

# Package ‘NeuroSpatCo’

May 22, 2011

**Type** Package

**Title** Búsqueda de patrones organizacionales en corteza cerebral humana

**Version** 2.0

**Date** 2011-04-16

**Author** Camilo A Herrera R <hr.camilo@hrglobalideas.com>

**Maintainer** Camilo A Herrera R <hr.camilo@hrglobalideas.com>

**Depends** R (>= 2.10), tcltk, cluster, sm, stats, pgirmess, spatstat, fitdistrplus, rgl, MASS

**Suggests** Rcmdr

**Description** Herramienta para analisis de datos contenidos en imagenes de neurohistoquímica.

**License** GPL (>=2)

**LazyLoad** yes

## R topics documented:

NeuroSpatCo-package . . . . .	2
assignCluster . . . . .	2
Cuadrant . . . . .	3
distClust . . . . .	4
grafKS . . . . .	5
imprDist . . . . .	6
KernWallis . . . . .	7
KMeans . . . . .	8
read.folder . . . . .	9
reconsBeta . . . . .	9
trimExt . . . . .	10

<b>Index</b>	<b>12</b>
--------------	-----------

---

NeuroSpatCo-package

*NeuroSpatCo*

---

### Description

Este paquete es una implementación realizada para la tesis de pregrado de Camilo A Herrera, Estudiante de Estadística de la Universidad del Valle, Colombia. El paquete sirve para la búsqueda de patrones organizacionales en la corteza prefrontal de los humanos tanto en plano Horizontal somas como en plano Vertical minicolumnas.

### Details

Package:	NeuroSpatCo
Type:	Package
Version:	2.0
Date:	2011-05-22
License:	GPL (>=2)
LazyLoad:	yes

### Author(s)

Camilo Alberto Herrera Rozo, Contact: Camilo Alberto Herrera Rozo < hr.camilo@hrglobalideas.com>

---

assignCluster

*Append a Cluster Membership Variable to a Dataframe*

---

### Description

Correctly creates a cluster membership variable that can be attached to a dataframe when only a subset of the observations in that dataframe were used to create the clustering solution. NAs are assigned to the observations of the original dataframe not used in creating the clustering solution.

### Usage

```
assignCluster(clusterData, origData, clusterVec)
```

**Arguments**

<code>clusterData</code>	The data matrix used in the clustering solution. The data matrix may have only a subset of the observations contained in the original dataframe.
<code>origData</code>	The original dataframe from which the data used in the clustering solution were taken.
<code>clusterVec</code>	An integer variable containing the cluster membership assignments for the observations used in creating the clustering solution. This vector can be created using <code>cutree</code> for clustering solutions generated by <code>hclust</code> or the <code>cluster</code> component of a list object created by <code>kmeans</code> or <code>KMeans</code> .

**Value**

A factor (with integer labels) that indicate the cluster assignment for each observation, with an NA value given to observations not used in the clustering solution.

**Author(s)**

Dan Putler

**See Also**

[hclust](#), [cutree](#), [kmeans](#), [KMeans](#)

---

Cuadrant

*Cuadrant*

---

**Description**

Esta funcion realiza el test de aleatoriedad por el algoritmo de recorte de cuadrantes

**Usage**

```
Cuadrant(Carpeta, graf = FALSE)
```

**Arguments**

<code>Carpeta</code>	parametro de entrada, en el cual se coloca el objeto a ser explorado, el objeto generalmente es un conjunto de datos por ejemplo: un objeto resultante de la funcion <code>trimExt</code> .
<code>graf</code>	se indica si se desea obtener las graficas de los cuandrantos (TRUE), Default=(FALSE)

**Author(s)**

Camilo Alberto Herrera Roza < [hr.camilo@hrglobalideas.com](mailto:hr.camilo@hrglobalideas.com) >

## References

Diggle, P.J. Statistical analysis of spatial point patterns. Academic Press, 2003. Stoyan, D. and Stoyan, H. (1994) Fractals, random shapes and point fields: methods of geometrical statistics. John Wiley and Sons.

