Package ‘DynamicGP’

November 8, 2022

Title Modelling and Analysis of Dynamic Computer Experiments
Version 1.1-9
Description Emulating and solving inverse problems for dynamic computer experiments. It contains two major functionalities: (1) localized GP model for large-scale dynamic computer experiments using the algorithm proposed by Zhang et al. (2018) <arXiv:1611.09488>; (2) solving inverse problems in dynamic computer experiments. The current version only supports 64-bit version of R.
License GPL (>= 2)
Date 2022-11-07
Maintainer Ru Zhang <heavenmarshal@gmail.com>
Depends R (>= 2.14)
Imports lhs, parallel, stats
NeedsCompilation yes
Author Ru Zhang [aut, cre], Chunfang Devon Lin [aut], Pritam Ranjan [aut], Robert B Gramacy [ctb], Nicolas Devillard [ctb], Jorge Nocedal [ctb], Jose Luis Morales [ctb], Ciyou Zhu [ctb], Richard Byrd [ctb], Peihuang Lu-Chen [ctb], Berend Hasselman [ctb], Jack Dongarra [ctb], Jeremy Du Croz [ctb], Sven Hammarling [ctb], Richard Hanson [ctb], University of Chicago [cph], University of California [cph]
Repository CRAN
Date/Publication 2022-11-08 10:10:09 UTC
**R topics documented:**

- **DynamicGP-package** .................................................. 2
- ESL2D ................................................................. 3
- knnsvdGP ............................................................ 5
- lasvdGP ............................................................... 7
- saEI ................................................................. 10
- SL2D ................................................................. 12
- svdGP ............................................................... 14

**Index** ................................................................. 16

---

**DynamicGP-package  Modelling and Analysis of Dynamic Computer Experiments**

**Description**

For emulating dynamic computer experiments, three functions are included. The function `svdGP` fits full SVD-based GP model which is computationally demanding for large-scale dynamic computer experiments. As is well known, the time complexity of fitting a GP model is $O(N^3)$ where $N$ is the number of training/design points. Since fitting a common GP model for really large $N$ would be computationally burdensome, we fit local SVD-based GP models on a sequentially selected small neighborhood set for every test inputs. The function `knnsvdGP` fits K-nearest neighbor SVD-based GP models which selects neighborhood sets based on the Euclidean distance with respect to the test points. The function `lasvdGP` fits local approximate SVD-based GP model using the new algorithm proposed by Zhang et al. (2018).

The `lasvdGP` is an extension of the local approximate GP (laGP) model developed by Gramacy and Lee (2015) for the emulation of large-scale scalar valued computer experiments. The neighborhood selection and SVD-based GP model fitting algorithm is suitable for parallelization. We use both the R package "parallel" and the OpenMP library for this task. The parallelization can achieve nearly linear speed since the procedure on each test point is independent and identical.

For the inverse problem in dynamic computer experiments, we also provide three functions. The function `ESL2D` minimizes the expected squared $L_2$ discrepancy between the target response and the simulator outputs to estimate the solution to the inverse problem, where the expectation is taken with respect to the predictive distribution of the `svdGP` model. A naive estimation approach `SL2D` simply minimizes the squared $L_2$ discrepancy between the target response and the predicted mean response of the SVD-based GP model. The function `saEI` performs the sequential design procedure for the inverse problem. It selects the follow-up design points as per an expected improvement criterion whose values are numerically approximated by the saddlepoint approximation technique. Details of the three methods for the inverse problem are provided in Chapter 4 of Zhang (2018).

**Author(s)**

- Ru Zhang <heavenmarshal@gmail.com>,
- C. Devon Lin <devon.lin@queensu.ca>,
- Pritam Ranjan <pritam@itimdr.ac.in>
References


---

ESL2D

*Expected Squared L_2 Discrepancy Approach for Estimating the Solution to the Inverse Problem*

Description

This function fits an SVD-based GP model on the training dataset design and response matrix resp, and minimizes the expected squared $L_2$ discrepancy on the test set candidate to estimate the solution to the inverse problem. The details are provided in Chapter 4 of Zhang (2018).

