Package: BBMM (via r-universe)

November 16, 2026

Type Package

Title Brownian Bridge Movement Model

Version 3.0.9001

Author R. M. Nielson [aut], H. Sawyer [aut], T. L. McDonald [aut], Ryan Nielson [cre]

Maintainer Ryan Nielson <rnielson@west-inc.com>

Description The model provides an empirical estimate of a movement path using discrete location data obtained at relatively short time intervals. Compatibility updates by Renova.

License GNU General Public License

NeedsCompilation no

Repository https://renova-r.r-universe.dev

Date/Publication 2026-11-16 17:54:31 UTC

RemoteUrl https://github.com/renova-r/BBMM

RemoteRef renova-3.0.9001

RemoteSha 4d76d885ae2bff7b4e975a78f7ebca88666495bd

Contents

BBMM-package 2
bbmm.contour 3
bbmm.summary 4
brownian.bridge 5
brownian.motion.variance 7
locations 8

Index **9**

BBMM-package *BBMM - Brownian bridge movement model*

Description

This package fits a Brownian bridge movement model to observed locations in space and time. This is a continuous-time stochastic model of movement in which the probability of being in an area during the time of observation is conditioned on starting and ending locations. A BBMM is typically fit to animal location data obtained by a Global Positioning System (GPS) or Very High Frequency (VHF) device. The model provides an empirical estimate of the movement path of an animal using discrete location data obtained at relatively short time intervals. This package allows the user to specify the grid cell size and extent over which the probability of use will be estimated.

Details

Package:	BBMM
Type:	Package
Version:	3.0
Date:	2013-06-03
License:	GNU General Public License

List of routines:

`brownian.bridge` Work-horse routine for estimating a Brownian bridge movement model.

`brownian.motion.variance` Estimates the Brownian motion variance.

`bbmm.summary` Summary method for bbmm objects.

`bbmm.contour` Contour plot for bbmm objects.

Author(s)

Ryan Nielson, Hall Sawyer, and Trent McDonald (WEST, Inc., www.west-inc.com)

Maintainer: Ryan Nielson <rnielson@west-inc.com>

References

Horne, J. S., E. O. Garton, S. M. Krone, and J. S. Lewis. 2007. Analyzing animal movements using Brownian bridges. Ecology 88:2354-2363.

Sawyer, H., M. J. Kauffman, R. M. Nielson, and J. S. Horne. 2009. Identifying and prioritizing ungulate migration routes for landscpe-level conservation. Ecological Applications 19:2016-2025.

bbmm.contour *Create density contours from a Brownian bridge movement model*

Description

Calculates and (optionally) plots density contours from a utilization distribution.

Usage

```
bbmm.contour(x, levels, locations = NULL, plot = TRUE)
```

Arguments

x	An object of class bbmm, which is a list with vectors x, y and probabilities.
levels	A numeric vector of desired contour levels (e.g., c(95, 99)).
locations	(optional) data frame or matrix of x and y coordinates for original location data used to estimate the Brownian bridge and utilization distribution surface.
plot	Logical. If TRUE, the countour plot is drawn.

Value

An list with four components.

Components of the returned object are as follows:

contour	Requested contour level(s).
z	values that separate the specified contour level(s).

Author(s)

Ryan Nielson, Hall Sawyer, and Trent McDonald (WEST, Inc., www.west-inc.com)

Maintainer: Ryan Nielson <rnielson@west-inc.com>

See Also

brownian.bridge

Examples

```
data(locations)
BBMM <- brownian.bridge(x=locations$x, y=locations$y,
            time.lag=locations$time.lag[-1], location.error=20,
            cell.size=50)
contours <- bbmm.contour(BBMM, levels=c(95, 99), locations=locations, plot=TRUE)

# Create a shapefile with contour lines
# Not run: library(maptools)
# Not run: library(raster)
```

```
# Not run: out <- data.frame(x=BBMM$x,y=BBMM$y,z=BBMM$probability)
# Make sure the data is properly projected
# Not run: out.raster <- rasterFromXYZ(out,crs=CRS("+proj=utm +zone=12 +datum=WGS84"),digits=2)
# Not run: raster.contour <- rasterToContour(out.raster,levels=contours$Z)
# Not run: raster.contour <- spChFIDs(raster.contour,paste(c(95, 99),"% Contour Line",sep=""))
# Not run: library(rgdal) # can't be loaded before CRS call in line above
# Not run: writeOGR(obj=raster.contour,dsn=".",layer="BBMM",driver="ESRI Shapefile")
```

---

bbmm.summary *Summary of a Brownian bridge movement model fit*

Description

A function used to produce a result summary from a Brownian bridge movement model.

