Package 'manifgen'

July 31, 2014

Type Package

Title Data Sets on Manifolds

Version 1.2-1

Date 2013-11-30

Author Kerstin Johnsson, Lund University

Maintainer Kerstin Johnsson <johnsson@maths.lth.se>

Description
Generates data points from (usually uniform) distributions on manifolds possibly with noise.

License GPL (>= 2)

LazyLoad yes

R topics documented:

manifgen-package *Data Sets on Manifolds*

Description

Generates data points from (usually uniform) distributions on manifolds possibly with noise.

Details

Package:	manifgen
Type:	Package
Version:	1.0
Date:	2012-07-04
License:	GPL (>= 2)
LazyLoad:	yes

Author(s)

Kerstin Johnsson, Lund University

Maintainer: Kerstin Johnsson <johnsson@maths.lth.se>

addNoise *Add Noise to Data Set*

Description

Embeds the data in `n` dimensions and adds normal isotropic noise to the data set. Hence `n` has to be at least equal to the dimension (the number of columns) of the data set, otherwise the function terminates with an error.

Usage

```
addNoise(data, n = ncol(data), sd)
```

Arguments

`data`	data set. Each row corresponds to a data point.
`n`	dimension of noise.
`sd`	standard deviation of noise. The covariance matrix of the noise is $sd^2 \cdot I$.

Author(s)

Kerstin Johnsson, Lund University

Examples

```
datap <- hcEdges(100, 1, 2)
datap <- addNoise(datap, 3, .05)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])
```

`cornerPlane` *Corner Plane*

Description

Generates a sample from a uniform distribution on a bent plane. Half of the plane is in the xz-plane and half of the plane is bent over the x-axis, so that the resulting surface has an edge along the x-axis.

Usage

```
cornerPlane(Ns, theta = pi/4)
```

Arguments

`Ns`	number of data points.
`theta`	angle at the x-axis.

Author(s)

Kerstin Johnsson, Lund University

Examples

```
datap <- cornerPlane(400)
par(mfrow = c(1, 2))
plot(datap[,1], datap[,2])
plot(datap[,1], datap[,3])
```

`cuthplane` *Piece of Noisy Hyperplane*

Description

Generates `Ns` data points within the unit ball from a hyperplane through the origin with noise added. `n` has to be at least `d`, otherwise the function terminates with an error.

Usage

```
cuthplane(Ns, d, n, sd)
```

Arguments

Ns	number of data points.
d	dimension of hyperplane.
n	dimension of noise.
sd	standard deviation of noise.

Details

The data set is generated the following way: First data points are sampled uniformly in a d-ball. After this, (n-d)-dimensional orthogonal noise with standard deviation sd in each direction is added. No noise is added in the directions parallel to the hyperplane since on an infinite plane adding isotropic noise to a uniform distribution does not change the distribution. Finally all data points within distance 1 from the origin are considered as candidates for the data set that will be returned, out of the candidates Ns data points are chosen randomly to be returned. If there are less than Ns candidates more candidates will be generated in the same way.

The data generated by this function can be used to evaluate how much local dimension estimators are affected by noise.

Warning

If sd is high, cuthplane will be slow and might not even be able to return a data set. If so, it will return NULL.

Author(s)

Kerstin Johnsson, Lund University

See Also

cuthsphere

Examples

```
datap <- cuthplane(100, 2, 3, 0.01)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])
```

---

cuthsphere *Piece of Noisy Hypersphere*

Description

Generates Ns data points cut out from a noisy hypersphere. n has to be at least d+1, otherwise the function terminates with an error.

Usage

```
cuthsphere(Ns, rat, d, n, sd)
```

Arguments

Ns	number of data points.
rat	ratio between cut-off radius and radius of sphere.
d	(intrinsic) dimension of hypersphere.
n	dimension of noise.
sd	standard deviation of noise.

Details

The returned data are within distance rat the point $1/\sqrt{d+1}(1...1)$ and are obtained from a unit distribution on the d-sphere overlaid with n-dimensional normal noise.

The data generated by this function can be used to evaluate the performance of local dimension estimators.

Warning

If sd is high, cuthsphere will be slow and might not even be able to return a data set. If so, it will return NULL.

Author(s)

Kerstin Johnsson, Lund University

See Also

cuthplane

Examples

```
datap <- cuthsphere(100, rat = .5, 1, 3, 0.01)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])

datap <- cuthsphere(100, rat = 2, 1, 3, 0.11)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])
```

---

hcube *Hypercube*

Description

Generates a sample from a uniform distribution on a hypercube, the faces of a hypercube or the "edges" of a hyper cube.

Usage

```
hcube(Ns, n, side = 1)
hcFaces(Ns, n)
hcEdges(Ns, d, n)
```

Arguments

Ns	number of data points.
d	dimension of edges.
n	dimension of the hypercube.
side	the length of the side of the hyper cube.

Details

The hypercube is $[0, 1]^n$. The edges of dimension d of the hypercube are the d-dimensional boundaries of the hypercube. The hypercube faces are the hyper cube edges of dimension n-1.

Author(s)

Kerstin Johnsson, Lund University.

Examples

```
datap <- hcEdges(200, 1, 3)
par(mfrow = c(1, 3))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])
plot(datap[, 2], datap[, 3])
```

---

lens *Length of vectors*

Description

Computes the Euclidean length of each row in vectors .

Usage

```
lens(vectors)
```

Arguments

vectors	a matrix where each row corresponds to a vector.

Author(s)

Kerstin Johnsson, Lund University.

