

Package ‘sphereplot’

October 14, 2022

Type Package

Title Spherical Plotting

Version 1.5.1

Date 2013-09-22

Author Aaron Robotham

Maintainer Aaron Robotham <aaron.robatham@uwa.edu.au>

Description Various functions for creating spherical coordinate system plots via extensions to rgl.

License GPL-2

Depends rgl

NeedsCompilation no

Repository CRAN

Date/Publication 2022-06-27 05:20:55 UTC

R topics documented:

sphereplot-package	2
car2sph	2
pointsphere	3
rgl.sphcirc	4
rgl.sphgrid	5
rgl.sphMW	6
rgl.sphpoints	7
rgl.sphsun	8
rgl.sphtext	9
sph2car	10
Index	11

sphereplot-package *Spherical plotting routines*

Description

Various low level and high level routines for generate spherical plots. Includes celestial sphere style 3D grid and routines for overlaying additional points and text. Requires rgl.

Details

Package: sphereplot
Type: Package
Version: 1.5
Date: 2013-09-22
License: GPL-2

Standard usage is to run rgl.sphgrid to make the 3D coordinate grid, then add points using rgl.sphpoints.

Author(s)

Aaron Robotham

Maintainer: Aaron Robotham <aaron.robotham@uwa.edu.au>

car2sph *Transforms 3D cartesian coordinates to spherical coordinates*

Description

Transforms 3D cartesian coordinates to spherical coordinates. The user can choose to return the spherical coordinates in degrees or radians.

Usage

```
car2sph(x, y, z, deg = TRUE)
```

Arguments

x x values, can also contain a matrix of x, y and z (in that order).
y y values.
z z values
deg Should degrees be returned (default) or radians.

Details

This is a low level function that is used for plot transformations.

Value

A data.frame is returned containing the columns long (longitude), lat (latitude) and radius.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [rgl.sphMW](#), [pointsphere](#), [sph2car](#)

Examples

```
print(car2sph(x=1,y=1,z=0,deg=TRUE))
```

pointsphere	<i>Random sphere pointing</i>
-------------	-------------------------------

Description

Randomly generates data points within a sphere that are uniformly distributed.

Usage

```
pointsphere(N = 100, longlim = c(0, 360), latlim = c(-90, 90), rlim = c(0, 1))
```

Arguments

N	Number of random points.
longlim	Limits of longitude in degrees.
latlim	Limits of latitude in degrees.
rlim	Limits of radius.

Details

This function randomly generates data points within a sphere that are uniformly distributed. 3D pointing is based in efficient inversion of random uniform distributions, rather than a Monte-Carlo approach.

Value

Returns a data.frame containing the longitude, latitude and radius of the random points generated.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [rgl.sphMW](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()
rgl.sphpoints(pointsphere(100,c(0,90),c(0,45),c(0.25,0.8)),deg=TRUE)
```

rgl.sphcirc

Great circle generator

Description

Function to generate a new great circle with arbitrary inclinations and radius.

Usage

```
rgl.sphcirc(CrossEq = 0, PeakDec = 0, radius = 1, deg = TRUE,
col = "black", ...)
```

Arguments

CrossEq	The right ascension where the new circle crosses the equator. This should be the crossing before the Declination values of the new great circle become positive.
PeakDec	The peak declination the new great circle will reach.
radius	The radius of the new great circle.
deg	Specifies if input is in degrees (default) or radians.
col	The colour of the new great circle line.
...	Other arguments carried to lines3d.

Details

See [rgl.sphsun](#) and [rgl.sphMW](#) for examples of rgl.sphcirc in use.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [rgl.sphMW](#), [pointsphere](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()
rgl.sphcirc(radius=0.5,col='blue')
```

rgl.sphgrid	<i>Create a spherical plotting grid</i>
-------------	---

Description

Creates a spherical plotting grid, within which further points can be added. Useful for astronomical plotting in particular, where this becomes the celestial sphere.

Usage

```
rgl.sphgrid(radius = 1, col.long='red', col.lat='blue', deggap = 15, longtype = "H",
add = FALSE, radaxis=TRUE, radlab='Radius')
```

Arguments

radius	The radial extent of the spherical grid.
deggap	The attempted separation between spherical grid lines in degrees.
col.long	The colour for longitude labels.
col.lat	The colour for latitude labels.
longtype	Specifies if longitudes should be labelled in hours (H- default) or degrees (D).
add	Should the grid be added to the current plot, or if FALSE a new rgl device is launched.
radaxis	Logical determining whether the sphere radius vector is drawn and labeled (default is TRUE). Pretty labeling is used to choose the location of ticks and labels.
radlab	If 'radaxis' is TRUE then the 'radlab' parameter determines the name of the label.

Details

This function should be called first, and can generally be used with only the declaration of radius to good effect.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

Examples

```
rgl.sphgrid()
```

```
rgl.sphMW
```

Plot Galactic plane + Galactic centre

Description

This function overplots the Galactic plane on the default Equatorial coordinates, and optionally will add the Galactic centre.

