Package 'ETRep'

November 4, 2024

Type Package

Title Analysis of Elliptical Tubes Under the Relative Curvature Condition

Version 0.1.0

Maintainer Mohsen Taheri <MohsenTaheriShalmani@gmail.com>

Description Analysis of elliptical tubes with applications in biological modeling. The package is based on the references: Taheri, M., Pizer, S. M., & Schulz, J. (2024) ``The Mean Shape under the Relative Curvature Condition." arXiv <doi:10.48550/arXiv.2404.01043>. Mohsen Taheri Shalmani (2024) ``Shape Statistics via Skeletal Structures", PhD Thesis, University of Stavanger, Norway <doi:10.13140/RG.2.2.34500.23685>. Key features include constructing discrete elliptical tubes, calculating transformations, validating structures under the Relative Curvature Condition, computing means, and generating simulations. Supports intrinsic and non-intrinsic mean calculations and transformations, size estimation, plotting, and random sample generation based on a reference tube.

License MIT + file LICENSE

URL https://github.com/MohsenTaheriShalmani/Elliptical_Tubes

Depends R (>= 4.0.0)

Author Mohsen Taheri [aut, cre]

Encoding UTF-8

LazyData true

Imports rgl, shapes, Morpho, matlib, RiemBase, RSpincalc, rotations, SphericalCubature, Rvcg, fields, Matrix, pracma, truncnorm, ggplot2, reshape2, dplyr, ptinpoly

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2024-11-04 12:40:13 UTC

Contents

`check_Tube_Legality` *Check the Legality of an Elliptical Tube (ETRep)*

Description

Checks the validity of a given ETRep based on the Relative Curvature Condition (RCC) and principal radii such that forall i a_i>b_i.

Usage

```
check_Tube_Legality(tube)
```

Arguments

`tube`	List containing ETRep details.

Value

Logical value: TRUE if valid, FALSE otherwise.

Examples

```
# Load tube
data("colon3D")
check_Tube_Legality(tube = colon3D)
```

colon3D *Data*

Description

A colon sample as an elliptical tube.

Usage

```
colon3D
```

Format

A list containing the information of an e-tube

Source

Generated and stored in the package's 'data/' folder.

create_Elliptical_Tube *Create a Discrete Elliptical Tube (ETRep)*

Description

Constructs a discrete elliptical tube (ETRep) based on specified parameters.

Usage

```
create_Elliptical_Tube(
  numberOfFrames,
  method,
  materialFramesBasedOnParents = NA,
  initialFrame = diag(3),
  initialPoint = c(0, 0, 0),
  EulerAngles_Matrix = NA,
  ellipseResolution = 10,
  ellipseRadii_a,
  ellipseRadii_b,
  connectionsLengths,
  plotting = TRUE,
  add = FALSE
)
```

Arguments

numberOfFrames	Integer, specifies the number of consecutive material frames.
method	String, either "basedOnEulerAngles" or "basedOnMaterialFrames", defines the material frames method.
materialFramesBasedOnParents	Array (3 x 3 x numberOfFrames) with pre-defined material frames.
initialFrame	Matrix 3 x 3 as the initial frame
initialPoint	Real vector with three elemets as the initial point
EulerAngles_Matrix	Matrix of dimensions numberOfFrames x 3 with Euler angles to define material frames.
ellipseResolution	Integer, resolution of elliptical cross-sections (default is 10).
ellipseRadii_a	Numeric vector for the primary radii of cross-sections.
ellipseRadii_b	Numeric vector for the secondary radii of cross-sections.
connectionsLengths	Numeric vector for lengths of spinal connection vectors.
plotting	Logical, enables plotting of the ETRep (default is TRUE).
add	Logical, enables overlay plotting

Value

List containing tube details (orientation, radii, connection lengths, boundary points, etc.).

