Package ‘mousetrack’

February 20, 2015

Type Package

Title Mouse-Tracking Measures from Trajectory Data

Version 1.0.0

Date 2015-01-29

Author Moreno I. Coco and Nicholas D. Duran with contributions of Rick Dale, Denis O'Hora and Michael J. Spivey.

Maintainer Moreno I. Coco <moreno.cocoi@gmail.com>

Description Extract from two-dimensional x-y coordinates of an arm-reaching trajectory, several dependent measures such as area under the curve, latency to start the movement, x-flips, etc.; which characterize the action-dynamics of the response. Mainly developed to analyze data coming from mouse-tracking experiments.

License GPL (>= 2)

Depends R (>= 3.0.0), pracma

Collate 'angdiffvec.R' 'areaunder.R' 'getmouseDV.R' 'interpltraj.R' 'makerun.R' 'maxdev.R' 'pathoffset.R' 'sampen.R' 'sampenc.R' 'sampense.R' 'trim0.R' 'trim1.R' 'velaj.R' 'xflip.R'

NeedsCompilation no

Repository CRAN

Date/Publication 2015-01-29 11:43:37

R topics documented:

`mousetrack-package` *Mouse-tracking measures from trajectory data.*

Description

Extract from two-dimensional (x-y coordinates) of a trajectory several dependent measures, such as area under the curve, latency to start the movement, x-flips, etc. Mainly developed to analyze mouse-tracking data.

Details

Package:	mousertrack
Type:	Package
Version:	1.0
Date:	2015-01-28
License:	GPL >= 2

`getmouseDV`: Mother function to extract 40 different measures characterizing the action-dynamics response underlying the arm-reaching trajectory.

`interpltraj`: Interpolate one (or two-dimensional) trajectory into a fix number of bins.

Author(s)

Moreno I. Coco (moreno.cocoi@gmail.com) and Nicholas D. Duran (nicholas.duran@asu.edu)

References

Spivey, M., and Dale, R. (2006). Continuous dynamics in real-time cognition. Current Directions in Psychological Science, 15(5), 207-211.

Freeman, J. B., and Ambady, N. (2010). Mousetracker: Software for studying real-time mental processing using a computer mouse-tracking method. Behavior Research Methods, 42(1), 226-241.

Examples

```
data(mousemove) ## load data

unit = 25; dwellfin = escape = escapeinit = 100
velajbin = 6

x = mousemove$x; y = mousemove$y;
counterb = as.character( mousemove$counterb[1] )
refcounterb = "R"
t = mousemove$time

ans =  getmouseDV(x, y, t, unit, counterb,
```

```
        refcounterb, dwellfin, velajbin,
        escape, escapeinit)

str(ans)
```

---

getmouseDV *Get Mouse Dependent Variables*

Description

Mother function of the package, which computes from 2D (x, y) mouse-tracking trajectories, dependent measures characterizing the underlying action-dynamics patterns.

Usage

```
getmouseDV(x, y, t, unit, counterb, refcounterb,
dwellfin, velajbin, escape, escapeinit)
```

Arguments

`x`	x-coordinate point of the trajectory
`y`	y-coordinate point of the trajectory
`t`	a vector with time indexes
`unit`	the sampling unit, expressed in milliseconds
`counterb`	the position of the yes-no response button
`refcounterb`	the position of the yes-no response button, which we want to use as reference to transpose all trajectories on the same side of the display
`dwellfin`	Region (in pixels) around response button where measures of dwell are computed
`velajbin`	Number of timesteps used to average velocity
`escape`	Amount of pixels to escape when trimming the trajectory
`escapeinit`	Region around origin (in pixels) in which initial angle is measured

Value

It returns a list with 40 DVs:

`DVtotaltime`	The total duration of the trajectory
`DVlatency`	The latency of the start of the movement
`DVinmot`	The total time of motion
`DVdwell`	The dwell time to commit to a final response

