Package 'SphericalK'

October 2, 2015

Type Package

Title Spherical K-Function

Version 1.2

Date 2015-10-01

Author Scott Robeson, Ao Li, Chunfeng Huang

Maintainer Ao Li<liao@umail.iu.edu>

Description Spherical K-function for point-pattern analysis on the sphere.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-10-02 22:01:17

R topics documented:

GUAN *GCOS Upper-Air Network (GUAN)*

Description

GUAN is a global-scale sampling networks created for many geophysical phenomena. In most cases, the goal of this system is to provide good converge of the entire planet in terms of (i) the desired density of points and (ii) the relative evenness of sample.

Usage

```
data(GUAN)
```

Format

A data frame with 172 observations on the following 8 variables.

`WMO` a numeric vector, Country(Area) index number

`Station.Name` a factor with levels

`FIPS` a factor with levels, Federal information processing standards

`Latitude` a numeric vector

`Longitude` a numeric vector

`Elevation` a numeric vector

`Begin` a numeric vector

`End` a numeric vector

Source

GCOS, 2007. GCOS Upper-Air Network (GUAN): Justification, Requirements,Sitting and Instrument Options. GCOS-112, WMO-TD 1379.

References

Robeson, S.M., Li, A., Huang, C., 2014. Point-pattern analysis on the sphere. Spatial Statistics. 10, 76-86.

See Also

`sphere_montekhat`, `sphere_khat`

Hex272 *Global Hexagonal Grids*

Description

The points are produced by a hexagonal tessellation at a resolution of 272 grid locations to evaluate the point-pattern properties.

Usage

```
data(Hex272)
```

Format

`number` a numeric vector

`longitude` a numeric vector

`latitude` a numeric vector

References

Robeson, S.M., Li, A., Huang, C., 2014. Point-pattern analysis on the sphere. Spatial Statistics. 10, 76-86.

Hex92 *Global Hexagonal Grids*

Description

The points are produced by a hexagonal tessellation at a resolution of 92 grid locations to evaluate the point-pattern properties.

Usage

```
data(Hex92)
```

Format

A data frame with 92 observations on the following 3 variables.

`number` a numeric vector

`longitude` a numeric vector

`latitude` a numeric vector

See Also

`sphere_montekhat`, `sphere_khat`

Examples

```
#Spherical K function (minus CSR) with 90% confidence intervals
#for point patterns associated with 92 points global hexagonal grid

data(Hex92)
lath92<-Hex92[,3]
lonh92<-Hex92[,2]
d<-seq(from=0,to=pi,by=0.35)
nd<-length(d)
d[nd]<-pi
khatsh92<-sphere_khat(lath92,lonh92,d)
Kcih92<-sphere_montekhat(92,50,d)
plot(d,khatsh92,type='n', ylim=c(-0.4,0.4),xlim=c(0,pi),xaxt = "n",
     ylab = expression(K - CSR),xlab = expression("Spherical Angle"))
axis(1, at = c(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi),
     labels = expression(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi))
polygon(c(d, rev(d)), c(Kcih92[3,], rev(Kcih92[47,])),col = "grey79", border = FALSE)
lines(d,khatsh92,col = 4, lwd=2)
lines(y=c(0,0),x=c(0,pi),type='l',lty=2,lwd=2)
```

sphere_grid *Latitude-Longitude Grids*

Description

Generates widely used Latitude-longitude grids for structuring global-scale data. A 10 * 10 equal-area grid is created in the example for future point-pattern analysis.

Usage

```
sphere_grid(degr)
```

Arguments

degr	A number that is applied to create a degr * degr grid, where 2.5 * 2.5 or 5 * 5 or 10 * 10 grids are commonly used.

Value

latitudes Latitudes of degr * degr grid points

longitudes Longitudes of degr * degr grid points

See Also

sphere_montekhat, sphere_khat

Examples

```
#Spherical K function (minus CSR) with 90% confidence intervals
#for point patterns associated with 25 * 25 latitude-longitude grid

spheregrid<-sphere_grid(25)
latm<-as.vector(spheregrid$latitudes)
lonm<-as.vector(spheregrid$longitudes)
d<-seq(from=0,to=pi,by=0.3)
nd<-length(d)
d[nd]<-pi
khatsg<-sphere_khat(latm,lonm,d)
Kcisg<-sphere_montekhat(98,50,d)
plot(d,khatsg,type='n', ylim=c(-0.4,0.4),xlim=c(0,pi),xaxt = "n",
ylab = expression(K - CSR),xlab = expression("Spherical Angle"))
axis(1, at = c(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi),
labels = expression(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi))
polygon(c(d, rev(d)), c(Kcisg[3,], rev(Kcisg[47,])),col = "grey79", border = FALSE)
lines(d,khatsg,col = 4, lwd=2)
lines(y=c(0,0),x=c(0,pi),type='l',lty=2,lwd=2)
```

sphere_khat *Calculate Spherical K-Function*

Description

Main function to obtain spherical K-function for point-pattern analysis on the sphere.

