

Package ‘mined’

December 16, 2018

Type Package

Title Minimum Energy Designs

Version 1.0-1

Date 2018-11-24

Author Dianpeng Wang and V. Roshan Joseph

Maintainer Dianpeng Wang <wdp@bit.edu.cn>

Description This is a method (MinED) for mining probability distributions using deterministic sampling which is proposed by Joseph, Wang, Gu, Lv, and Tuo (2018). The MinED samples can be used for approximating the target distribution. They can be generated from a density function that is known only up to a proportionality constant and thus, it might find applications in Bayesian computation. Moreover, the MinED samples are generated with much fewer evaluations of the density function compared to random sampling-based methods such as MCMC and therefore, this method will be especially useful when the unnormalized posterior is expensive or time consuming to evaluate.

License LGPL-2.1

Imports Rcpp (>= 0.12.17)

LinkingTo Rcpp, RcppEigen

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-12-16 14:40:02 UTC

R topics documented:

mined-package	2
Lattice	2
mined	3
SelectMinED	5

Index	7
--------------	----------

 mined-package

mined package

Description

Generate minimum energy design (MinED) samples from an unnormalized probability density function. The asymptotic distribution of MinED samples converges to the target distribution and therefore, MinED can be viewed as a deterministic sample from the target distribution. The details of MinED and the algorithm used for generating it can be found in Joseph, Dasgupta, Tuo, and Wu (2015) and Joseph, Wang, Gu, Lv, and Tuo (2018).

Details

Package: mined
 Type: Package
 Version: 1.0-1
 Date: 2018-10-06
 License: LGPL-2.1

Important functions in this package are: `mined` generates Minimum Energy Design samples from an unnormalized density function, `SelectMinED` selects Minimum Energy Design samples from candidate points, and `Lattice` generates good rank-1 lattice rules.

Author(s)

Dianpeng Wang and V. Roshan Joseph
 Maintainer: Dianpeng Wang <wdp@bit.edu.cn>

References

Joseph, V. R., Dasgupta, T., Tuo, R., and Wu, C. F. J. (2015). "Sequential Exploration of Complex Surfaces Using Minimum Energy Designs". *Technometrics*, 57, 64-74.

Joseph, V. R., Wang, D., Gu, L., Lv, S., and Tuo, R. (2018). "Deterministic Sampling of Expensive Posteriors Using Minimum Energy Designs", *Technometrics*, arXiv:1712.08929.

 Lattice

Good lattice points

Description

Generate good rank-1 lattice points with prime number of points by using the fast component-by-component construction algorithm of Nuyens and Cools (2006). Refer Nuyens (2007) for more details.

Usage

```
Lattice(n, p)
```

Arguments

n	The number of points, which should be a prime.
p	The number of dimensions.

Value

An n-by-p matrix containing the good lattice points.

Author(s)

Dianpeng Wang <wdp@bit.edu.cn> and V. Roshan Joseph <roshan@gatech.edu>

References

Nuyens, D. and Cools, R. (2006). "Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces.", *Mathematics of Computation*, 75, 903-920.

Nuyens, D. (2007). "Fast Construction of Good Lattice Rules.", *Ph.D Thesis*, Katholieke Universiteit Leuven, Leuven, Belgium.

Examples

```
library(mined)
res <- Lattice(101, 2)
plot(res[, 1], res[, 2], col='red', xlab='First dimension', ylab='Second dimension', pch=15)
```

mined

Minimum Energy Design

Description

Generate MinED samples from an unnormalized density function.

Usage

```
mined(initial, logf)
```

Arguments

initial	An n-by-p matrix containing the initial uniform samples from $[0,1]^p$.
logf	An R function to compute the logarithm of unnormalized density function. The input region should be scaled in $[0,1]^p$.

Details

This is the main function of the package, which is used for generating the MinED samples. The MinED sample can be viewed as a deterministic sample from the probability density specified in the mined function. Since only the unnormalized density is needed to generate the MinED samples, this method could be used in Bayesian computation to approximate the posterior. The method uses few evaluations of the unnormalized posterior compared to random sampling-based methods and therefore, it will be useful when the evaluations are expensive or time consuming.

There are many parameters that control the performance of the algorithm, which are fixed at some reasonable values as specified in Joseph et al. (2018). The only thing user need to choose is the region for scaling the variables in $[0,1]^p$. Ideally it should be the highercube containing the highest posterior density region with good coverage. However, the algorithm is robust to this choice to some extent as it can shrink or expand from the initial region. Therefore, it can be chosen based on user's guessed range of each variable.