Examples

```
numberOfFrames<-15
EulerAngles_alpha<-c(rep(0,numberOfFrames))
EulerAngles_beta<-c(rep(-pi/20,numberOfFrames))
EulerAngles_gamma<-c(rep(0,numberOfFrames))
EulerAngles_Matrix<-cbind(EulerAngles_alpha,
                          EulerAngles_beta,
                          EulerAngles_gamma)
tube <- create_Elliptical_Tube(numberOfFrames = numberOfFrames,
                               method = "basedOnEulerAngles",
                               EulerAngles_Matrix = EulerAngles_Matrix,
                               ellipseResolution = 10,
                               ellipseRadii_a = rep(3, numberOfFrames),
                               ellipseRadii_b = rep(2, numberOfFrames),
                               connectionsLengths = rep(4, numberOfFrames),
                               plotting = FALSE)
 # Plotting
 plot_Elliptical_Tube(tube = tube,plot_frames = FALSE,
                      plot_skeletal_sheet = TRUE,
                      plot_r_project = FALSE,
                      plot_r_max = FALSE,add = FALSE)
```

intrinsic_Distance_Between2tubes *Calculating the intrinsic distance between two ETReps*

Description

Calculating the intrinsic distance between two ETReps

Usage

```
intrinsic_Distance_Between2tubes(tube1, tube2)
```

Arguments

tube1	List containing ETRep details.
tube2	List containing ETRep details.

Value

Numeric

Examples

```
# Load tubes
data("tube_A")
data("tube_B")
intrinsic_Distance_Between2tubes(tube1 = tube_A,tube2 = tube_B)
```

intrinsic_mean_tube *Calculate Intrinsic Mean of ETReps*

Description

Computes the intrinsic mean of a set of ETReps.

Usage

```
intrinsic_mean_tube(tubes, type = "sizeAndShapeAnalysis", plotting = TRUE)
```

Arguments

tubes	List of ETReps.
type	String, "ShapeAnalysis" or "sizeAndShapeAnalysis" (default is "sizeAndShapeAnalysis").
plotting	Logical, enables visualization of the mean (default is TRUE).

Value

List representing the mean ETRep.

Examples

```
#Example 1
# Load tubes
data("tube_A")
data("tube_B")
intrinsic_mean<-
  intrinsic_mean_tube(tubes = list(tube_A,tube_B),
                      plotting = FALSE)
# Plotting
plot_Elliptical_Tube(tube = intrinsic_mean,
                     plot_frames = FALSE,
                     plot_skeletal_sheet = FALSE,
                     plot_r_project = FALSE,
                     plot_r_max = FALSE,
                     add = FALSE)

#Example 2
data("simulatedColons")
intrinsic_mean<-
  intrinsic_mean_tube(tubes = simulatedColons,
                      plotting = FALSE)
# Plotting
plot_Elliptical_Tube(tube = intrinsic_mean,
                     plot_frames = FALSE,
                     plot_skeletal_sheet = FALSE,
                     plot_r_project = FALSE,
                     plot_r_max = FALSE,
                     add = FALSE)
```

---

intrinsic_Transformation_Elliptical_Tubes

Intrinsic Transformation Between Two ETReps

Description

Performs the intrinsic transformation from one ETRep to another.

Usage

```
intrinsic_Transformation_Elliptical_Tubes(
  tube1,
  tube2,
  type = "sizeAndShapeAnalysis",
  numberOfSteps = 5,
  plotting = TRUE,
```

```
  colorBoundary = "blue"
)
```

Arguments

tube1	List containing details of the first ETRep.
tube2	List containing details of the second ETRep.
type	String defining the type of analysis as sizeAndShapeAnalysis or shapeAnalysis
numberOfSteps	Integer, number of transformation steps.
plotting	Logical, enables visualization during transformation (default is TRUE).
colorBoundary	String defining the color of the e-tube

Value

List containing intermediate ETReps.

