Package 'DynaRankR'

January 20, 2025

Title Inferring Longitudinal Dominance Hierarchies

Version 1.1.0

Author Eli D. Strauss

Maintainer Eli D. Strauss <straussed@gmail.com>

Description Provides functions for inferring longitudinal dominance hierarchies, which describe dominance relationships and their dynamics in a single latent hierarchy over time. Strauss & Holekamp (in press).

License GPL-2

Encoding UTF-8

RoxygenNote 7.0.2

LazyData true

Imports dplyr, rlang

Suggests testthat

URL

NeedsCompilation no

Repository CRAN

Date/Publication 2020-02-13 15:10:02 UTC

Contents

C.crocuta.female   *Female spotted hyena dominance data*

Description

Data structures for inferring the ranks of adult female spotted hyenas (Crocuta crocuta) from a single social group in the Maasai Mara National Reserve in southern Kenya. Data are from the Talek clan collected between 1988 and 1995 by the Mara Hyena Project.

Usage

    C.crocuta.female

Format

List of 3 elements:

initial.ranks Starting order of females in 1988.

contestants Dataframe of 182 rows and 4 variables. There is one row per female per study year. **id** is the identity of the contestant. **period** is the study year. **convention1** is the mother of the contestant, because female spotted hyena hierarchies are structured by maternal rank inheritance. **convention2** is the intra-litter rank for contestants who were part of a twin litter.

interactions Dataframe of 2043 rows and 3 variables. Each row corresponds to the outcome of one aggressive interaction between adult females.

C.crocuta.male   *Male spotted hyena dominance data*

Description

Data structures for inferring the ranks of adult male spotted hyenas (Crocuta crocuta) from a single social group in the Maasai Mara National Reserve in southern Kenya. Data are from the Talek clan collected between 1988 and 1995 by the Mara Hyena Project.

Usage

    C.crocuta.male

Format

List of 3 elements:

initial.ranks Starting order of males in 1988.

contestants Dataframe of 143 rows and 3 variables. There is one row per male per study year. **id** is the identity of the contestant. **period** is the study year. **convention1** is the date that the male joined the clan, because male spotted hyena hierarchies are structured by a tenure-based convention.

interactions Dataframe of 474 rows and 3 variables. Each row corresponds to the outcome of one aggressive interaction between adult males.

---

dyadic_similarity *Compare two rank orderings*

Description

This function compares two rank orderings that have the same elements. For an order of length *n* there are *n choose 2* dyadic relationships implied by that order. For example, the order a, b, c implies that a > b, a > c, and b > c. The dyadic similarity between two orders is the proportion of implied dyadic relationships that are shared by the two orders.

Usage

```
dyadic_similarity(order1, order2)
```

Arguments

order1	The first rank ordering to be compared. Alternatively, this can be supplied as an interaction matrix with identities as the dimension names. All identities in order1 must be in order2.
order2	The second rank ordering to be compared. Alternatively, this can be supplied as an interaction matrix with identities as the dimension names. All identities in order2 must be in order1.

Value

The proportion of dyadic relationships that are shared by the two orders. This value is 1 if the orders are identical and 0 if the orders are exact opposites.

Examples

```
dyadic_similarity(letters[1:20], letters[1:20]) #identical orders
dyadic_similarity(letters[1:20], letters[20:1]) #opposite orders
dyadic_similarity(sample(letters[1:20]), sample(letters[1:20])) #random orders
```

---

edgelist_to_matrix   *Convert data from edgelist to interaction matrix*

Description

This function converts data in edgelist format to an interaction matrix.

Usage

```
edgelist_to_matrix(edgelist, identities)
```

Arguments

edgelist   A two column matrix or dataframe with the identities of winners in the first
                column and losers in the second column.

identities   A list of contestant identities. This list dictates the order in which contestants
                are arranged in the resulting matrix.

Value

Produces an interaction matrix with winners in the rows and losers in the columns. Contestants are
arranged according to the order specified by identities.

Examples

```
edges <- C.crocuta.female$interactions[C.crocuta.female$interactions$period == 1989,1:2]
ids <- C.crocuta.female$contestants[C.crocuta.female$contestants$period == 1989,'id']
edgelist_to_matrix(edgelist = edges, identities = ids)
```

---

get_dynamics   *Calculate hierarchy dynamics*

Description

Calculates hierarchy dynamics from an inferred longitudinal hierarchy, returning the provided longitudinal hierarchy with added columns. The function calculates hierarchy dynamics for each individual in each period (excluding the first period). Dynamics can be calculated in rank units or score units by specifying the **type** parameter. Rank dynamics are returned as the total dynamics (delta), active dynamics (delta.active), and passive dynamics (delta.passive). Score dynamics are calculated as delta. See Strauss & Holekamp (in revision) for more details.