Usage

```
rgl.sphMW(radius = 1, col = "purple", type = "s", MWcenrad = 0.02, addMWplane = TRUE)
```

Arguments

radius	The radius at which to draw the Galactic plane and Galactic centre.
col	The colour of the Galactic plane line and the Galactic centre.
type	Rgl plot type for the Galactic centre, default is to draw it as a 3D sphere, i.e. type 's'.
MWcenrad	The radius of the Galactic centre if plotted as a sphere.
addMWplane	Should the Galactic plane be drawn.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [pointsphere](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()
rgl.sphMW()
```

rgl.sphpoints *Add points to spherical plots*

Description

This function allows the native plotting of spherical coordinates (in degrees or radians) and is expected to be used in conjunction with rgl.sphgrid, which produces the spherical grid.

Usage

```
rgl.sphpoints(long, lat, radius, deg = TRUE, col='black', ...)
```

Arguments

long	longitude values, can also contain a matrix of long, lat and radius (in that order).
lat	latitude values.
radius	radius values.
deg	Specifies if input is in degrees (default) or radians.
col	Specifies point colour.
...	Other arguments carried to points3d.

Details

This function uses sph2car in conjunction with points3d to plot points on a spherical coordinate system.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [rgl.sphMW](#), [pointsphere](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()  
rgl.sphpoints(40, 50, 0.5, deg=TRUE, col='red', cex=2)
```

`rgl.sphsun`*Plot ecliptic + Sun*

Description

This function overplots the ecliptic plane on the default Equatorial coordinates, and optionally will add the Sun either for a desired date, or for today.

Usage

```
rgl.sphsun(Ydate = c(3, 21), radius = 1, col = "yellow", type = "s", sunrad = 0.02,  
addeclip = TRUE, addsun=TRUE)
```

Arguments

<code>Ydate</code>	The date for the location of the Sun on the spherical grid. Vector in c(M,D) format. If set to 'get' then the function will return the Sun's location for today.
<code>radius</code>	The radius at which to draw the ecliptic plane and Sun.
<code>col</code>	The colour of the ecliptic line and for the Sun.
<code>type</code>	Rgl plot type for the Sun, default is to draw it as a 3D sphere, i.e. type 's'.
<code>sunrad</code>	The radius of the Sun if plotted as a sphere.
<code>addeclip</code>	Should the ecliptic plane be drawn.
<code>addsun</code>	Should the location of the Sun be plotted.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphMW](#), [pointsphere](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()  
rgl.sphsun()  
  
rgl.sphgrid()  
rgl.sphsun('get', radius=2, col='red')  
  
open3d()
```

rgl.sphtext *Add text to spherical plot*

Description

Adds generic text to a spherical coordinate plot.

Usage

```
rgl.sphtext(long, lat, radius, text, deg = TRUE, col='black', ...)
```

Arguments

long	longitude values, can also contain a matrix of long, lat and radius (in that order).
lat	latitude values.
radius	radius values.
text	text values to be plotted.
deg	Specifies if input is in degrees (default) or radians.
col	Specifies text colour.
...	Other arguments carried to points3d.

Details

This function uses sph2car in conjunction with text3d to plot text on a spherical coordinate system.

Value

No value, used for plotting side effect.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphsun](#), [rgl.sphMW](#), [pointsphere](#), [sph2car](#), [car2sph](#)

Examples

```
rgl.sphgrid()  
rgl.sphtext(40,50,0.5, 'HI!', deg=TRUE, col='red', cex=2)
```

`sph2car`*Transforms 3D spherical coordinates to cartesian coordinates*

Description

Transforms 3D spherical coordinates to cartesian coordinates. The user can choose to input the spherical coordinates in degrees or radians.

Usage

```
sph2car(long, lat, radius = 1, deg = TRUE)
```

Arguments

<code>long</code>	longitude values, can also contain a matrix of long, lat and radius (in that order).
<code>lat</code>	latitude values.
<code>radius</code>	radius values.
<code>deg</code>	Specifies if input is in degrees (default) or radians.

Details

This is a low level function that is used for plot transformations.

Value

A data.frame is returned containing the columns x, y and z.

Author(s)

Aaron Robotham

See Also

[rgl.sphgrid](#), [rgl.sphcirc](#), [rgl.sphpoints](#), [rgl.sphtext](#), [rgl.sphsun](#), [rgl.sphMW](#), [pointsphere](#), [car2sph](#)

Examples

```
print(sph2car(45,0,sqrt(2),deg=TRUE))
```

Index

- * **circle**
 - rgl.sphcirc, 4
 - * **galactic**
 - rgl.sphMW, 6
 - * **package**
 - sphereplot-package, 2
 - * **plot3d**
 - rgl.sphpoints, 7
 - rgl.sphtext, 9
 - * **plot**
 - rgl.sphgrid, 5
 - * **points3d**
 - rgl.sphpoints, 7
 - * **points**
 - rgl.sphpoints, 7
 - * **random**
 - pointsphere, 3
 - * **sun**
 - rgl.sphsun, 8
 - * **text3d**
 - rgl.sphtext, 9
 - * **text**
 - rgl.sphtext, 9
 - * **transform**
 - car2sph, 2
 - sph2car, 10
- car2sph, 2, 4–10
- pointsphere, 3, 3, 5–10
- rgl.sphcirc, 3, 4, 4, 6–10
- rgl.sphgrid, 3–5, 5, 6–10
- rgl.sphMW, 3–5, 6, 7–10
- rgl.sphpoints, 3–7, 7, 8–10
- rgl.sphsun, 3–7, 8, 9, 10
- rgl.sphtext, 3–8, 9, 10
- sph2car, 3–9, 10
- sphereplot (sphereplot-package), 2
- sphereplot-package, 2