Package 'Voss'

January 20, 2025

Type Package

Title Generic Voss Algorithm (Random Sequential Additions)

Version 0.1.5

Author Pavel V. Moskalev

Maintainer Pavel V. Moskalev <moskalefff@gmail.com>

Description

Generating realizations of a fractal Brownian function on uniform 1D & 2D grid with classic and generic versions of the Voss algorithm (random sequential additions).

License GPL-3

Encoding UTF-8

Depends fields

NeedsCompilation no

Repository CRAN

Date/Publication 2022-05-09 09:20:02 UTC

Contents

Voss-package *Generic Voss algorithm (random sequential additions)*

Description

Generating realizations of fractal Brownian functions on uniform 1D & 2D grid with classic and generic versions of the Voss algorithm (random sequential additions).

Details

Package:	Voss
Type:	Package
Version:	0.1.5
Date:	2022-05-09
License:	GPL-3

`voss1d()` and `voss2d()` functions generate realizations of fractal Brownian functions on uniform 1D & 2D grid with a classic version of the Voss algorithm (random sequential additions).
`voss1g()` and `voss2g()` functions generate realizations of fractal Brownian functions on uniform 1D & 2D grid with a generic version of the Voss algorithm (random sequential additions).

Author(s)

Pavel V. Moskalev

voss1d *Fractal Brownian function on 1D grid with a classic Voss algorithm*

Description

`voss1d()` function generates realizations of a fractal Brownian function on uniform 1D grid (FBF(x)) with a classic version of the Voss algorithm (random sequential additions).

Usage

```
voss1d(g=7, H=0.5, r=0.5, center=TRUE)
```

Arguments

`g`	a number of iteration.
`H`	a Hurst parameter: `(0<H)&(H<1)`.
`r`	a partition coefficient for iteration segments.
`center`	logical; if `center=TRUE` then the `y`-coordinates of prefractal points will be centered.

Details

The Voss algorithm on 1D grid is based on an iterative partitioning of the initial segment into smaller subsegments by linear interpolation of additional points.

At each iteration, all values of the fractal Brownian function get normal pseudorandom additions with zero mean and standard deviation, which depends on the iteration index `i`.

In the classical version of the Voss algorithm a standard deviation is exponentially distributed by iteration: `s[i] <- s0*r^(i*H)`, where the initial value `s0 <- H*log(1/r)`.

Value

A list of Cartesian coordinates of prefractal points.

Author(s)

Pavel V. Moskalev

References

Moskalev P.V. (2008) Visualization of wavelet spectra of fractal Brownian motion, Technical Physics, Vol.53, No.10, pp.1261-1266, doi:10.1134/S1063784208100022.

See Also

voss2d, voss1g

Examples

```
set.seed(20120522)
plot(voss1d(), type="l", xlab="x", ylab="y",
     main="FBF(x) with a parameter H=0.5")
abline(h=0, lty=2)
```

`voss1g` *Fractal Brownian function on 1D grid with a generic Voss algorithm*

Description

`voss1g()` function generates realizations of a fractal Brownian function on uniform 1D grid (FBF(x)) with a generic version of the Voss algorithm (random sequential additions).

Usage

```
voss1g(p=cbind(n=0.5^-seq(0,7)+1,
               s=dchisq(seq(0,7), df=2)),
       center=TRUE)
```

Arguments

p	a matrix of parameters: nrow(p) a number of iterations; p[,"n"] a number of partition points in the iteration process; p[,"s"] a standard deviation of normal pseudorandom additions;
center	logical; if center=TRUE then the y-coordinates of prefractal points will be centered.

Details

The Voss algorithm on 1D grid is based on an iterative partitioning of the initial segment into smaller subsegments by linear interpolation of additional points.

At each iteration, all values of the fractal Brownian function get normal pseudorandom additions with zero mean and standard deviation, which depends on the iteration index s[i].

By default, the iterative distribution of standard deviation in the generic version of the Voss algorithm is equal to the probability density of the chi-square distribution with 2 degrees of freedom: s[i] <- dchisq(i, df=2).

Value

A list of Cartesian coordinates of prefractal points.

Author(s)

Pavel V. Moskalev

References

Moskalev P.V. (2008) Visualization of wavelet spectra of fractal Brownian motion, Technical Physics, Vol.53, No.10, pp.1261-1266, doi:10.1134/S1063784208100022.