---

distClust

*distClust*


---

## Description

Esta funcion lee objetos tipo NeuroSpatCo y genera agrupaciones (clusters) de puntos en sentido vertical. finalmente entrega como salida una lista con todos las distancias entre clusters hallados y objetos con el condensado de cada carpeta en evaluacion.

## Usage

```
distClust(Carpeta, Grafica)
```

## Arguments

Carpeta	Objeto de la Clase (NeuroSpatCo) que contiene la informacion de las carpetas a ser evaluadas.
Grafica	Grafica = 0: solo entrega los objetos de vuelta. Grafica = 1: entrega los objetos y se pueden visualizar los clusters encontrados Grafica = 2: entrega los objetos y ademas guarda la imagen de los clusters (en ejes de componentes principales) en las carpetas donde se encuentran los archivos iniciales (.txt)

## Author(s)

Camilo Alberto Herrera Roza < hr.camilo@hrglobalideas.com >

## Examples

```
## se genera un objetos de la clase (NeuroSpatCo)
Carpetas<-read.folder()
##se hace uso de la funcion distClust
distancias<-distClust(Carpetas,0)
```

---

grafKS	<i>Graficas test de aleatoriedad completa grafKS(Carpeta, Kest = TRUE, Lest = FALSE, density = FALSE)</i>
--------	---

---

## Description

Esta funcion realiza las graficas del test de aleatoriedad completa las cuales son la guia para el analisis de patrones de organizacion en casos en los que se evalua la existencia de patrones puntuales.

## Usage

```
grafKS(Carpeta, Kest = TRUE, Lest = FALSE, density = FALSE)
```

## Arguments

Carpeta	Entrada de datos, variables de la clase NeuroSpatCo.
Kest	si Kest=TRUE se grafican las funciones de aleatoriedad completa K de ripley.
Lest	si Lest=TRUE se grafican las funciones de aleatoriedad completa L que es una transformacion de las funcion K de ripley con el fin de evaluar la aleatoriedad teniendo en cuenta variaciones estocasticas mediante el metodo de montecarlo.
density	si density=TRUE se grafican imagenes guia que seran de gran ayuda al momento de evaluar si realmente los patrones encontrados pertenecen a patrones de agrupacion o si es debido a algun otro factor externo.

## Author(s)

Camilo Alberto Herrera Rozo < hr.camilo@hrglobalideas.com >

## References

NACressie, N.A.C. Statistics for spatial data. John Wiley and Sons, 1991. Diggle, P.J. Statistical analysis of spatial point patterns. Academic Press, 1983. NANARipley, B.D. Statistical inference for spatial processes. Cambridge University Press, 1988. Stoyan, D, Kendall, W.S. and Mecke, J. (1995) Stochastic geometry and its applications. 2nd edition. Springer Verlag. Stoyan, D. and Stoyan, H. (1994) Fractals, random shapes and point fields: methods of geometrical statistics. John Wiley and Sons.

## Examples

```
## se genera un objetos de la clase (NeuroSpatCo)
#Carpetas<-read.folder()
##se realiza una de las graficas (K_ripley) para cada patron puntual
#grafKS(Carpetas, Kest = TRUE, Lest = FALSE, density = FALSE)
##el resultado queda consignado en la carpeta de lectura de datos inicial.
```

---

imprDist

*imprDist*


---

### Description

se realiza la busqueda grafica de la distribución de una serie de variables contenidas en un listado list(), generalmente filtrados previamente.

### Usage

```
imprDist(recortados)
```

### Arguments

recortados      parametro de entrada de datos contenidos en un listado list(), generalmente filtrados previamente.

### Value

imagen            contiene la grafica resultante de la aplicacion del grafico de cullen and frey  
dist#            se guardaran cada uno de los parametros de cada una de las distribuciones de los datos evaluados

### Author(s)

Camilo Alberto Herrera Rozo < hr.camilo@hrglobalideas.com>

### References

Cullen AC and Frey HC (1999) Probabilistic techniques in exposure assessment. Plenum Press, USA, pp. 81-159. Evans M, Hastings N and Peacock B (2000) Statistical distributions. John Wiley and Sons Inc. Fisher RA (1930) The moments of the distribution for normal samples of measures of departures from normality. Proc. R. Soc. London, Series A 130, 16-28.

