Package 'cmR'

July 19, 2023

Type Package

Title Analysis of Cardiac Magnetic Resonance Images

Version 1.1

Date 2023-07-12

Author Volker Schmid [aut, cre]

Maintainer Volker Schmid <volker.schmid@lmu.de>

Depends R (>= 3.5.0)

Imports Matrix, splines, fields, graphics, parallel, plotrix

Description Computes maximum response from Cardiac Magnetic Resonance Images using spatial and voxel wise spline based Bayesian model. This is an implementation of the methods described in Schmid (2011) <doi:10.1109/TMI.2011.2109733> ``Voxel-Based Adaptive Spatio-Temporal Modelling of Perfusion Cardiovascular MRI". IEEE TMI 30(7) p. 1305 - 1313.

License GPL-3

RoxygenNote 7.2.3

Encoding UTF-8

LazyData true

URL https://bioimaginggroup.github.io/cmr/

Suggests knitr, rmarkdown, codetools, testthat (>= 3.0.0), R.rsp

VignetteBuilder knitr, R.rsp

BugReports https://github.com/bioimaginggroup/cmR/issues

NeedsCompilation no

Repository CRAN

Date/Publication 2023-07-19 10:30:02 UTC

R topics documented:

`bullseye` *Bullseye plot*

Description

Bullseye plot

Usage

```
bullseye(x, lim = NULL, reverse = TRUE, legend = TRUE, text = TRUE, cex = 1)
```

Arguments

`x`	vector of length 16 or 17
`lim`	limits of x values
`reverse`	boolean, reverse colors?
`legend`	boolean, add legend?
`text`	boolean, should text legend be added?
`cex`	cex for text legend

Value

plot

Examples

```
bullseye(1:16)
```

cmr *Bayesian analysis of cardiovascular magnetic resonance imaging*

Description

Bayesian analysis of cardiovascular magnetic resonance imaging

Usage

```
cmr(
  data,
  input,
  mask = NULL,
  method = "spatial",
  quantiles = c(0.25, 0.75),
  cores = parallel::detectCores()
)
```

Arguments

data	3D or 4D array of CMR signal
input	input function
mask	2d array of mask. Voxel with 0 or FALSE will be omitted from analysis. Default NULL: use NA values in data as mask
method	"spatial" or "local"
quantiles	quantiles used for credible interval, default: c(0.25, 0.75)
cores	number of cores for parallel computation. Spatial model only computes slices parallel, local can be parallelized on voxel level

Value

list of mbf (point estimation) and ci (credible interval)

cmr.local *Spline analysis of cardiovascular magnetic resonance imaging*

Description

Spline analysis of cardiovascular magnetic resonance imaging

Usage

```
cmr.local(data, mask, input, quantiles = c(0.25, 0.75), cores = 1)
```

Arguments

data	3d array of CMR signal
mask	2d array of mask. Voxel with 0 or FALSE will be omitted from analysis
input	input function
quantiles	quantiles used for credible interval, default: c(0.25, 0.75)
cores	number of cores to use in parallel computing

Value

list of mbf (point estimation) and ci (credible interval)

Examples

```
oldpar <- par(no.readonly = TRUE)
 library(cmR)
 data(cmrsim)
 local.mbf=local.ci=array(NA,c(30,30,3))
 for (i in 1:3){
  mask=array(NA,c(30,30))
  mask[cmrdata_sim[,,i,1]!=0]=1
  temp=cmr.local(cmrdata_sim[,,i,], mask, input_sim, cores=2)
  local.mbf[,,i]=t(as.matrix(temp$mbf))
  local.ci[,,i]=t(as.matrix(temp$ci))
 }
 par(mfrow=c(2,1))
 imageMBF(maxresp_sim, zlim=c(0,5))
 imageMBF(local.mbf, zlim=c(0,5))
 imageMBF(local.ci, zlim=c(0,0.8))
par(oldpar)
```