Examples

```
datap <- hsphere(15, 10)
lens(datap)
```

M_hein *12-dimensional manifold from Hein and Audibert (2005)*

Description

Generates a 12-dimensional manifold with extrinsic dimension 72 (not uniformly sampled).

Usage

```
M_hein(Ns)
```

Arguments

Ns	number of data points.

Value

A 72-dimensional data set.

Author(s)

Kerstin Johnsson, Lund University.

References

Hein, M. and Audibert, J-Y. (2005) Intrinsic Dimensionality Estimation of Submanifolds in R^d. *Proceedings of ICML*, 289-296.

Examples

```
datap <- M_hein(800)
par(mfrow = c(1, 3))
plot(datap[,1], datap[,3])
plot(datap[,2], datap[,3])
plot(datap[,1], datap[,2])
```

M_rozza *Manifolds from Rozza et al. (2012)*

Description

Generates data sets from Rozza et al. (2012). M14 is an 18-dimensional manifold with intrinsic dimension 72. M14 is a 24-dimensional manifold with extrinsic dimension 96. Note that M14 and M15 are not uniformly sampled.

Usage

```
M14(Ns)
M15(Ns)
```

Arguments

Ns	number of data points.

Value

A 72-dimensional or 96-dimensional data set respectively.

Author(s)

Kerstin Johnsson, Lund University.

References

Rozza, A. et al. (2012) Novel high intrinsic dimensionality estimators. *Machine Learning*, 89:37-65.

Examples

```
datap <- M14(800)
par(mfrow = c(1, 3))
plot(datap[,1], datap[,3])
plot(datap[,2], datap[,3])
plot(datap[,1], datap[,2])
datap <- M15(800)
par(mfrow = c(1, 3))
plot(datap[,1], datap[,3])
plot(datap[,2], datap[,3])
plot(datap[,1], datap[,2])
```

---

oblong *Oblong Normal Distribution*

Description

Generates a sample from a certain anisotropic normal distribution centered around the origin.

Usage

```
oblong(Ns, n)
```

Arguments

Ns	number of data points.
n	dimension of the distribution (and the data points).

Details

In the first half of the dimensions (rounded down if n is odd) the standard deviation is 1 and in the rest the standard deviation is 0.25 .

Author(s)

Kerstin Johnsson, Lund University

Examples

```
datap <- oblong(100, 10)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 6])
```

Spherical *Isotropic distributions with or without noise*

Description

Generates a sample from isotropic distributions in d dimensions with n-dimensional noise added to it.

Usage

```
hball(Ns, d, n = d, sd = 0)
hsphere(Ns, d, n = d + 1, sd = 0)
normal(Ns, d, n = d, sd = 0)
```

Arguments

Ns	number of points.
d	intrinsic dimension of the support of the distribution (the manifold.)
n	dimension of noise.
sd	standard deviation of noise.

Details

hball draws a sample from a uniform distribution on a hyperball of radius 1. hsphere draws a sample from a uniform distribution on a hypersphere of radius 1. normal draws a sample from a isotropic normal distribution with identity covariance matrix.

Author(s)

Kerstin Johnsson, Lund University

Examples

```
datap <- hsphere(100, 1, 3, sd = .1)
par(mfrow = c(1, 2))
plot(datap[, 1], datap[, 2])
plot(datap[, 1], datap[, 3])
```

swissRoll *Swiss roll with or without 3-sphere inside*

Description

Generates a sample from a uniform distribution on a Swiss roll-surface, possibly together with a sample from a uniform distribution on a 3-sphere inside the Swiss roll.

Usage

```
swissRoll(Ns, a = 1, b = 2, nturn = 1.5, h = 4)
swissRoll3Sph(Ns, Nsph, a = 1, b = 2, nturn = 1.5, h = 4)
```

Arguments

Ns	number of data points on the Swiss roll.
Nsph	number of data points on the 3-sphere.
a	minimal radius of Swiss roll and radius of 3-sphere.
b	maximal radius of Swiss roll.
nturn	number of turns of the surface.
h	height of Swiss roll.

Value

swissRoll returns three-dimensional data points. swissRoll3Sph returns four-dimensional data points with the Swiss roll in the three first dimensions (columns). The Ns first data points lie on the Swiss roll and the Nsph last data points lie on the 3-sphere.

Author(s)

Kerstin Johnsson, Lund University.

Examples

```
datap <- swissRoll3Sph(300, 100)
par(mfrow = c(1, 3))
plot(datap[,1], datap[,2])
plot(datap[,1], datap[,3])
plot(datap[,1], datap[,4])
```

`twinPeaks` *Twin Peaks*

Description

Generates data points from a two- or higher-dimensional Twin Peaks manifold.

Usage

```
twinPeaks(Ns, h = 1)
nTwinPeaks(Ns, n, h = 1)
```

Arguments

`Ns`	number of data points.
`n`	dimension of the (hyper) plane from which the peaks stand out. For `twinPeaks` `n` is 2.
`h`	height of the peaks.

Details

The height of the points is computed as $\prod_1^n \sin(x_i)$, where $x_1, ..., x_n$ are the coordinates of the point in the (hyper) plane.

Value

A `n+1`-dimensional data set, where the last dimension represents the height of the points.

Author(s)

Kerstin Johnsson, Lund University.

Examples

```
datap <- twinPeaks(400)
par(mfrow = c(1, 3))
plot(datap[,1], datap[,3])
plot(datap[,2], datap[,3])
plot(datap[,1], datap[,2])
```

Index