Usage

```
bbmm.summary(x)
```

Arguments

x	An object of class 'bbmm', which is a result of `brownian.bridge(...)`.

Value

The estimated Brownian motion variance, along with the number and size of grid cells over which the Brownian bridge was estimated.

Author(s)

Ryan Nielson, Hall Sawyer, and Trent McDonald (WEST, Inc., www.west-inc.com)

Maintainer: Ryan Nielson <rnielson@west-inc.com>

See Also

brownian.bridge

Examples

```
data(locations)
BBMM <- brownian.bridge(x=locations$x, y=locations$y,
            time.lag=locations$time.lag[-1], location.error=20,
            cell.size=50)
bbmm.summary(BBMM)
```

`brownian.bridge` *Brownian bridge movement model*

Description

Estimate a Brownian bridge model of movement in which the probability of a mobile object being in an area is conditioned on starting and ending locations. The model provides an empirical estimate of a movement path using discrete location data obtained at relatively short time intervals. The Brownian bridge probability density connecting each pair of successive locations is an estimate of the relative time spent in an area during the time interval between those locations.

Usage

```
brownian.bridge(x, y, time.lag, location.error, area.grid = NULL,
    cell.size = NULL, time.step = 10, max.lag = NULL)
```

Arguments

`x`	Vector of x coordinates (in meters) of locations, ordered in time.
`y`	Vector of y coordinates (in meters) of locations, ordered in time.
`time.lag`	Vector of time differences (in minutes) between successive locations where `length(time.lag)` is the same as `length(x)-1`. `time.lag[1]` is the length of time between locations 1 and 2, and `time.lag[2]` is the difference between locations 2 and 3, and so on.
`location.error`	The standard deviation of normally distributed location errors (single value or vector of 1 value for each observation).
`area.grid`	(optional) Matrix or data frame of x and y coordinates of cell center ponts on a rectangular grid that defines the area in which to estimate probability of use. If missing, a grid is created by expanding the range of x and y by 1 standard deviation and using cell.size.
`cell.size`	(optional) Cell size for `area.grid`, if `area.grid` not provided. Must specify either `area.grid` or `cell.size`.
`time.step`	(optional) The Brownian bridge probability density function must be integrated to find the fraction of time spent in each region. While the probability density function cannot be integrated, it can be approximated by discretizing time into arbitrarily small intervals according to time.step. The default is 10 units (same as `time.lag`). A longer `time.step` speeds up estimation, but reduces precision.
`max.lag`	(optional) Maximum time lag (same units as `time.lag`) between successful locations to use in calculating the Brownian motion variance and probability of use. This can be important if some scheduled locations are missing, either via an unsuccessful fix attempt to locate the individual/object or because of a lack of effort. Including large time gaps in sequence of locations can artificially inflate/deflate the Brownian motion variance and potentially bias estimates of probability of use (to what degree is unknown). If two successive locations are

>max.lag, then a Brownian bridge is not estimated between those two locations. The default sets the max.lag equal to the maximum time difference in time.lag+1.

Details

This is the main routine for estimating a Brownian bridge. It calls brownian.motion.variance to estimate the Brownian motion variance via maximum likelihood and then calculates the probabilities of use across the area.grid. Larger data sets and larger grids require more computing time, which can be a few of hours on a 32-bit PC or just a fraction of an hour on a 64-bit PC running R x64.

Value

An object (list) of class("bbmm") with four components.

Components of the returned object are as follows:

Brownian motion variance	Estimated Brownian motion variance.
x	Vector of x coordinates for grid cells.
y	Vector of y coordinates for grid cells.
probability	Estimated proportion of time spent in each grid cell.

Author(s)

Ryan Nielson, Hall Sawyer, and Trent McDonald (WEST, Inc., www.west-inc.com)

Maintainer: Ryan Nielson <rnielson@west-inc.com>

References

Horne, J. S., E. O. Garton, S. M. Krone, and J. S. Lewis. 2007. Analyzing animal movements using Brownian bridges. Ecology 88:2354-2363.

Sawyer, H., M. J. Kauffman, R. M. Nielson, and J. S. Horne. 2009. Identifying and prioritizing ungulate migration routes for landscpe-level conservation. Ecological Applications 19:2016-2025.