Usage

ESL2D(design,resp,yobs,candidate,frac=.95,nstarts=5,
       mtype=c("zmean","cmean","lmean"),
       gstart=0.0001)

Arguments

design     An $N$ by $d$ matrix of $N$ training/design inputs.
resp       An $L$ by $N$ response matrix of design, where $L$ is the length of the time series outputs, $N$ is the number of design points.
yobs        A vector of length $L$ of the time-series valued field observations or the target response.
candidate  An $M$ by $d$ matrix of $M$ candidate points on which the estimated solution to the inverse problem is extracted.
frac        The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.
nstarts     The number of starting points used in the numerical maximization of the posterior density function. The larger nstarts will typically lead to more accurate prediction but longer computational time. The default value is 5.
mtype       The type of mean functions for the GP models. The choice "zmean" denotes zero-mean, "cmean" indicates constant-mean, "lmean" indicates linear-mean. The default choice is "zmean".
gstart     The starting number and upper bound for estimating the nugget parameter. If gstart = sqrt(.Machine$double.eps), the nugget parameter will be fixed at sqrt(.Machine$double.eps), since sqrt(.Machine$double.eps) is the lower bound of the nugget term. The default value is 0.0001.
Value

xhat  The estimated solution to the inverse problem obtained from the candidate set candidate

Author(s)

Ru Zhang <heavenmarshal@gmail.com>,
C. Devon Lin <devon.lin@queensu.ca>,
Pritam Ranjan <pritamr@iimidr.ac.in>

References


See Also

SL2D, saEI, svdGP.

Examples

```r
library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  par2 <- x[2]*16+4
  par3 <- x[3]*6+1
  t <- t+shift
  y <- (par1*t-2)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
design <- lhs::randomLHS(30,3)
candidate <- lhs::randomLHS(500,3)
candidate <- rbind(candidate,design)
## evaluate the response matrix on the design matrix
resp <- apply(design,1,forretal,timepoints)
x0 <- runif(3)
y0 <- forretal(x0,timepoints)
yobs <- y0+rnorm(200,0,sd(y0)/sqrt(50))
xhat <- ESL2D(design,resp,yobs,candidate,nstarts=1)
yhat <- forretal(xhat,timepoints)
## draw a figure to illustrate
plot(y0,ylim=c(min(y0,yhat),max(y0,yhat)))
lines(yhat,col="red")
```
knnsvdGP

**K-nearest neighbor SVD-Based GP model**

**Description**

Fits a K-nearest neighbour SVD-based GP model on a test set $X_0$, training set design and response matrix $\text{resp}$. The local neighbourhood sets consist of $nn$ points which are selected by the Euclidean distance with respect to the test points. See Zhang et al. (2018) for details. This function supports the parallelization via both the R packages "parallel" and the OpenMP library.

**Usage**

```r
knnsvdGP(design, resp, X0=design, nn=20, nsvd = nn, frac = .95,
          gstart = 0.0001, nstarts = 5, centralize=FALSE, maxit=100,
          errlog = "", nthread = 1, clutype="PSOCK")
```

**Arguments**

- `design`: An $N$ by $d$ matrix of $N$ training/design inputs.
- `resp`: An $L$ by $N$ response matrix of design, where $L$ is the length of the time series outputs, $N$ is the number of design points.
- `X0`: An $M$ by $d$ matrix of $M$ test inputs. The localized SVD-based GP models will be fitted on every point (row) of $X0$. The default value of $X0$ is design.
- `nn`: The number of neighborhood points selected by the Euclidean distance. the default value is 20.
- `nsvd`: The number of design points closest to the test points on whose response matrix to perform the initial singular value decomposition. The default value is $nn$.
- `frac`: The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.
- `gstart`: The starting number and upper bound for estimating the nugget parameter. If `gstart = sqrt(.Machine$double.eps)`, the nugget parameter will be fixed at `sqrt(.Machine$double.eps)`, since `sqrt(.Machine$double.eps)` is the lower bound of the nugget term. The default value is 0.0001.
- `nstarts`: The number of starting points used in the numerical maximization of the posterior density function. The larger `nstarts` will typically lead to more accurate prediction but longer computational time. The default value is 5.
- `centralize`: If `centralize=TRUE` the response matrix will be centralized (subtract the mean) before the start of the algorithm. The mean will be added to the predictive mean at the finish of the algorithm. The default value is `FALSE`.
- `maxit`: Maximum number of iterations in the numerical optimization algorithm for maximizing the posterior density function. The default value is 100.
- `errlog`: The path of a log file that records the errors occur in the process of fitting local SVD-based GP models. If an empty string is provided, no log file will be produced.
nthread  The number of threads (processes) used in parallel execution of this function. nthread=1 implies no parallelization. The default value is 1.

clutype  The type of parallelization utilized by this function. If clutype="OMP", it will use the OpenMP parallelization. Otherwise, it indicates the type of cluster in the R package "parallel". The default value is "PSOCK". Required only if nthread>1.