Usage

```
get_dynamics(ranks, type = c("rank", "score"))
```

Arguments

ranks	A dataframe such as the output of the ranking functions provided in the DynaRankR package. Should include at least the following columns:

    period Study period.

    id Identity of contestant.

    rank Rank of id in period. Only required if calculating rank dynamics.

    score Score of id in period. Only required if calculating score dynamics.

type	A character string, either 'score' or 'rank'. Determines whether rank dynamics or score dynamics are calculated.

Value

Returns the supplied dataframe with new column(s) for hierarchy dynamics. New individuals receive NA for all dynamics because they can not have undergone any changes.

References

Strauss ED & Holekamp KE (in revision). Journal of Animal Ecology.

Examples

```
female.ranks <- informed_elo(contestants = C.crocuta.female$contestants, convention = 'mri',
initial.ranks = C.crocuta.female$initial.ranks,
interactions = C.crocuta.female$interactions)

female.ranks.score <- get_dynamics(ranks = female.ranks, type = 'score')
female.ranks.rank <- get_dynamics(ranks = female.ranks, type = 'rank')
```

---

get_edgelist	*Convert data from interaction matrix to edgelist*

Description

This function converts data in an interaction matrix to edgelist format.

Usage

```
get_edgelist(mat)
```

Arguments

mat	Interaction matrix containing outcomes of interactions. Dimension names are interpreted as individual identities.

Value

A two-column dataframe with winners in the first column and losers in the second column.

Examples

```
edges <- C.crocuta.female$interactions[C.crocuta.female$interactions$period == 1989,1:2]
ids <- C.crocuta.female$contestants[C.crocuta.female$contestants$period == 1989,'id']
mat <- edgelist_to_matrix(edgelist = edges, identities = ids)
get_edgelist(mat)
```

---

informed_ds   *David's Score method informed by prior information*

Description

Use David's Score method to infer a dominance hierarchy over multiple study periods. New contestants are added according to the convention specified by the user. Scores are calculated using Dij and are normalized. Full description of the addition of new individuals is described in Strauss & Holekamp (in revision). To run the original David's Score procedure, use convention flag 'none'.

Usage

```
informed_ds(contestants, convention, initial.ranks = NULL, interactions)
```

Arguments

contestants   A dataframe with the identities of the contestants for each study period along with the relevant data for adding them to the hierarchy. There should be one row per contestant per study period. Periods should appear in chronological order. The dataframe should contain the following columns:

period Study period.

id Identity of contestant.

convention1 The primary convention by which new individuals are added to the hierarchy. Interpretation of this column varies depending on the value of the **convention** argument. If **convention** = none, this column is optional.

convention2 Optional. The secondary data for resolving ties in convention1. Interpretation of this column varies depending on the value of the **convention** argument.

convention   A flag determining how new individuals are added to the hierarchy. The value of this flag influences how the convention1 and convention2 columns of the **contestants** argument are interpreted. Currently this function supports five options:

none The standard David's Score procedure (using Dij) is run. Individuals are not added according to prior information and scores are calculated independently for each period.

mri New contestants are added to the hierarchy according to maternal rank inheritance with youngest ascendancy. **convention1** should be a vector of mother identities for each contestant. **convention2** should be an optional vector of intra-litter ranks (lower numbers = higher rank) for resolving the order of contestants from the same mother joining the hierarchy in the same study period.

tenure New contestants are added to the hierarchy according their tenure in the group. **convention1** should be a vector of dates on which each contestant joined the group. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

age New contestants are added to the hierarchy according their age (older = higher rank). **convention1** should be a vector of birthdates or numerical age classes. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

phys_attr New contestants are added to the hierarchy according to some physical attribute (larger value = higher rank). **convention1** should be a vector of numerical attribute measurements. **convention2** should be an optional vector of numerical data for resolving ties in convention1. Higher values are considered higher rank.

initial.ranks   The initial ordering of individuals for the first study period. Required if using maternal rank inheritance as the convention. For other conventions, if initial.ranks is not specified, the order determined by convention1 is used to create the initial order.

interactions   A dataframe of interaction data with the following columns:

winner Identities of winners.

loser Identities of losers.

period Study period in which interactions occurred.

Value

Produces a dataframe with the following columns:

period Study period.

id Identity of contestant.

score David's Score of contestant.

rank Ordinal rank of contestant in study period. Lower numbers equal higher rank.

stan.rank Rank of contestant standardized for group size. Values range from 1 (highest rank) to -1 (lowest rank).

old.order Identity of contestants arranged in the previous order (the order they were in before updating the order based on observations from the current study period).

