

Package ‘enerscape’

October 13, 2022

Type Package

Title Compute Energy Landscapes

Version 0.1.3

Author Emilio Berti

Maintainer Emilio Berti <emilio.berti@idiv.de>

Description Compute energy landscapes using a digital elevation model raster and body mass of animals. Vignette available at: <<https://emilio-berti.github.io/enerscape.html>>.

License GPL-3

Encoding UTF-8

LazyData true

Imports raster, gdistance, rgdal, rgeos, sp, Matrix

Suggests knitr, rmarkdown

RoxygenNote 7.1.1

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2021-10-01 10:10:02 UTC

R topics documented:

.calc_arc	2
.calc_arc_cond	2
.calc_cycling	3
.calc_cycling_cond	4
circuitscape_skeleton	4
enerscape	5
en_extrapolation	6
en_lcp	7
en_passage	8
en_path	9
omniscap_skeleton	10
pontzer	10

Index**12**

<code>.calc_arc</code>	<i>Compute energy costs</i>
------------------------	-----------------------------

Description

Internal function for enerscape - calculate work.

Usage

```
.calc_arc(slope, m, work_in_kcal = TRUE, j_to_kcal = 4184)
```

Arguments

<code>slope</code>	slope transition matrix.
<code>m</code>	species body mass (kg).
<code>work_in_kcal</code>	if work should be expressed in kilocalories.
<code>j_to_kcal</code>	joules to kilocalories conversion constant.

Details

Internal function of enerscape, don't call directly.

Value

A transition layer with values the energy cost of movement between cells (J or kcal).

<code>.calc_arc_cond</code>	<i>Compute conductance</i>
-----------------------------	----------------------------

Description

Internal function for enerscape - calculate conductance

Usage

```
.calc_arc_cond(slope, m, work_in_kcal = TRUE, j_to_kcal = 4184)
```

Arguments

<code>slope</code>	slope transition matrix.
<code>m</code>	species body mass (kg).
<code>work_in_kcal</code>	if work should be expressed in kilocalories.
<code>j_to_kcal</code>	joules to kilocalories conversion constant.

Details

Internal function of enerscape, don't call directly.

Value

A transition layer with values the conductance between cells, i.e. the distance that can be traveled per unit of energy (1 / J or 1 / kcal).

.calc_cycling	<i>Compute energy costs for a cyclist</i>
---------------	---

Description

Internal function for enerscape - calculate work.

Usage

```
.calc_cycling(height, slope, m, v, work_in_kcal = TRUE, j_to_kcal = 4184)
```

Arguments

height	height transition matrix.
slope	slope transition matrix.
m	species body mass (kg).
v	speed of cyclist.
work_in_kcal	if work should be expressed in kilocalories.
j_to_kcal	joules to kilocalories conversion constant.

Details

Internal function of enerscape, don't call directly. This assumes no wind, a bike of 7 kg, optimal pedal frequency, and constant mechanical efficiency of 25

Value

A transition layer with values the energy cost of movement between cells (J or kcal).

`.calc_cycling_cond` *Compute conductance for a cyclist*

Description

Internal function for enerscape - calculate work.

Usage

```
.calc_cycling_cond(height, slope, m, v, work_in_kcal = TRUE, j_to_kcal = 4184)
```

Arguments

<code>height</code>	height transition matrix.
<code>slope</code>	slope transition matrix.
<code>m</code>	species body mass (kg).
<code>v</code>	speed of cyclist.
<code>work_in_kcal</code>	if work should be expressed in kilocalories.
<code>j_to_kcal</code>	joules to kilocalories conversion constant.

Details

Internal function of enerscape, don't call directly. This assumes no wind, a bike of 7 kg, optimal pedal frequency, and constant mechanical efficiency of 25

Value

A transition layer with values the energy cost of movement between cells (J or kcal).

`circuitscape_skeleton` *Create the initialization file for the julia package Circuitscape*

Description

This creates the init file for the julia package Circuitscape: <https://juliapackages.com/p/circuitscape>.

Usage

```
circuitscape_skeleton(en = NULL, path = NULL, points = NULL)
```

Arguments

en	an enerscape object.
path	full path where to write the .ini file.
points	data.frame with origin and destination coordinates.