See Also

voss2g, voss1d

Examples

```
# Example 1: FBF(x) with a s[i]=dchisq(i,df=2)
set.seed(20120522)
plot(voss1g(), type="l", xlab="x", ylab="y",
     main="FBF(x) with a s[i]=dchisq(i,df=2)")
abline(h=0, lty=2)

# Example 2: FBF(x) with a s[i]=dlnorm(i,sdlog=1)
set.seed(20120522)
voss <- voss1g(p=cbind(n=0.5^-seq(0,7)+1,
                       s=dlnorm(seq(0,7), sdlog=1)))
plot(voss, type="l", xlab="x", ylab="y",
     main="FBF(x) with a s[i]=dlnorm(i,sdlog=1)")
```

```
abline(h=0, lty=2)

# Example 3: FBF(x,y) with a s[i]=df(i,df1=7,df2=7)
set.seed(20120522)
voss <- voss1g(p=cbind(n=0.5^-seq(0,7)+1,
                           s=df(seq(0,7), df1=7, df2=7)))
plot(voss, type="l", xlab="x", ylab="y",
     main="FBF(x) with a s[i]=df(i,df1=7,df2=7)")
abline(h=0, lty=2)
```

---

voss2d *Fractal Brownian function on 2D grid with a classic Voss algorithm*

Description

voss2d() function generates realizations of a fractal Brownian fFunction on uniform 2D grid (FBF(x,y)) with a classic version of the Voss algorithm (random sequential additions).

Usage

```
voss2d(g=7, H=0.5, r=0.5, center=TRUE)
```

Arguments

g	a number of iteration.
H	a Hurst parameter: (0<H)&(H<1).
r	a partition coefficient for iteration segments.
center	logical; if center=TRUE then the y-coordinates of prefractal points will be centered.

Details

The Voss algorithm on 2D grid is based on an iterative partitioning of the initial domain into smaller subdomains by bilinear interpolation of additional points.

At each iteration, all values of the fractal Brownian function get normal pseudorandom additions with zero mean and standard deviation, which depends on the iteration index i.

In the classical version of the Voss algorithm standard deviation is exponentially distributed by iteration: s[i] <- s0*r^(i*H), where the initial value s0 <- H*log(1/r).

Value

A list of Cartesian coordinates of prefractal points.

Author(s)

Pavel V. Moskalev

References

Shitov V.V. and Moskalev P.V. (2005) Modification of the Voss algorithm for simulation of the internal structure of a porous medium, Technical Physics, Vol.50, No.2, pp.141-145, doi:10.1134/1.1866426.

See Also

voss1d, voss2g

Examples

```
set.seed(20120522)
voss <- voss2d()
image(voss, xlab="x", ylab="y",
      main="FBF(x,y) with a parameter H=0.5")
contour(voss, levels=0, add=TRUE)
```

---

voss2g *Fractal Brownian function on 2D grid with a generic Voss algorithm*

Description

`voss2g()` function generates realizations of a fractal Brownian function on uniform 2D grid (FBF(x,y)) with a generic version of the Voss algorithm (random sequential additions).

Usage

```
voss2g(p=cbind(n=0.5^-seq(0,7)+1,
               s=dchisq(seq(0,7), df=2)),
       center=TRUE)
```

Arguments

`p`	a matrix of parameters: `nrow(p)` a number of iterations; `p[,"n"]` a number of partition points in the iteration process; `p[,"s"]` a standard deviation of normal pseudorandom additions;
`center`	logical; if `center=TRUE` then the y-coordinates of prefractal points will be centered.

Details

The Voss algorithm on 2D grid is based on an iterative partitioning of the initial domain into smaller subdomains by bilinear interpolation of additional points.

At each iteration, all values of the fractal Brownian function get normal pseudorandom additions with zero mean and standard deviation, which depends on the iteration index `s[i]`.

By default, the iterative distribution of standard deviation in the generic version of the Voss algorithm is equal to the probability density of the chi-square distribution with 2 degrees of freedom: `s[i] <- dchisq(i, df=2)`.

Value

A list of Cartesian coordinates of prefractal points.

Author(s)

Pavel V. Moskalev

References

Shitov V.V. and Moskalev P.V. (2005) Modification of the Voss algorithm for simulation of the internal structure of a porous medium, Technical Physics, Vol.50, No.2, pp.141-145, doi:10.1134/1.1866426.

See Also

voss1g, voss2d

Examples

```
# Example 1: FBF(x,y) with a s[i]=dchisq(i,df=2)
set.seed(20120522)
voss <- voss2g()
image(voss, xlab="x", ylab="y",
      main="FBF(x,y) with a s[i]=dchisq(i,df=2)")
contour(voss, levels=0, add=TRUE)

# Example 2: FBF(x,y) with a s[i]=dlnorm(i,sdlog=1)
set.seed(20120522)
voss <- voss2g(p=cbind(n=0.5^-seq(0,7)+1,
                       s=dlnorm(seq(0,7), sdlog=1)))
image(voss, xlab="x", ylab="y",
      main="FBF(x,y) with a s[i]=dlnorm(i,sdlog=1)")
contour(voss, levels=0, add=TRUE)

# Example 3: FBF(x,y) with a s[i]=df(i,df1=7,df2=7)
set.seed(20120522)
voss <- voss2g(p=cbind(n=0.5^-seq(0,7)+1,
                       s=df(seq(0,7), df1=7, df2=7)))
image(voss, xlab="x", ylab="y",
      main="FBF(x,y) with a s[i]=df(i,df1=5,df2=5)")
contour(voss, levels=0, add=TRUE)
```

Index