References

Strauss ED & Holekamp KE (in revision). Journal of Animal Ecology.

de Vries H, Stevens JMG, Vervaecke H (2006). Animal Behavior.

Examples

```
##Informed ds
female.ranks <- informed_ds(contestants = C.crocuta.female$contestants, convention = 'mri',
```

```
    initial.ranks = C.crocuta.female$initial.ranks,
    interactions = C.crocuta.female$interactions)

    ##Standard ds
    female.ranks <- informed_ds(contestants = C.crocuta.female$contestants, convention = 'none',
    interactions = C.crocuta.female$interactions)
```

---

informed_elo	*Elo-rating method informed by prior information*

Description

Use Elo-rating method to infer a dominance hierarchy over multiple study periods. New contestants
are added according to the convention specified by the user. Full description of the addition of
new individuals is described in Strauss & Holekamp (in revision). To run the original Elo-rating
procedure, use convention flag 'none'.

Usage

```
informed_elo(
  contestants,
  convention,
  K = 200,
  lambda = 100,
  initial.ranks = NULL,
  interactions
)
```

Arguments

contestants   A dataframe with the identities of the contestants for each study period along
                with the relevant data for adding them to the hierarchy. There should be one row
                per contestant per study period. Periods should appear in chronological order.
                The dataframe should contain the following columns:

                period Study period.

                id Identity of contestant.

                convention1 The primary convention by which new individuals are added to
                    the hierarchy. Interpretation of this column varies depending on the value
                    of the **convention** argument. If **convention** = none, this column is optional.

                convention2 Optional. The secondary data for resolving ties in convention1.
                    Interpretation of this column varies depending on the value of the **conven-
                    tion** argument.

convention   A flag determining how new individuals are added to the hierarchy. The value of
                this flag influences how the convention1 and convention2 columns of the **con-
                testants** argument are interpreted. Currently this function supports five options:

none The standard Elo-rating procedure is run. Individuals joining the hierarchy receive a score equal to the mean of other group members.

mri New contestants are added to the hierarchy according to maternal rank inheritance with youngest ascendancy. **convention1** should be a vector of mother identities for each contestant. **convention2** should be an optional vector of intra-litter ranks (lower numbers = higher rank) for resolving the order of contestants from the same mother joining the hierarchy in the same study period.

tenure New contestants are added to the hierarchy according their tenure in the group. **convention1** should be a vector of dates on which each contestant joined the group. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

age New contestants are added to the hierarchy according their age (older = higher rank). **convention1** should be a vector of birthdates or numerical age classes. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

phys_attr New contestants are added to the hierarchy according to some physical attribute (larger value = higher rank). **convention1** should be a vector of numerical attribute measurements. **convention2** should be an optional vector of numerical data for resolving ties in convention1. Higher values are considered higher rank.

K   Parameter influencing the magnitude of score changes after each outcome.

lambda   Parameter influencing the shape of the logistic function linking the difference in score between winner and loser to the expected probability of each contestant winning.

initial.ranks   The initial ordering of individuals for the first study period. Required if using maternal rank inheritance as the convention. For other conventions, if initial.ranks is not specified, the order determined by convention1 is used to create the initial order.

interactions   A dataframe of interaction data with the following columns:

winner Identities of winners.

loser Identities of losers.

period Study period in which interactions occurred.

Value

Produces a dataframe with the following columns:

period Study period.

id Identity of contestant.

score Elo-rating score of contestant.

rank Ordinal rank of contestant in study period. Lower numbers equal higher rank.

stan.rank Rank of contestant standardized for group size. Values range from 1 (highest rank) to -1 (lowest rank).

old.order Identity of contestants arranged in the previous order (the order they were in before updating the order based on observations from current study period).

References

Strauss ED & Holekamp KE (in revision). Journal of Animal Ecology.

Albers PCH & de Vries H (2000). Animal Behavior.

Examples

```
##Informed elo
female.ranks <- informed_elo(contestants = C.crocuta.female$contestants, convention = 'mri',
initial.ranks = C.crocuta.female$initial.ranks,
interactions = C.crocuta.female$interactions)

##Standard elo
female.ranks <- informed_elo(contestants = C.crocuta.female$contestants, convention = 'none',
interactions = C.crocuta.female$interactions)
```

---

informed_matreorder   *Infer longitudinal hierarchy using informed matrix reordering*

Description

Implements the Informed MatReorder method described in Strauss & Holekamp (in revision) to infer a dominance hierarchy over multiple study periods. For each study period, ranks are inferred as modifications of the ranks from the previous study period. First, new contestants are added according to the convention specified by the user, and emigrated/dead contestants are removed. Then, matrix reordering is used to change the position of contestants for whom data from the current study period are inconsistent with this ordering. The optimal order is selected as the order that is most consistent with the data from the current period and is minimally changed from the previous study period.