### Examples

```
## se genera un objetos de la clase (NeuroSpatCo)
#Carpetas<-read.folder()
## se hace uso de la funcion distClust
#distancias<-distClust(Carpetas,0)
##se recortan los datos segun criterio del experto
#recortados<-trimExt(distancias,0.10)
##se realiza el proceso de descripcion de los datos
#distribuciones<-imprDist(recortados)
```

KernWallis

*KernWallis***Description**

Realiza el test de Kruskal-Wallis y su prueba post, ademas genera graficas de Kernels si Kern=TRUE.

**Usage**

```
KernWallis(recortados, Kern = FALSE)
```

**Arguments**

recortados	Variable que contiene los datos a comparar preferiblemente despues de haber sido recortados.
Kern	default=FALSE, si Kern=TRUE se realizara la grafica de Kernels conjunta de todas las distribuciones de datos a evaluar.

**Value**

[1]	contiene el resultado del test de Kruskal-Wallis
[2]	contiene el resultado del test post Kruskal-wallis que sera la guia para saber entre que grupos existen diferencias

**Author(s)**

Camilo Alberto Herrera Roza < hr.camilo@hrglobalideas.com>

**References**

Bowman, A.W. and Azzalini, A. (1997). Applied Smoothing Techniques for Data Analysis: the Kernel Approach with S-Plus Illustrations. Oxford University Press, Oxford. NASiegel and Castellan (1988) Non parametric statistics for the behavioural sciences. MacGraw Hill Int., New York. pp 213-214

**Examples**

```
# se genera un objetos de la clase (NeuroSpatCo)
Carpetas<-read.folder()
# se hace uso de la funcion distClust
distancias<-distClust(Carpetas,0)
#se recortan los datos segun criterio del experto
recortados<-trimExt(distancias,0.10)
#No se realizan las graficas de kernels y el test se guarda en una variable Resultados
Resultados<-KernWallis(recortados, Kern = FALSE)
```

---

**KMeans***K-Means Clustering Using Multiple Random Seeds*

---

**Description**

Finds a number of k-means clustering solutions using R's `kmeans` function, and selects as the final solution the one that has the minimum total within-cluster sum of squared distances.

**Usage**

```
KMeans(x, centers, iter.max=10, num.seeds=10)
```

**Arguments**

<code>x</code>	A numeric matrix of data, or an object that can be coerced to such a matrix (such as a numeric vector or a dataframe with all numeric columns).
<code>centers</code>	The number of clusters in the solution.
<code>iter.max</code>	The maximum number of iterations allowed.
<code>num.seeds</code>	The number of different starting random seeds to use. Each random seed results in a different k-means solution.

**Value**

A list with components:

<code>cluster</code>	A vector of integers indicating the cluster to which each point is allocated.
<code>centers</code>	A matrix of cluster centres (centroids).
<code>withinss</code>	The within-cluster sum of squares for each cluster.
<code>tot.withinss</code>	The within-cluster sum of squares summed across clusters.
<code>betweenss</code>	The between-cluster sum of squared distances.
<code>size</code>	The number of points in each cluster.

**Author(s)**

Dan Putler

**See Also**

[kmeans](#)



---

read.folder	<i>Read.folder</i>
-------------	--------------------

---

**Description**

lee carpetas enteras de datos en archivos .txt con el formato establecido (centroidex,centroidey,area,orientacion,largo,ancho)

**Usage**

```
read.folder()
```

**Details**

Esta funcion sirve para leer los datos tanto de las imagenes de corte horizontal como las imagenes de corte vertical.

**Author(s)**

Camilo Alberto Herrera Roza < hr.camilo@hrglobalideas.com>

**Examples**

```
Carpetas<-read.folder()
```

---

reconsBeta	<i>reconstruccion distribucion Beta y patrones Horizontales</i>
------------	---

---

**Description**

En esta funcion se puede realizar la recontruccion en 3D, para los patrones tanto horizontales como verticales conjuntamente.