Value

Nothing, only write the circuitscape.ini file to disk.

enerscape	<i>Compute the energy landscape</i>
-----------	-------------------------------------

Description

This is the main function to compute energy landscapes from a digital elevation model and body mass of animals based on the model from Pontzer (2016). The core of the computations are done using the *gdistance* (Etten, 2017) package.

Usage

```
enerscape(dem, m, unit = "joule", neigh = 16, method = "ARC", v = NULL)
```

Arguments

dem	raster file of the digital elevation model, either a raster or a full path location of the file.
m	species body mass (kg).
unit	if joules ('joule') or kilocalories ('kcal').
neigh	number of neighbor cells that are connected together.
method	method to use to compute the energy costs. 'ARC' refers to the model from Pontzer (2016) and 'cycling' to the model for cyclist from di Prampero et al. (1979).
v	speed of cyclist (km / h), only for method = 'cycling'.

Details

From the digital elevation model, transition slopes, energy costs and conductances (1 / work) are computed based on the model described in Pontzer (2016).

Value

A list with elements a rasterStack of the digital elevation model, slope, energy landscape, and conductance and the conductance as a transitionLayer for path analysis.

References

- Etten, J. van. (2017). R Package gdistance: Distances and Routes on Geographical Grids. *Journal of Statistical Software*, 76(1), 1–21. doi: [10.18637/jss.v076.i13](https://doi.org/10.18637/jss.v076.i13).
- Pontzer, H. (2016). A unified theory for the energy cost of legged locomotion. *Biology Letters*, 12(2), 20150935. doi: [10.1098/rsbl.2015.0935](https://doi.org/10.1098/rsbl.2015.0935).
- di Prampero, P. E., Cortili, G., Mognoni, P., & Saibene, F. (1979). Equation of motion of a cyclist. *Journal of Applied Physiology*, 47(1), 201–206. doi: [10.1152/jappl.1979.47.1.201](https://doi.org/10.1152/jappl.1979.47.1.201)

Examples

```
library(raster)
data("volcano")
dem <- raster(volcano)
en <- enerscape(dem, 10, unit = "kcal", neigh = 16)
```

en_extrapolation	<i>Check if model extrapolates</i>
------------------	------------------------------------

Description

This check if computation of the energy landscape extrapolates from the test set of `enerscape::pontzer` (2016).

Usage

```
en_extrapolation(en, plot = TRUE)
```

Arguments

<code>en</code>	an <code>enerscape</code> object.
<code>plot</code>	plot areas where slope is extrapolated.

Details

Check if body mass or incline are outside the test range of the model. If slope extrapolations are detected and `plot = TRUE`, a plot of where extrapolations occur is displayed.

Value

A list with booleans if body size or inclines extrapolates and a `rasterLayer` for where incline extrapolates. The `rasterLayer` is returned only if extrapolations are present.

References

- `enerscape::pontzer`, H. (2016). A unified theory for the energy cost of legged locomotion. *Biology Letters*, 12(2), 20150935. doi: [10.1098/rsbl.2015.0935](https://doi.org/10.1098/rsbl.2015.0935).

Examples

```
library(raster)
library(enerscape)
data("volcano")
dem <- raster(volcano)
en <- enerscape(dem, 10, unit = "kcal", neigh = 16)
en_extrapolation(en, plot = TRUE)
```

en_lcp

Compute least-cost paths

Description

Calculate the least-cost path (lcp) between origin and destination

Usage

```
en_lcp(en, or, dest, simulate_random_points = FALSE, rep = 10, plot = TRUE)
```

Arguments

en	an enerscape object obtained with enerscape().
or	origin point.
dest	destination point.
simulate_random_points	if to simulate least-cost path among random points. default = FALSE.
rep	number of random origin and destination points if simulate_random_points = TRUE. default = 10.
plot	if to plot the output.

Details

If or and dest are not specified, the least-cost path is specified by setting simulate_random_points = TRUE and rep equal to the number of random paths to compute.

Value

A list with point locations, least-cost path as SpatialLines, energy costs and distances.

Examples

```
library(raster)
data("volcano")
dem <- raster(volcano)
en <- enerscape(dem, 10, unit = "kcal", neigh = 16)
p <- xyFromCell(dem, sample(ncell(dem), 2))
lcp <- en_lcp(en, or = p[1, ], dest = p[2, ])
```

en_passage

Compute probability of passage of random walks

Description

Calculate the net number of passages of random walks between origin and destination.