Usage

```
informed_matreorder(
  contestants,
  convention,
  n = 50,
  shuffles = 10,
  require.corroboration = FALSE,
  initial.ranks = NULL,
  interactions
)
```

Arguments

contestants	A dataframe with the identities of the contestants for each study period along with the relevant data for adding them to the hierarchy. There should be one row per contestant per study period. Periods should appear in chronological order. The dataframe should contain the following columns:

  period Study period.

  id Identity of contestant.

  convention1 The primary convention by which new individuals are added to the hierarchy. Interpretation of this column varies depending on the value of the **convention** argument.

  convention2 Optional. The secondary data for resolving ties in convention1. Interpretation of this column varies depending on the value of the **convention** argument.

convention	A flag determining how new individuals are added to the hierarchy. The value of this flag influences how the convention1 and convention2 columns of the contestants argument are interpreted. Currently this function supports four options:

  mri New contestants are added to the hierarchy according to maternal rank inheritance with youngest ascendancy. **convention1** should be a vector of mother identities for each contestant. **convention2** should be an optional vector of intra-litter ranks (lower numbers = higher rank) for resolving the order of contestants from the same mother joining the hierarchy in the same study period.

  tenure New contestants are added to the hierarchy according their tenure in the group. **convention1** should be a vector of dates on which each contestant joined the group. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

  age New contestants are added to the hierarchy according their age (older = higher rank). **convention1** should be a vector of birthdates or numerical age classes. **convention2** should be an optional vector of numerical data for resolving ties in convention1 (e.g., body size). Higher values are considered higher rank.

  phys_attr New contestants are added to the hierarchy according to some physical attribute (larger value = higher rank). **convention1** should be a vector of numerical attribute measurements. **convention2** should be an optional vector of numerical data for resolving ties in convention1. Higher values are considered higher rank.

n	Number of separate reordering attempts per study period. Recommended 100.
shuffles	Number of reshuffling steps per reordering attempt. Recommended at least 10.
require.corroboration	

  A logical indicating whether to require corroborating evidence from multiple study periods before changing a contestant's position in the order. Useful for reducing the sensitivity of the method to aberrant observations that don't reflect a lasting change in the true latent hierarchy. If true, evidence indicating a change in status must be corroborated by an additional observation in the following periods. See Strauss & Holekamp (in revision) for full details.

initial.ranks   The initial ordering of individuals for the first study period. Required if us-
                  ing maternal rank inheritance as the convention. For other conventions, if ini-
                  tial.ranks is not specified, the order determined by convention1 is used to create
                  the initial order.

interactions   A dataframe of interaction data with the following columns:

                  winner Identities of winners.

                  loser Identities of losers.

                  period Study period in which interactions occurred.

Value

Produces a dataframe with the following columns:

period Study period.

id Identity of contestant.

rank Ordinal rank of contestant in study period. Lower numbers equal higher rank.

stan.rank Rank of contestant standardized for group size. Values range from 1 (highest rank) to -1
(lowest rank).

old.order Identity of contestants arranged in the previous order (the order they were in before
updating the order based on observations from current study period).

References

Strauss ED & Holekamp KE (in revision). Journal of Animal Ecology.

Examples

```
conts <- C.crocuta.female$contestants[C.crocuta.female$contestants$period <= 1990,]
female.ranks <- informed_matreorder(contestants = conts,
convention = 'mri', n =1, shuffles = 10, require.corroboration = TRUE,
initial.ranks = C.crocuta.female$initial.ranks,
interactions = C.crocuta.female$interactions)
```

---

plot_ranks   *Plot individual ranks or scores*

Description

Plot ranks of individuals in a single social group over multiple study periods.

Usage

```
plot_ranks(ranks, type = c("rank", "stan.rank", "score"))
```

Arguments

ranks	A dataframe of ranks. There should be one row per contestant per study period. Must include at least the following columns:

period Study periods. They should appear in chronological order.

id The identity of each contestant. Each contestant should appear once per study period.

rank The rank of each contestant in each study period. This can be absolute rank or standardized rank. Only required if **type** is 'rank'.

score The score of each contestant. Only required if **type** is 'score'.

type	A character string, either 'score', 'rank', or 'stan.rank'. Determines whether scores, ranks, or standardized ranks are plotted.

Examples

```
female.ranks <- informed_elo(contestants = C.crocuta.female$contestants, convention = 'mri',
initial.ranks = C.crocuta.female$initial.ranks,
interactions = C.crocuta.female$interactions)

plot_ranks(female.ranks, type = 'rank')
plot_ranks(female.ranks, type = 'score')
plot_ranks(female.ranks, type = 'stan.rank')
```

Index