Examples

```
# Load tubes
data("tube_A")
data("tube_B")
numberOfSteps <- 10
transformation_Tubes<-
  intrinsic_Transformation_Elliptical_Tubes(
    tube1 = tube_A,tube2 = tube_B,
    numberOfSteps = numberOfSteps,
    plotting = FALSE)
# Plotting
for (i in 1:length(transformation_Tubes)) {
  plot_Elliptical_Tube(tube = transformation_Tubes[[i]],
  plot_frames = FALSE,plot_skeletal_sheet = FALSE
  ,plot_r_project = FALSE,
  plot_r_max = FALSE,
  add = FALSE)
}
```

---

nonIntrinsic_Distance_Between2tubes

Calculating the non-intrinsic distance between two ETReps

Description

Calculating the non-intrinsic distance between two ETReps

Usage

```
nonIntrinsic_Distance_Between2tubes(tube1, tube2)
```

Arguments

tube1	List containing ETRep details.
tube2	List containing ETRep details.

Value

Numeric

Examples

```
# Load tubes
data("tube_A")
data("tube_B")
intrinsic_Distance_Between2tubes(tube1 = tube_A,tube2 = tube_B)
```

---

nonIntrinsic_mean_tube

Calculate Non-Intrinsic Mean of ETReps

Description

Computes the non-intrinsic mean of a set of ETReps.

Usage

```
nonIntrinsic_mean_tube(tubes, type = "sizeAndShapeAnalysis", plotting = TRUE)
```

Arguments

tubes	List of ETReps.
type	String, "ShapeAnalysis" or "sizeAndShapeAnalysis" (default is "sizeAndShapeAnalysis").
plotting	Logical, enables visualization of the mean (default is TRUE).

Value

List representing the mean ETRep.

Examples

```
#Example 1
# Load tubes
data("tube_A")
data("tube_B")
nonIntrinsic_mean<-
  nonIntrinsic_mean_tube(tubes = list(tube_A,tube_B),
                         plotting = FALSE)
```

```
# Plotting
plot_Elliptical_Tube(tube = nonIntrinsic_mean,
                     plot_frames = FALSE,
                     plot_skeletal_sheet = FALSE,
                     plot_r_project = FALSE,
                     plot_r_max = FALSE,
                     add = FALSE)

#Example 2
data("simulatedColons")
nonIntrinsic_mean<-
  nonIntrinsic_mean_tube(tubes = simulatedColons,
                         plotting = FALSE)
# Plotting
plot_Elliptical_Tube(tube = nonIntrinsic_mean,
                     plot_frames = FALSE,
                     plot_skeletal_sheet = FALSE,
                     plot_r_project = FALSE,
                     plot_r_max = FALSE,
                     add = FALSE)
```

`nonIntrinsic_Transformation_Elliptical_Tubes`
Non-intrinsic Transformation Between Two ETReps

Description

Performs the non-intrinsic transformation from one ETRep to another.

Usage

```
nonIntrinsic_Transformation_Elliptical_Tubes(
  tube1,
  tube2,
  type = "sizeAndShapeAnalysis",
  numberOfSteps = 4,
  plotting = TRUE,
  colorBoundary = "blue",
  add = FALSE
)
```

Arguments

`tube1`	List containing details of the first ETRep.
`tube2`	List containing details of the second ETRep.
`type`	String defining the type of analysis as sizeAndShapeAnalysis or shapeAnalysis
`numberOfSteps`	Integer, number of transformation steps.
`plotting`	Logical, enables visualization during transformation (default is TRUE).

colorBoundary	String defining the color of the e-tube
add	Logical, enables overlay plotting

Value

List containing intermediate ETReps.

Examples

```
# Load tubes
data("tube_A")
data("tube_B")
numberOfSteps <- 10
transformation_Tubes<-
  nonIntrinsic_Transformation_Elliptical_Tubes(
    tube1 = tube_A,tube2 = tube_B,
    numberOfSteps = numberOfSteps,
    plotting = FALSE)
# Plotting
for (i in 1:length(transformation_Tubes)) {
  plot_Elliptical_Tube(tube = transformation_Tubes[[i]],
  plot_frames = FALSE,plot_skeletal_sheet = FALSE
  ,plot_r_project = FALSE,
  plot_r_max = FALSE,
  add = FALSE)
}
```

---

plot_Elliptical_Tube *Plot an Elliptical Tube (ETRep)*

Description

Plots a given ETRep with options for boundary, material frames, and projection visualization.

