies more smoothing). Default is 0.04
nosim	number of simulations (default = 200)
alpha	confidence level (0 < alpha < 1; default = 0.05)
LB	vector of lower bounds for x (2*1 vector) above which the maximum is sought
UB	vector of upper bounds for x (2*1 vector) below which the maximum is sought
triangularRegion	
	logical: if TRUE it will constrain the maximum points to lie inside a triangle defined by the coordinates (0,0), and those in 'vertex1', and 'vertex2', see below (in addition to being constrained to lie inside the region defined by LB and UB). NOTE: use TRUE when the treatments form a triangular experimental region in shape. If FALSE, maxima will only be constrained to lie inside the rectangular region defined by LB and UB. Default is FALSE.
vertex1	2 times 1 vector with coordinates defining one of the 3 vertices of a triangular region. Must be provided if triangularRegion is TRUE (NOTE: vertices numbered clockwise)
vertex2	2 times 1 vector with coordinates defining a second vertex of a triangular region (third vertex is (0,0) by default). Must be provided if triangularRegion is TRUE (NOTE: vertices numbered clockwise)
maximization	logical: if TRUE (default) it maximizes it FALSE it minimizes
xlab	text label for x axis in confidence region plot (default: "Protein eaten (mg)")
ylab	text label for y axis in confidence region plot (default: "Carbohydrates eaten (mg)")
outputPDFFile	name of the PDF file where the CR plot is saved (default: "CR_plot.pdf")
outputOptimaFile	
	name of the text file containing the coordinates of all the optima found (same information as in output vector xin, see below)

Details

This program approximates the confidence region (CR) of the location of the optimum of a Thin Plate Spline (TPS) in 2 regressors x constrained inside a rectangular region defined by LB and UB. If triangularRegion=TRUE it will also contrain the optimum to lie inside the experimental region assumed to be well approximated by a triangle. The CR is generated pointwise by bootstrapping the residuals of a TPS fit to the given (X,y) data, refitting Tps models, and solving the corresponding constrained maximization (or minimization) problems. The confidence region is approximated by the convex hull of all the optimal solutions found. The CR computation is based on the "CS" boot-strapping approach for building a confidence set of a parametric function described in Woutersen and Ham (2013). This version of the program uses nonparametric bootstrapping confidence regions

to get the Confidence region of the Tps parameters,using the notion of data depth according to Yeh and Singh (1997). Hence, this version does not rely on the normality assumption of the data. The TPS models are fit using the "fields" R package (Douglas Nychka et al. 2017) and its "Tps" function.

Value

Upon completion, a PDF file containing the CR plot with name as set in ouputPDFFile is created and a text file with all optima in the CR is created too. Also, the function returns a list containing the following 2 components:

meanPoint a 2x1 vector with the coordinates of the mean optimum point (displayed as a red dot in the CR plot in output PDF file)

xin an mx2 matrix with the x,y coordinates of all simulated points that belong to the confidence region (dim(m) is (1-alpha)*nosim)

Author(s)

Enrique del Castillo <exd13@psu.edu>, Peng Chen <pfc5098@psu.edu>, Adam Meyers <akm5733@psu.edu>, John Hunt <J.Hunt@westernsydney.edu.au> and James Rapkin <jr297@exeter.ac.uk>.

References

Del Castillo E, Chen P, Meyers A, Hunt J, Rapkin J (2020). "Confidence regions for the location of response surface optima: the R package OptimaRegion." *Communications in Statistics-Simulation and Computation*, 1–21.

Douglas Nychka, Reinhard Furrer, John Paige, Stephan Sain (2017). "fields: Tools for spatial data." doi:10.5065/D6W957CT, R package version 9.8-3, https://github.com/NCAR/Fields.

Woutersen T, Ham JC (2013). "Confidence sets for Continuous and Discontinuous functions of parameters." *Technical report, University of Arizona*.

Yeh AB, Singh K (1997). "Balanced confidence regions based on Tukey's depth and the bootstrap." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **59**(3), 639–652.

Examples

```
## Not run:
# Example 1: randomly generated 2-variable response surface data
X <- cbind(runif(100, -2, 2), runif(100, -2, 2))
y <- as.matrix(72 - 11.78 * X[, 1] + 0.74 * X[, 2] - 7.25 * X[, 1]^2 -
  7.55 * X[, 2]^2 - 4.85 * X[, 1] * X[, 2] + rnorm(100, 0, 8))
# Find a 95 percent confidence region for the maximum of a Thin Plate Spline
# model fitted to these data
out <- OptRegionTps(
  X = X, y = y, nosim = 200, LB = c(-2, -2), UB = c(2, 2),
  xlab = "X1", ylab = "X2"
)
```

```
# Example 2: a mixture-amount experiment in two components (Drug dataset) with
# non-normal data. Note triangular experimental region. Resulting 95p confidence
# region of the maxima of a TPS model has area > 0. Contrast with region for
# quadratic polynomial model. Note: 500 bootstrap iterations may take a few minutes.
out <- OptRegionTps(
  X = Drug[, 1:2], y = Drug[, 3], nosim = 500, lambda = 0.05, LB = c(0, 0),
  UB = c(0.08, 11), xlab = "Component 1 (mg.)", ylab = "Component 2 (mg.)",
  triangularRegion = TRUE, vertex1 = c(0.02, 11), vertex2 = c(0.08, 1.8),
  outputPDFFile = "Mixture_plot.pdf"
)

## End(Not run)
```

---

quad_3D   *Central composite design experiment dataset (3 factors)*

Description

Box and Draper (1987)'s three factor experimental dataset

Usage

```
quad_3D
```

Format

A data frame with 16 observations on the following 4 variables

x1 percentage concentration of the 1st constituent, in coded unit

x2 percentage concentration of the 2nd constituent, in coded unit

x3 temperature, in coded unit

y elasticity of certain polymer

References

Box GEP, Draper NR (1987). *Empirical Model Building and Response Surfaces.*. John Wiley and Sons, New York,NY.

Index