Value

pmean  An L by M matrix of posterior predicted mean for the response at the test set X0.

ps2  An L by M matrix of posterior predicted variance for the response at the test set X0.

flags  A vector of integers of length M which indicates the status for fitting the local SVD-based GP models for each of the M input points in the test set. The value 0 indicates successful fitting, the value 1 indicates an error in Cholesky decomposition of the correlation matrices, the value 2 indicates an error in SVD of the local response matrix, and the value 3 indicates an error in optimizing the nugget term.

Author(s)

Ru Zhang <heavenmarshal@gmail.com>,
C. Devon Lin <devon.lin@queensu.ca>,
Pritam Ranjan <pritamr@iimidr.ac.in>

References


See Also

lasvdGP, svdGP.

Examples

library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  par2 <- x[2]*16+4
  par3 <- x[3]*6+1
  t <- t+shift
  y <- (par1*t-par3)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
design <- lhs::randomLHS(100,3)
test <- lhs::randomLHS(20,3)

## evaluate the response matrix on the design matrix
resp <- apply(design,1,forretal,timepoints)

nn <- 15
gs <- sqrt(.Machine$double.eps)

## knnsvdGP with mutiple (5) start points for GP model estimation
## It use the R package "parallel" for parallelization
retknnmsp <- knnsvdGP(design,resp,test,nn,frac=.95,gstart=gs,
                          centralize=TRUE,nstarts=5,nthread=2,clutype="PSOCK")

## knnsvdGP with single start point for GP model estimation
## It does not use parallel computation
retknnss <- knnsvdGP(design,resp,test,nn,frac=.95,gstart=gs,
                          centralize=TRUE,nstarts=1,nthread=1)

---

lasvdGP

Local Approximate SVD-Based GP Models

Description

Fits a local approximate SVD-based GP model on a test set \(X_0\), training/design set \(\text{design}\) and response matrix \(\text{resp}\). The local neighborhood sets consist of \(n_0\) out of which \(n_0\) points are selected by the Euclidean distance with respect to the test points. The remaining \(n_0 - n_0\) neighborhood points are selected sequentially by a greedy algorithm proposed by Zhang et al. (2018). This function supports the parallelization via both the R packages "parallel" and the OpenMP library.

Usage

```r
lasvdGP(design, resp, X0=design, n0=10, nn=20, 
        nfea = min(1000,nrow(design)),
        nsvd = nn, nadd = 1, frac = .95, gstart = 0.0001,
        resvdThres = min(5, nn-n0), every = min(5,nn-n0),
        nstarts = 5,centralize=FALSE, maxit=100,
        errlog = "", nthread = 1, clutype="PSOCK")
```

Arguments

- `design`: An \(N \times d\) matrix of \(N\) training/design inputs.
- `resp`: An \(L \times N\) response matrix of design, where \(L\) is the length of the time series outputs, \(N\) is the number of design points.
- `X0`: An \(M \times d\) matrix of \(M\) test inputs. The localized SVD-based GP models will be fitted on every point (row) of \(X_0\). The default value of \(X_0\) is \(\text{design}\).
- `n0`: The number of points in the initial neighborhood set. The initial neighborhood set is selected by the Euclidean distance. The default value is 10.
The total number of neighborhood points. The \( n_n-n_0 \) points are selected sequentially by the proposed algorithm. The default value is 20.

**nfea**
The number of feasible points within which to select the neighborhood points. This function will only consider the \( n_{fea} \) design points closest to the test point in terms of Euclidean distance when selecting neighborhood points. The default value is the minimum of \( N \) and 1000.

**nsvd**
The number of design points closest to the test points on whose response matrix to perform the initial singular value decomposition. The default value is \( nn \).

**nadd**
The number of neighborhood points selected at one iteration. The default value is 1.

**frac**
The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.

**gstart**
The starting number and upper bound for estimating the nugget parameter. If \( g_{start} = \sqrt{\text{sqrt}(\text{Machine}$\text{double.eps})} \), the nugget parameter will be fixed at \( \sqrt{\text{sqrt}(\text{Machine}$\text{double.eps})} \), since \( \sqrt{\text{Machine}$\text{double.eps})} \) is the lower bound of the nugget term. The default value is 0.0001.