**Usage**

```
reconsBeta(recortados, horizontales, zonaA = 1, zonaB = 1, grabar = FALSE)
```

**Arguments**

recortados	variable de entrada de datos, generalmente es una variable que contiene objetos que ha sido filtrados de errores (eje: vertical).
horizontales	variable de entrada de objetos del eje horizotal.
zonaA	En esta variable se ingresa la asignacion en al cual se encuentra el patron vertical de distribucion beta a ser graficado.
zonaB	En esta variable se ingresa la asignacion en al cual se encuentra el patron horizontal de distribucion poisson homogenea a ser graficado.
grabar	si grabar=TRUE se generara el grafico y posteriormente se grabara una animacion de 360° del patron conjunto encontrado.

## Details

se debe tener en cuenta que las asignaciones de zonaA y zonaB debes ser congruentes para realmente describir un patron de organizacion existente.

## Author(s)

Camilo Alberto Herrera Roza < hr.camilo@hrglobalideas.com>

## References

Cullen AC and Frey HC (1999) Probabilistic techniques in exposure assessment. Plenum Press, USA, pp. 81-155. Venables WN and Ripley BD (2002) Modern applied statistics with S. Springer, New York, pp. 435-446. Vose D (2000) Risk analysis, a quantitative guide. John Wiley & Sons Ltd, Chischester, England, pp. 99-143.

## Examples

```
## se genera un objetos de la clase (NeuroSpatCo) que contiene los datos (Verticales)
# Carpetas<-read.folder()
## se hace uso de la funcion distClust
# distancias<-distClust(Carpetas,0)
## se recortan los datos segun criterio del experto
# recortados<-trimExt(distancias,0.10)
## se genera un objetos de la clase (NeuroSpatCo) que contiene los datos (horizontales)
# hori<-read.folder()
## se genera la reconstruccion en 3D
# reconsBeta(recortados, horizontales, zonaA = 1, zonaB = 1, grabar = FALSE)
```

---

trimExt

*trimExt*


---

## Description

recorta datos extremos superiores e inferiores con el porcentaje de datos indicado.

## Usage

```
trimExt(distancias, percent)
```

## Arguments

distancias	nombre del objeto que contiene el conjunto de datos con las distancias encontradas entre las minicolumnas.
percent	porcentaje de recorte tanto superior como inferior.

**Author(s)**

Camilo Alberto Herrera Roza < hr.camilo@hrglobalideas.com>

**Examples**

```
# se genera un objetos de la clase (NeuroSpatCo)
Carpetas<-read.folder()
# se hace uso de la funcion distClust
distancias<-distClust(Carpetas,0)
#se recortan los datos segun criterio del experto
recortados<-trimExt(distancias,0.10)
```

# Index

## \*Topic \textasciitildekwd1

Cuadrant, [3](#)  
distClust, [3](#)  
grafKS, [4](#)  
imprDist, [5](#)  
KernWallis, [6](#)  
read.folder, [8](#)  
trimExt, [10](#)

## \*Topic \textasciitildekwd2

Cuadrant, [3](#)  
distClust, [3](#)  
grafKS, [4](#)  
imprDist, [5](#)  
KernWallis, [6](#)  
read.folder, [8](#)  
reconsBeta, [9](#)  
trimExt, [10](#)

## \*Topic \textasciitildereconstruction

reconsBeta, [9](#)

## \*Topic misc

assignCluster, [2](#)  
KMeans, [7](#)

## \*Topic package

NeuroSpatCo-package, [1](#)

assignCluster, [2](#)

Cuadrant, [3](#)  
cutree, [3](#)

distClust, [3](#)

grafKS, [4](#)

hclust, [3](#)

imprDist, [5](#)

KernWallis, [6](#)  
KMeans, [3](#), [7](#)  
kmeans, [3](#), [8](#)

NeuroSpatCo

(*NeuroSpatCo-package*), [1](#)

NeuroSpatCo-package, [1](#)

read.folder, [8](#)

reconsBeta, [9](#)

trimExt, [10](#)