Value

The value returned from the function is a list containing the following components:

points	A matrix containing n MinED samples.
logf	Log-unnormalized density function values of MinED samples.
cand	Full set of samples used in the algorithm.
candlf	Log-unnormalized density function values of the samples in cand.

Author(s)

Dianpeng Wang <wdp@bit.edu.cn> and V. Roshan Joseph <roshan@gatech.edu>

References

Joseph, V. R., Wang, D., Gu, L., Lv, S. and Tuo, R. (2018). "Deterministic Sampling of Expensive Posteriors Using Minimum Energy Designs.", *Technometrics*, arXiv:1712.08929.

Examples

```
require(mined)
p <- 2
n <- 109 # largest prime number less than 100+5p
initial <- Lattice(n, p)

# suppose x1 is in [-40,40] and x2 in [-25,10]
logf <- function(para)
{
  l1 <- -40
  u1 <- 40
  l2 <- -25
  u2 <- 10
  x1 <- l1 + (u1 - l1) * para[1]
  x2 <- l2 + (u2 - l2) * para[2]
  val <- -.5 * (x1 ^2 / 100 + (x2+ .03 * x1^2 -3)^2)
```

```

    return(val)
  }

  res <- mined::mined(initial, logf)
  dim(res$points)
  dim(res$cand)

  x1 <- seq(0, 1, length.out = 200)
  x2 <- seq(0, 1, length.out = 200)
  y <- matrix(0.0, 200, 200)
  for(i in 1:200)
  {
    for(j in 1:200)
    {
      y[i, j] = logf(c(x1[i], x2[j]))
    }
  }
  image(x1, x2, exp(y), col = cm.colors(5), xlab = expression(x[1]), ylab = expression(x[2]))
  points(res$cand[, 1], res$cand[, 2], pch = 11, col = rgb(red = 0, green = 0, blue = 1,
    alpha = 0.35), cex = .25)
  points(res$points[, 1], res$points[, 2], pch = 17, col = 'black', cex = .75)
  legend("bottom", c('Candidates points', 'MinED samples'), pch = c(11, 17),
    col = c(rgb(red = 0, green = 0, blue = 1, alpha = 0.35), 'black'),
    inset = .02, bg = 'transparent', bty = 'n')

```

 SelectMinED

Select Minimum Energy Design samples from a candidate set

Description

Select MinED samples from candidates by optimizing the generalized MinED criterion in Joseph et al. (2018).

Usage

```
SelectMinED(candidates, candlf, n, gamma=1, s=2)
```

Arguments

<code>candidates</code>	Candidate samples from the target distribution, which can be MC, QMC, or MCMC samples.
<code>candlf</code>	The log-unnormalized density function values corresponding to the candidates.
<code>n</code>	The required number of MinED samples.
<code>gamma</code>	The parameter in the annealed version of density function. Optional, default is “1”.
<code>s</code>	The parameter in generalized distance. Optional, default is “2”.

Details

This function select MinED samples from a given set of candidate samples. The function is used internally in the `mined` function repeatedly for K times, where K is the number of annealing steps in the algorithm. Refer to Joseph et al., (2018) for more details.

Value

The value returned from the function is a list containing the following components:

<code>points</code>	The MinED samples selected from the candidates.
<code>logf</code>	The log-unnormalized density function values of the points.

Author(s)

Dianpeng Wang <wdp@bit.edu.cn> and V. Roshan Joseph <roshan@gatech.edu>

References

Joseph, V. R., Wang, D., Gu, L., Lv, S. and Tuo, R. (2018). "Deterministic Sampling of Expensive Posteriors Using Minimum Energy Designs.", *Technometrics*, arXiv:1712.08929.

See Also

[mined](#)

Examples

```
cand <- matrix(runif(10000, min = -4, max = 4), ncol = 1)
candlf <- log(dnorm(cand))
res <- mined::SelectMinED(cand, as.vector(candlf), 150, 1.0, 2.0)
print(res)
par(mfrow=c(1,2))
hist(cand)
hist(res$points)
```

Index

*Topic **Bayesian Computation**

mined, [3](#)

*Topic **Experimental Design**

Lattice, [2](#)

mined, [3](#)

SelectMinED, [5](#)

*Topic **Quasi-Monte Carlo**

Lattice, [2](#)

mined, [3](#)

*Topic **package**

mined-package, [2](#)

Lattice, [2](#), [2](#)

mined, [2](#), [3](#), [6](#)

mined-package, [2](#)

SelectMinED, [2](#), [5](#)