**resvdThres**
The threshold to re-perform SVD. After every \( resvdThres \) points have been included into the neighborhood set, the SVD of the response matrix will be re-performed and the SVD-based GP model will be refitted. The default value is the minimum of \( nn-n_0 \) and 5.

**every**
The threshold to refit GP models without re-perform SVD. After every every points have been included into the neighborhood set, the GP models will be refitted. But the SVD will not be re-performed. It is suggested \( every <= resvdThres \). The default value is the minimum of \( nn-n_0 \) and 5.

**nstarts**
The number of starting points used in the numerical maximization of the posterior density function. The larger \( nstarts \) will typically lead to more accurate prediction but longer computational time. The default value is 5.

**centralize**
If \( centralize=\text{TRUE} \) the response matrix will be centralized (subtract the mean) before the start of the algorithm. The mean will be added to the predictive mean at the finish of the algorithm. The default value is \( \text{FALSE} \).

**maxit**
Maximum number of iterations in the numerical optimization algorithm for maximizing the posterior density function. The default value is 100.

**errlog**
The path of a log file that records the errors occur in the process of fitting local SVD-based GP models. If an empty string is provided, no log file will be produced.

**nthread**
The number of threads (processes) used in parallel execution of this function. \( nthread=1 \) implies no parallelization. The default value is 1.

**clutype**
The type of parallelization utilized by this function. If \( clutype=\text{"OMP"} \), it will use the OpenMP parallelization. Otherwise, it indicates the type of cluster in the R package "parallel". The default value is "PSOCK". Required only if \( nthread>1 \).

**Value**

**pmean**
An \( L \) by \( M \) matrix of posterior predicted mean for the response at the test set \( \mathbf{X}_0 \).
ps2  An $L$ by $M$ matrix of posterior predicted variance for the response at the test set $X_0$.

flags  A vector of integers of length $M$ which indicates the status for fitting the local SVD-based GP models for each of the $M$ input points in the test set. The value 0 indicates successful fitting, the value 1 indicates an error in Cholesky decomposition of the correlation matrices, the value 2 indicates an error in SVD of the local response matrix, the value 3 indicates an error in optimizing the nugget term.

Author(s)

Ru Zhang <heavenmarshal@gmail.com>,
C. Devon Lin <devon.lin@queensu.ca>,
Pritam Ranjan <pritamr@iimidr.ac.in>

References


See Also

knnsvdGP, svdGP.

Examples

library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  par2 <- x[2]*16+4
  par3 <- x[3]*6+1
  t <- t+shift
  y <- (par1*t-2)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
design <- lhs::randomLHS(100,3)
test <- lhs::randomLHS(20,3)

## evaluate the response matrix on the design matrix
resp <- apply(design,1,forretal,timepoints)

n0 <- 14
nn <- 15
gs <- sqrt(.Machine$double.eps)

## lasvdGP with mutiple (5) start points for GP model estimation,
## It use the R package “parallel” for parallelization
reTLS <- lasvdGP(design,resp,test,n0,nn,frac=.95,gstart=gs,
## saEI

### Saddlepoint Approximate Expected Improvement Criterion for the Sequential Design for Inverse Problems

**Description**

This function performs the sequential design procedure for the inverse problem. It starts from an initial design set $x_i$ and selects the follow-up design points from the candidate set $c_{andei}$ as per the expected improvement (EI) criterion which is numerically approximated by the saddlepoint approximation technique in Huang and Oosterlee (2011). The surrogate is refitted using the augmented data via svdGP. After the selection of $nadd$ follow-up points, the solution of the inverse problem is estimated either by the ESL2D approach or by the SL2D approach. Details are provided in Chapter 4 of Zhang (2018).

**Usage**

```r
saEI(xi, yi, yobs, nadd, candei, candest, func, ..., 
    mtype=c("zmean","cmean","lmean"),
    estsol=c("ESL2D","SL2D"),
    frac=.95, nstarts=5, gstart=0.0001,
    nthread=1, clutype="PSOCK")
```