Usage

```
en_passage(
  en,
  or,
  dest,
  theta = 4,
  simulate_random_points = FALSE,
  rep = 10,
  plot = TRUE
)
```

Arguments

en	an enerscape object obtained with enerscape().
or	origin point.
dest	destination point.
theta	the degree from which the path randomly deviates from the least-cost path.
simulate_random_points	if to simulate least-cost path among random points. default = FALSE.
rep	number of random origin and destination points if simulate_random_points = TRUE. default = 10.
plot	if to plot the output.

Details

If or and dest are not specified, the least-cost path is specified by setting simulate_random_points = TRUE and rep equal to the number of random paths to compute.

Value

A list with point locations, rasterLayer of net passage of random walks, and rasterLayer of cumulative net passage if simulate_random_points = TRUE.

Examples

```
library(raster)
data("volcano")
dem <- raster(volcano)
en <- enerscape(dem, 10, unit = "kcal", neigh = 16)
p <- xyFromCell(dem, sample(ncell(dem), 2))
walk <- en_passage(en, or = p[1, ], dest = p[2, ], theta = 4)
```

en_path

Compute the energy costs for a chosen path

Description

This returns the distance and energy costs of traveling a chosen path. Optionally, the path can be selected by specifying the number of nodes and clicking on the plot.

Usage

```
en_path(en, p = NULL, draw = FALSE, n = NULL, plot = TRUE)
```

Arguments

en	an enerscape object.
p	path as SpatialLines.
draw	if TRUE the path will be chosen by drawing it on the map/
n	number of node points for the path.
plot	if TRUE plot the path

Value

A list with elements the path, its travel distance and energy costs.

References

- Etten, J. van. (2017). R Package gdistance: Distances and Routes on Geographical Grids. *Journal of Statistical Software*, 76(1), 1–21. doi: [10.18637/jss.v076.i13](https://doi.org/10.18637/jss.v076.i13).
- Pontzer, H. (2016). A unified theory for the energy cost of legged locomotion. *Biology Letters*, 12(2), 20150935. doi: [10.1098/rsbl.2015.0935](https://doi.org/10.1098/rsbl.2015.0935).
- di Prampero, P. E., Cortili, G., Mognoni, P., & Saibene, F. (1979). Equation of motion of a cyclist. *Journal of Applied Physiology*, 47(1), 201–206. doi: [10.1152/jappl.1979.47.1.201](https://doi.org/10.1152/jappl.1979.47.1.201)

Examples

```
library(raster)
library(enerscape)
data("volcano")
dem <- raster(volcano)
en <- enerscape(dem, 10, unit = "kcal", neigh = 16)
```

omniscAPE_skeleton	<i>Create the initialization file for the julia package OmniscAPE</i>
--------------------	---

Description

This creates the init file for the julia package OmniscAPE: <https://juliapackages.com/p/omniscAPE>.

Usage

```
omniscAPE_skeleton(en = NULL, path = NULL, radius = NULL, aggr_fact = 1)
```

Arguments

en	an enerscape object.
path	full path where to write the .ini file.
radius	radius in pixels of the moving window.
aggr_fact	the block size to compute the OmniscAPE.

Value

Nothing, only write the omniscAPE.ini file to disk.

pontzer	<i>Energy cost of transport from Pontzer (2016)</i>
---------	---

Description

Energy cost of transport from Pontzer (2016)

Usage

```
pontzer
```

Format

A data frame with 92 rows and 5 variables:

Species species name

Incline incline of movement

Mass species body mass

Cost.of.Transport cost of transport

Source original source of data

Source

doi: [10.1098/rsbl.2015.0935](https://doi.org/10.1098/rsbl.2015.0935)

References

#⁷ Pontzer, H. (2016). A unified theory for the energy cost of legged locomotion. *Biology Letters*, 12(2), 20150935.

Index

* datasets

- pontzer, [10](#)
- .calc_arc, [2](#)
- .calc_arc_cond, [2](#)
- .calc_cycling, [3](#)
- .calc_cycling_cond, [4](#)

- circuitscape_skeleton, [4](#)

- en_extrapolation, [6](#)
- en_lcp, [7](#)
- en_passage, [8](#)
- en_path, [9](#)
- enerscape, [5](#)

- omniscap_skeleton, [10](#)

- pontzer, [10](#)