`DVdist`	Euclidean distance of the trajectory
`DVdistinmot`	Distance traveled outside of the escape region
`DVvelmax`	The maximum velocity reached
`DVvelmaxstart`	The latency when maximum velocity was observed
`DVaccmax`	Maximum acceleration
`DVaccmaxstart`	The latency when maximum acceleration was observed
`arclengthtotal`	The length of the arc subtending the trajectory after motion was initiated
`maxpathoff`	Maximum offset of the trajectory
`DVxflplat`	Change in x-direction after escape region: latency
`DVxflpmot`	Change in x-direction after escape region: in motion
`DVafllat`	Change of angle-flipping in the dweel region: latency
`DVaflpmot`	Change of angle-flipping in the dweel region: in motion
`DVxflpdwl`	Change in x-direction in dwell region
`DVaflpdwl`	Change of angle-flipping in dwell region
`DVxe`	Entropy along x-axis (default 5 time-points)
`DVxse`	Entropy (standard error) along x-axis (default 5 time-points)
`DVye`	Entropy along y-axis (default 5 time-points)
`DVyse`	Entropy (standard error) along y-axis (default 5 time-points)
`DVae`	Entropy along angle-trajectory (default 5 time-points)
`DVase`	Entropy (standard error) along angle-trajectory (default 5 time-points)
`trajang`	Angles tangent to the trajectory
`DVAUC`	Area under the Curve
`DVmaxpull`	Maximum pull towards the incorrect response button
`DVmaxpullstart`	Latency of pull towards incorrect response button
`DVmaxang`	Maximum severity of angle towards incorrect response button while in motion
`DVmaxangstart`	Time-point where maximum severity of angle was observed.
`DVinitang`	The angle of trajectory after escaping region in absolute value
`DVmaxx`	Maximum x-value observed
`DVminx`	Minimum x-value observed
`DVmaxy`	Maximum y-value observed
`DVminy`	Minimum y-value observed
`OLnegmove`	Percentage of trajectory not moving towards incorrect response, i.e., negative y values
`OL1`	Binary value indicating whether motion time was longer than latency
`OL2`	Binary value indicating whether maximum velocity was inside the latency region
`OL3`	Binary value indicating whether maximum acceleration is inside the latency region
`OL4`	Binary value indicating whether trajectory travels in negative y-values after escaping the latency region

Note

A substantial amount of this code has been based on original MATLAB code written by Rick Dale (rdale@ucmerced.edu) and Michael J. Spivey (spivey@ucmerced.edu)

Author(s)

Moreno I. Coco (moreno.cocoi@gmail.com) and Nicholas D. Duran (nicholas.duran@asu.edu)

References

Spivey, M., and Dale, R. (2006). Continuous dynamics in real-time cognition. Current Directions in Psychological Science, 15(5), 207-211.

Freeman, J. B., and Ambady, N. (2010). Mousetracker: Software for studying real-time mental processing using a computer mouse-tracking method. Behavior Research Methods, 42(1), 226-241.

Examples

```
data(mousemove)
unit = 25; dwellfin = escape = escapeinit = 100
velajbin = 6

x = mousemove$x; y = mousemove$y;
counterb = as.character( mousemove$counterb[1] )
refcounterb = "R"
t = mousemove$time

ans =  getmouseDV(x, y, t, unit, counterb,
        refcounterb, dwellfin, velajbin,
        escape, escapeinit)

str(ans)
```

---

interpltraj *Get Mouse Dependent Variables*

Description

Interpolate a one-dimensional (angle), or two-dimension (x-y) trajectories to a user specified number of time bins.

Usage

```
interpltraj(x, y, singlepoint, tsmax)
```

Arguments

`x`	x-coordinate point of the trajectory
`y`	y-coordinate point of the trajectory
`singlepoint`	a logical flag to indicate whether interpolation is done on a single coordinate point (TRUE) or two points (FALSE)
`tsmax`	the new length of the interpolated trajectory

Value

It returns the interpolated trajectory, either one-dimensional (singlepoint == TRUE), or two-dimensional (singlepoint == FALSE)

Author(s)

Moreno I. Coco (moreno.cocoi@gmail.com)

References

Spivey, M., Grosjean, M. and Knoblich, G. (2005). Continuous attraction toward phonological competitors. Proceedings of the National Academy of Sciences of the United States of America, 102(29), 10393-10398.

Examples

```
data(mousemove)
x = mousemove$x; y = mousemove$y;
singlepoint = FALSE; tsmax = 101
ans = interpltraj(x, y, singlepoint, tsmax)

str(ans)
```

`mousemove` *Mouse movement trajectory*

Description

A single trial for a mouse-movement trajectory (x, y, time), and the type of counterbalancing used ("Left") in this specific case.

Usage

```
mousemov
```

Format

A dataframe with 4 columns (x, y, time, counterbl), and 61 rows.

References

Coco, M.I., and Duran, D.N. (under review). Action-dynamics as revision costs: the interaction of plausibility and congruency.

Index