**Arguments**

- **xi**
  - An $N_0$ by $d$ matrix of $N_0$ initial design points.
- **yi**
  - An $L$ by $N_0$ response matrix of $x_i$, where $L$ is the length of the time series outputs, $N_0$ is the number of design points.
- **yobs**
  - A vector of length $L$ of the time-series valued field observations or the target response.
- **nadd**
  - The number of the follow-up design points selected by this function.
- **candei**
  - An $M_1$ by $d$ matrix of $M_1$ candidate points on which the follow-up design points are selected.
- **candest**
  - An $M_2$ by $d$ matrix of $M_2$ candidate points on which the (final) estimated solution to the inverse problem is extracted.
- **func**
  - An R function of the dynamic computer simulator. The first argument of func should be a vector of $d$-dimensional inputs. The simulator func should return a vector of length $L$ as the output.
- **...**
  - The remaining arguments of the simulator func.
The type of mean functions for the GP models. The choice "zmean" denotes zero-mean, "cmean" indicates constant-mean, "lmean" indicates linear-mean. The default choice is "zmean".

The method for estimating the final solution to the inverse problem after all follow-up design points are included, "ESL2D" denotes the ESL2D approach, "SL2D" denotes the SL2D approach. The default choice is "ESL2D".

The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.

The number of starting points used in the numerical maximization of the posterior density function. The larger nstarts will typically lead to more accurate prediction but longer computational time. The default value is 5.

The starting number and upper bound for estimating the nugget parameter. If gstart = sqrt(.Machine$double.eps), the nugget parameter will be fixed at sqrt(.Machine$double.eps), since sqrt(.Machine$double.eps) is the lower bound of the nugget term. The default value is 0.0001.

The number of threads (processes) used in parallel execution of this function. nthread=1 implies no parallelization. The default value is 1.

The type of cluster in the R package "parallel" to perform parallelization. The default value is "PSOCK". Required only if nthread>1.

The design set selected by the sequential design approach, which includes both the initial and the follow-up design points.

The response matrix collected on the design set xx.

The estimated solution to the inverse problem obtained on the candidate set candest from the final fitted surrogate.

A vector of length nadd, it collects the maximum value of the EI criterion in each iteration of the sequential design approach.

Ru Zhang <heavenmarshal@gmail.com>,
C. Devon Lin <devon.lin@queensu.ca>,
Pritam Ranjan <pritamr@iimidr.ac.in>


See Also

ESL2D, SL2D, svdGP.
Examples

```r
library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  par2 <- x[2]*16+4
  par3 <- x[3]*6+1
  t <- t+shift
  y <- (par1*t-2)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
xi <- lhs::randomLHS(30,3)
candi <- lhs::randomLHS(500,3)
candest <- lhs::randomLHS(500,3)
candest <- rbind(candest, xi)
## evaluate the response matrix on the design matrix
yi <- apply(xi,1,forretal,timepoints)
x0 <- runif(3)
y0 <- forretal(x0,timepoints)
yobs <- y0+rnorm(200,0,sd(y0)/sqrt(50))
ret <- saEI(xi,yi,yobs,1,candi,candest,forretal,timepoints,
nstarts=1, nthread=1)
yhat <- forretal(ret$xhat,timepoints)
```
Arguments

- **design**
  An $N$ by $d$ matrix of $N$ training/design inputs.

- **resp**
  An $L$ by $N$ response matrix of design, where $L$ is the length of the time series outputs, $N$ is the number of design points.

- **yobs**
  A vector of length $L$ of the time-series valued field observations or the target response.

- **candidate**
  An $M$ by $d$ matrix of $M$ candidate points on which the estimated solution to the inverse problem is extracted.

- **frac**
  The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.

- **nstarts**
  The number of starting points used in the numerical maximization of the posterior density function. The larger nstarts will typically lead to more accurate prediction but longer computational time. The default value is 5.

- **mtype**
  The type of mean functions for the GP models. The choice "zmean" denotes zero-mean, "cmean" indicates constant-mean, "lmean" indicates linear-mean. The default choice is "zmean".

- **gstart**
  The starting number and upper bound for estimating the nugget parameter. If gstart = sqrt(.Machine$double.eps), the nugget parameter will be fixed at sqrt(.Machine$double.eps), since sqrt(.Machine$double.eps) is the lower bound of the nugget term. The default value is 0.0001.

Value

- **xhat**
  The estimated solution to the inverse problem obtained from the candidate set candidate.

Author(s)

- Ru Zhang <heavenmarshal@gmail.com>
- C. Devon Lin <devon.lin@queensu.ca>
- Pritam Ranjan <pritamr@iimidr.ac.in>

References


See Also

ESL2D, saEI, svdGP.