---

cmr.space *Spatial spline analysis of cardiovascular magnetic resonance imaging*

Description

Spatial spline analysis of cardiovascular magnetic resonance imaging

Usage

```
cmr.space(data, mask, input, quantiles = c(0.25, 0.75))
```

Arguments

data	3d array of CMR signal
mask	2d array of mask. Voxel with 0 or FALSE will be omitted from analysis
input	input function
quantiles	quantiles used for credible interval, default: c(0.25, 0.75)

Value

list of mbf (point estimation) and ci (credible interval)

Examples

```
oldpar <- par(no.readonly = TRUE)
 library(cmR)
 data(cmrsim)
 mask=array(NA,c(30,30))
 space.mbf=space.ci=array(NA,c(30,30,3))
 for (i in 1:3){
  mask=array(NA,c(30,30))
  mask[cmrdata_sim[,,i,1]!=0]=1
  temp=cmr.space(cmrdata_sim[,,i,], mask, input_sim)
  space.mbf[,,i]=t(as.matrix(temp$mbf))
  space.ci[,,i]=t(as.matrix(temp$ci))
  }
 par(mfrow=c(2,1))
 imageMBF(maxresp_sim, zlim=c(0,5))
 imageMBF(space.mbf, zlim=c(0,5))
 imageMBF(space.ci, zlim=c(0,0.8))
 par(oldpar)
```

---

cmrdata_sim *Simulated data for CMR package.*

Description

This data set is provided as example for the usage of the cmR package. cmrdata_sim is a simulated CMR image.

Usage

```
cmrdata_sim
```

Format

A 4D array, 30x30 pixels for 3 slices at 30 time points.

imageMBF *Plotting of (voxelwise) cardiac MBF*

Description

Plotting of (voxelwise) cardiac MBF

Usage

```
imageMBF(img, zlim = NULL, reverse = TRUE)
```

Arguments

img	3d array ob MBF values
zlim	limits of MBF, default: NULL means zlim=c(0,max(img,na.rm=TRUE))
reverse	reverse color scheme

Value

plots

Examples

```
data(cmrsim)
imageMBF(maxresp_sim)
```

input_sim *Simulated data for CMR package.*

Description

This data set is provided as example for the usage of the cmR package. input_sim is the simulated input function.

Usage

```
input_sim
```

Format

Vector for 30 time points.

maxresp_sim *Simulated data for CMR package.*

Description

This data set is provided as example for the usage of the cmR package. maxresp_sim is the true maximum response used in the simulation.

Usage

```
maxresp_sim
```

Format

A 3D array, 30x30 pixels for 3 slices.

pseudobullseye *Pseudo bullseye plot*

Description

Pseudo bullseye plot

Usage

```
pseudobullseye(
  x,
  lim = range(x, na.rm = TRUE),
  legend = FALSE,
  text = TRUE,
  reverse = FALSE,
  center = TRUE,
  cex = 1,
  legend.width = 1
)
```

Arguments

x	3D array
lim	limits of x values
legend	boolean, add legend?
text	boolean, should text legend be added?
reverse	boolean, reverse colors?
center	boolean, should input x be centered before plotting
cex	cex for text legend
legend.width	Width in characters of the legend strip.

Value

plots

Examples

```
data(cmrsim)
pseudobullseye(maxresp_sim)
```

---

rmvnormcanon *Draw random vectors from multivariate Gaussian in canonical form*

Description

Draw random vectors from multivariate Gaussian in canonical form

Usage

```
rmvnormcanon(n, b, P)
```

Arguments

n	Number of draws
b	b parameter
P	Precision matrix

Value

matrix with n columns, vector if n=1

Examples

```
P<-matrix(c(1,.5,.5,1),ncol=2)
b=c(2,0)
# expected value and covariance matrix
Sigma = solve(P)
mu = b%*%Sigma
# sample
x<-rmvnormcanon(1000,b,P)
mu.hat=apply(x,1,mean)
print(mu.hat-mu)
Sigma.hat=var(t(x))
print(Sigma.hat-Sigma)
```

Index