Usage

```
sphere_khat(latitudes, longitudes, dis)
```

Arguments

latitudes	Latitudes of observed points on the sphere in degrees
longitudes	Longitudes of observed points on the sphere in degrees
dis	Vector of values for the argument r (from 0 to pi), at which K(r) is evaluated. By default, dis = seq(from=0,to=pi,by=0.1).

Value

Khats Estimated values of K-function

References

Robeson, S.M., Li, A., Huang, C., 2014. Point-pattern analysis on the sphere. Spatial Statistics. 10, 76-86.

See Also

sphere_montekhat

Examples

```
lat<-sphere_random(100)$latitudes; lon<-sphere_random(100)$longitudes
d<-seq(from=0,to=pi,by=0.1)
sphere_khat(lat,lon,d)
```

sphere_montekhat *K-functions Under Complete Spatial Randomness (CSR) By Monte Carlo Tests*

Description

Monte Carlo confidence intervals of K-functions under CSR are provided for point-pattern analysis.

Usage

```
sphere_montekhat(n, nsim, dis)
```

Arguments

n	Number of observed points
nsim	Number of simulations for K-function
dis	Vector of values for the argument r (from 0 to pi), at which K(r) is evaluated. By default, dis = seq(from=0,to=pi,by=0.1).

Value

Kci Simulated K-functions under CSR

References

Robeson, S.M., Li, A., Huang, C., 2014. Point-pattern analysis on the sphere. Spatial Statistics. 10, 76-86.

See Also

sphere_khat, GUAN

Examples

```
#Spherical K function (minus CSR) with 95% confidence intervals
#for point patterns associated with 172 upper-air monitoring stations points

data(GUAN)
latg<-GUAN[,4]
long<-GUAN[,5]
d<-seq(from=0,to=pi,by=0.1)
nd<-length(d)
d[nd]<-pi
khatsg<-sphere_khat(latg,long,d)
Kcig<-sphere_montekhat(172,16,d)
plot(d,khatsg,type='n', ylim=c(-0.1,0.15),xlim=c(0,pi),xaxt = "n",
     ylab = expression(K - CSR),xlab = expression("Spherical Angle"))
axis(1, at = c(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi),
     labels = expression(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi))
```

```
polygon(c(d, rev(d)), c(Kcig[2,], rev(Kcig[16,])),col = "grey79", border = FALSE)
lines(d,khatsg,col = 4, lwd=2)
lines(y=c(0,0),x=c(0,pi),type='l',lty=2,lwd=2)
```

---

sphere_random *Generate Random Points On Sphere*

Description

Generate random points on the sphere.

Usage

```
sphere_random(n)
```

Arguments

n	Nnumber of data points on the sphere

Value

latitudes Latitudes of n random points on the sphere

longitudes Longitudes of n random points on the sphere

See Also

sphere_khat,sphere_montekhat

Examples

```
#Spherical K function (minus CSR) with 99% confidence intervals
#for point patterns associated with 80 random points

sphererandom<-sphere_random(80)
latrd<-sphererandom$latitudes;lonrd<-sphererandom$longitudes
d<-seq(from=0,to=pi,by=0.15)
nd<-length(d)
d[nd]<-pi
khatrd<-sphere_khat(latrd,lonrd,d)
Kcird<-sphere_montekhat(80,75,d)
plot(d,khatrd,type='n', ylim=c(-0.4,0.5),xlim=c(0,pi),xaxt = "n",
     ylab = expression(K - CSR),xlab = expression("Spherical Angle"))
axis(1, at = c(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi),
     labels = expression(0,pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi))
polygon(c(d, rev(d)), c(Kcird[1,], rev(Kcird[74,])),col = "grey79", border = FALSE)
lines(d,khatrd,col = 4, lwd=2)
lines(y=c(0,0),x=c(0,pi),type='l',lty=2,lwd=2)
```

Index