Package 'usedist'

May 12, 2017

Type Package

Title Distance Matrix Utilities

Version 0.1.0

Author Kyle Bittinger

Maintainer Kyle Bittinger <kylebittinger@gmail.com>

Description Functions to re-arrange, extract, and work with distances.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Suggests testthat

NeedsCompilation no

Repository CRAN

Date/Publication 2017-05-12 06:10:36 UTC

R topics documented:

`dist_between_centroids` *Compute the distance between group centroids*

Description

Compute the distance between group centroids

Usage

```
dist_between_centroids(d, idx1, idx2)
```

Arguments

`d`	A distance matrix object of class ‘"dist"‘.
`idx1`	A vector of items in group 1
`idx2`	A vector of items in group 2

Value

The distance between group centroids (see details).

It is possible to infer the distance between group centroids directly from the distances between items in each group. The ‘adonis‘ test in the ecology package ‘vegan‘ takes advantage of this approach to carry out an ANOVA-like test on distances.

The approach rests on the assumption that the items in the distance matrix occupy some high-dimensional Euclidean space. However, we do not have to actually create the space to find the distance between centroids. Using the assumption that such a space exists, we can use an algebraic formula to find the centroid distance.

The formulas for this were presented by Apostol and Mnatsakanian in 2003, though we need to re-arrange equation 28 to get the value we want:

$$|c_1 - c_2| = \sqrt{\frac{1}{n_1 n_2}\sum_{(1,2)} - \frac{1}{n_1^2}\sum_{(1)} - \frac{1}{n_2^2}\sum_{(2)}},$$

where n_1 is the number of samples in group 1, $\sum_{(1)}$ is the sum of squared distances between items in group 1, and $\sum_{(1,2)}$ is the sum of squared distances between items in group 1 and those in group 2.

References

Apostol, T.M. and Mnatsakanian, M.A. Sums of squares of distances in m-space. Math. Assoc. Am. Monthly 110, 516 (2003).

dist_get — *Retrieve distances from a ‘"dist"‘ object. Check if square Check if numeric*

Description

Retrieve distances from a ‘"dist"‘ object. Check if square Check if numeric

Usage

```
dist_get(d, idx1, idx2)
```

Arguments

d	A distance matrix object of class ‘"dist"‘.
idx1, idx2	Indicies specifying the distances to extract.

Value

A vector of distances.

Examples

```
m4 <- matrix(1:16, nrow=4, dimnames=list(LETTERS[1:4]))
dm4 <- dist(m4)
dist_get(dm4, "A", "C")
dist_get(dm4, "A", c("A", "B", "C", "D"))
dist_get(dm4, c("A", "B", "C"), c("B", "D", "B"))
```

dist_groups — *Create a data frame of distances between groups of items.*

Description

Create a data frame of distances between groups of items.

Usage

```
dist_groups(d, g)
```

Arguments

d	A distance matrix object of class ‘"dist"‘.
g	A factor representing the groups of objects in ‘d‘.

Value

A data frame with 6 columns. "Item1" and "Item2" identify the items compared, using the label if available. Likewise, "Group1" and "Group2" identify the groups of the items. "Label" is a factor giving a convenient label for the type of comparison. Finally, "Distance" contains the distance of interest.

Examples

```
m4 <- matrix(1:16, nrow=4, dimnames=list(LETTERS[1:4]))
dm4 <- dist(m4)
g4 <- rep(c("Control", "Treatment"), each=2)
dist_groups(dm4, g4)
```

`dist_make` *Make a distance matrix using a custom distance function*

Description

Make a distance matrix using a custom distance function

Usage

```
dist_make(x, distance_fcn, method = NULL)
```

Arguments

`x`	A matrix of observations, one per row
`distance_fcn`	A function of two arguments, used to compute the distance between two rows of the data matrix.
`method`	Name for the distance method. If provided, will be stored in the ‘"method"‘ attribute of the result.

Value

A ‘"dist"‘ object containing the distances between rows of the data matrix.

Examples

```
x <- matrix(sin(1:30), nrow=5)
rownames(x) <- LETTERS[1:5]
manhattan_distance <- function (v1, v2) sum(abs(v1 - v2))
dist_make(x, manhattan_distance, "Manhattan (custom)")
```

dist_setNames *Set the names/labels of a ‘"dist"‘ object.*

Description

Set the names/labels of a ‘"dist"‘ object.

Usage

```
dist_setNames(d, nm)
```

Arguments

d	A distance matrix object of class ‘"dist"‘.
nm	New labels for the rows/columns.

Examples

```
m4 <- matrix(1:16, nrow=4, dimnames=list(LETTERS[1:4]))
dm4 <- dist(m4)
dist_setNames(dm4, LETTERS[9:12])
```

dist_subset *Extract parts of a ‘"dist"‘ object.*

Description

This function also works to re-arrange the elements of a distance matrix, if the indicies are provided in the desired order.

Usage

```
dist_subset(d, idx)
```

Arguments

d	A distance matrix object of class ‘"dist"‘.
idx	Indices specifying the subset of distances to extract.

Value

A distance matrix.

Examples

```
m4 <- matrix(1:16, nrow=4, dimnames=list(LETTERS[1:4]))
dm4 <- dist(m4)
dist_subset(dm4, c("A", "B", "C"))
dist_subset(dm4, c("D", "C", "B", "A"))
```

`dist_to_centroids` *Compute distances from each item to group centroids*

Description

Compute distances from each item to group centroids

Usage

```
dist_to_centroids(d, g)
```

Arguments

d	A distance matrix object of class ‘"dist"‘.
g	A factor representing the groups of items in ‘d‘.

Value

A data frame with distances to the group centroids (see details).

This function computes the distance from each item to the centroid positions of groups defined in the argument ‘g‘. This is accomplished without determining the centroid positions directly; see the documentation for `dist_between_centroids` for details on this procedure.

The result is a data frame with three columns:

Item A character vector of item labels from the dist object, or an integer vector of item locations if labels are not present.

CentroidGroup The group for which the centroid distance is given. The column type should match that of the argument g (the `unique` function is used to generate this column).

CentroidDistance Inferred distance from the item to the centroid position of the indicated group.

`usedist` *usedist: a package for working with distance matrices in R*

Description

In usedist, we provide a number of functions to help with distance matrix objects, such as those produced by the ‘dist‘ function. Some functions are geared towards making or altering distance matrix objects. Others relate to groups of items in the distance matrix. They provide access to within- or between-group distances, or use these distances to infer the distance to group centroids.

Index