Examples

```r
library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  ```
par2 <- x[2]*16+4
par3 <- x[3]*6+1
t <- t+shift
y <- (par1*t-2)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
design <- lhs::randomLHS(30,3)
candidate <- lhs::randomLHS(500,3)
candidate <- rbind(candidate,design)
## evaluate the response matrix on the design matrix
resp <- apply(design,1,forretal,timepoints)
x0 <- runif(3)
y0 <- forretal(x0,timepoints)
yobs <- y0+rnorm(200,0,sd(y0)/sqrt(50))
xhat <- SL2D(design,resp,yobs,candidate,nstarts=1)
yhat <- forretal(xhat,timepoints)
## draw a figure to illustrate
plot(y0,ylim=(min(y0,yhat),max(y0,yhat)))
lines(yhat,col="red")

svdGP

Full SVD-Based GP Models

Description
This function fits a full SVD-based GP model with test set X0, design set design and response matrix resp.

Usage

svdGP(design,resp,X0=design,nstarts=5,gstart=0.0001,
frac=.95,centralize=FALSE,nthread=1,clutype="PSOCK")

Arguments

design  An $N$ by $d$ matrix of $N$ training/design inputs.
resp    An $L$ by $N$ response matrix of design, where $L$ is the length of the time series outputs, $N$ is the number of design points.
X0      An $M$ by $d$ matrix of $M$ test inputs. The default value of X0 is design.
nstarts The number of starting points used in the numerical maximization of the posterior density function. The larger nstarts will typically lead to more accurate prediction but longer computational time. The default value is 5.
gstart The starting number and upper bound for estimating the nugget parameter. If gstart = sqrt(.Machine$double.eps), the nugget parameter will be fixed at sqrt(.Machine$double.eps), since sqrt(.Machine$double.eps) is the lower bound of the nugget term. The default value is 0.0001.
frac  The threshold in the cumulative percentage criterion to select the number of SVD bases. The default value is 0.95.

centralize  If centralize=TRUE the response matrix will be centralized (subtract the mean) before the start of the algorithm. The mean will be added to the predictive mean at the finish of the algorithm. The default value is FALSE.

nthread  The number of threads (processes) used in parallel execution of this function. nthread=1 implies no parallelization. The default value is 1.

clutype  The type of cluster in the R package "parallel" to perform parallelization. The default value is "PSOCK". Required only if nthread>1.

Value

pmean  An $L$ by $M$ matrix of posterior predicted mean for the response at the test set $X_0$.

ps2  An $L$ by $M$ matrix of posterior predicted variance for the response at the test set $X_0$.

Author(s)

Ru Zhang <heavenmarshal@gmail.com>,
C. Devon Lin <devon.lin@queensu.ca>,
Pritam Ranjan <pritamr@iimidr.ac.in>

See Also

knnsvdGP, lasvdGP.

Examples

library("lhs")
forretal <- function(x,t,shift=1)
{
  par1 <- x[1]*6+4
  par2 <- x[2]*16+4
  par3 <- x[3]*6+1
  t <- t+shift
  y <- (par1*t-2)^2*sin(par2*t-par3)
}
timepoints <- seq(0,1,len=200)
design <- lhs::randomLHS(50,3)
test <- lhs::randomLHS(50,3)

## evaluate the response matrix on the design matrix
resp <- apply(design,1,forretal,timepoints)

## fit full SVD-based GP model
ret <- svdGP(design,resp,test,frac=.95,nstarts=1,
            centralize=TRUE,nthread=2)
Index

* **GP model**
  - DynamicGP-package, 2
  - ESL2D, 3
  - knnsvdGP, 5
  - lasvdGP, 7
  - saEI, 10
  - SL2D, 12
  - svdGP, 14

* **Inverse problem**
  - ESL2D, 3
  - saEI, 10
  - SL2D, 12

* **SVD**
  - DynamicGP-package, 2
  - ESL2D, 3
  - knnsvdGP, 5
  - lasvdGP, 7
  - saEI, 10
  - SL2D, 12
  - svdGP, 14

* **neighborhood**
  - knnsvdGP, 5
  - lasvdGP, 7

* **package**
  - DynamicGP-package, 2

* **parallelization**
  - DynamicGP-package, 2

* **prediction**
  - knnsvdGP, 5
  - lasvdGP, 7
  - svdGP, 14

DynamicGP-package, 2

ESL2D, 3, 11, 13

knnsvdGP, 5, 9, 15

lasvdGP, 6, 7, 15

saEI, 4, 10, 13