See Also

brownian.motion.variance, bbmm.summary, bbmm.contour

Examples

```
data(locations)
BBMM <- brownian.bridge(x=locations$x, y=locations$y,
            time.lag=locations$time.lag[-1], location.error=20,
            cell.size=50)

# To export an Ascii grid with probabilities (UD).
# First delete any x and y coords that have probability of use < 0.00000001.
#  This saves some computing time and reduces the size of the output Ascii file.
# Not run: x <- BBMM$x[BBMM$probability >= 0.00000001]
```

```
# Not run: y <- BBMM$y[BBMM$probability >= 0.00000001]
# Not run: z <- BBMM$probability[BBMM$probability >= 0.00000001]
# Not run: tmp <- data.frame(x, y, z)
# Not run: library(maptools)
# Not run: m <- SpatialPixelsDataFrame(points = tmp[c("x", "y")], data=tmp)
# Not run: m <- as(m, "SpatialGridDataFrame")
# Not run: write.asciigrid(m, "BBMM.asc", attr=3)
```

`brownian.motion.variance` *Brownian motion variance*

Description

Estimates Brownian motion variance for a Brownian bridge movement model.

Usage

```
brownian.motion.variance(n.locs, time.lag, location.error, x, y, max.lag)
```

Arguments

`n.locs`	Number of observed locations in time and space.
`time.lag`	Vector of time differences (usually in minutes) between successive locations. The `length(time.lag)` = `length(x)` - 1.
`location.error`	The standard deviation of normally distributed location errors (single value or vector of 1 value for each observation).
`x`	Vector of x coordinates (meters) of locations, ordered in time.
`y`	Vector of y coordinates (meters) of locations, ordered in time.
`max.lag`	Maximum lag between successful fixes to consider.

Details

Users will generally not call this function directly. It is sourced by `brownian.bridge`.

Value

The estimate of Brownian motion variance, or diffusion coefficient related to the mobility of the object.

Author(s)

Ryan Nielson, Hall Sawyer, and Trent McDonald (WEST, Inc., www.west-inc.com)

Maintainer: Ryan Nielson <rnielson@west-inc.com>

References

Horne, J. S., E. O. Garton, S. M. Krone, and J. S. Lewis. 2007. Analyzing animal movements using Brownian bridges. Ecology 88:2354-2363.

Sawyer, H., M. J. Kauffman, R. M. Nielson, and J. S. Horne. 2009. Identifying and prioritizing ungulate migration routes for landscpe-level conservation. Ecological Applications 19:2016-2025.

See Also

brownian.bridge

Examples

```
data(locations)
BBMM <- brownian.bridge(x=locations$x, y=locations$y,
            time.lag=locations$time.lag[-1], location.error=20,
            cell.size=50)
```

---

locations *Example data for the Brownian bridge movement model function in package BBMM.*

Description

25 GPS locations from a female mule deer. Locations were recorded every 2 hours.

Usage

```
data(locations)
```

Format

A data frame with 25 observations on the following 3 variables.

`x` A numeric vector of x coordinates (UTM)

`y` A numeric vectorof y coordinates (UTM)

`time.lag` A numeric vector of time differences (min) between each successive observation.

Examples

```
data(locations)
BBMM <- brownian.bridge(x=locations$x, y=locations$y,
            time.lag=locations$time.lag[-1], location.error=20,
            cell.size=50)
bbmm.summary(BBMM)
```

Index

∗ **~Ascii**
    bbmm.contour, 3
∗ **~Brownian**
    brownian.bridge, 5
    brownian.motion.variance, 7
∗ **~GPS**
    brownian.bridge, 5
∗ **~UD**
    brownian.bridge, 5
∗ **~bridge**
    brownian.bridge, 5
∗ **~brownian.bridge**
    bbmm.summary, 4
∗ **~contour**
    bbmm.contour, 3
∗ **~motion**
    brownian.motion.variance, 7
∗ **~movement**
    brownian.bridge, 5
∗ **~variance**
    brownian.motion.variance, 7
∗ **datasets**
    locations, 8
∗ **package**
    BBMM-package, 2

BBMM *(*BBMM-package*)*, 2
BBMM-package, 2
bbmm.contour, *2*, 3, *6*
bbmm.summary, *2*, 4, *6*
bridge *(*brownian.bridge*)*, 5
Brownian *(*brownian.bridge*)*, 5
brownian.bridge, *2–4*, 5, *7, 8*
brownian.motion.variance, *2*, *6*, 7

locations, 8

model *(*brownian.bridge*)*, 5
movement *(*brownian.bridge*)*, 5