Usage

```
plot_Elliptical_Tube(
  tube,
  plot_boundary = TRUE,
  plot_r_max = FALSE,
  plot_r_project = TRUE,
  plot_frames = TRUE,
  frameScaling = NA,
  plot_spine = TRUE,
  plot_normal_vec = FALSE,
  plot_skeletal_sheet = TRUE,
  decorate = TRUE,
  colSkeletalSheet = "blue",
```

```
  colorBoundary = "blue",
  add = FALSE
)
```

Arguments

tube	List containing ETRep details.
plot_boundary	Logical, enables plotting of the boundary (default is TRUE).
plot_r_max	Logical, enables plotting of max projection size (default is FALSE).
plot_r_project	Logical, enables plotting of projection along normals (default is TRUE).
plot_frames	Logical, enables plotting of the material frames (default is TRUE).
frameScaling	Numeric, scale factor for frames.
plot_spine	Logical, enables plotting of the spine.
plot_normal_vec	Logical, enables plotting of the normals.
plot_skeletal_sheet	Logical, enables plotting of the surface skeleton.
decorate	Logical, enables decorate the plot
colSkeletalSheet	String, defining the color of the surface skeleton
colorBoundary	String, defining the color of the e-tube
add	Logical, enables overlay plotting

Value

Graphical output.

Examples

```
# Load tube
data("colon3D")
plot_Elliptical_Tube(tube = colon3D,
                     plot_frames = FALSE,
                     add=FALSE)
```

---

simulatedColons *Data*

Description

Simulated samples of e-tubes, modeled after a reference structure resembling a colon.

Usage

```
simulatedColons
```

Format

Five simulated samples of elliptical tubes, modeled after a reference structure resembling a colon.

Source

Generated and stored in the package's 'data/' folder.

`simulate_etube` *Simulate Random Elliptical Tubes (ETReps)*

Description

Generates random samples of ETReps based on a reference tube with added variation.

Usage

```
simulate_etube(
  referenceTube,
  numberOfSimulation,
  sd_v = 10^-10,
  sd_psi = 10^-10,
  sd_x = 10^-10,
  sd_a = 10^-10,
  sd_b = 10^-10,
  rangeSdScale = c(1, 2),
  plotting = TRUE
)
```

Arguments

`referenceTube`	List containing ETRep information as the reference.
`numberOfSimulation`	Integer, number of random samples.
`sd_v`	Standard deviations for various parameters.
`sd_psi`	Standard deviations for various parameters.
`sd_x`	Standard deviations for various parameters.
`sd_a`	Standard deviations for various parameters.
`sd_b`	Standard deviations for various parameters.
`rangeSdScale`	Numeric range for random scaling.
`plotting`	Logical, enables visualization of samples (default is FALSE).

Value

List of random ETReps.

Examples

```
# Load tube
data("colon3D")
#Set Parameters
sd_v<-sd_psi<-1e-03
sd_x<-sd_a<-sd_b<-1e-04
numberOfSimulation<-3
random_Tubes<-
  simulate_etube(referenceTube = colon3D,
                 numberOfSimulation = numberOfSimulation,
                 sd_v = sd_v,
                 sd_psi = sd_psi,
                 sd_x = sd_x,
                 sd_a = sd_a,
                 sd_b = sd_b,
                 rangeSdScale = c(1, 2),
                 plotting = FALSE)
# Plotting
rgl::open3d()
for (i in 1:numberOfSimulation) {
  plot_Elliptical_Tube(tube = random_Tubes[[i]],
                       plot_frames = FALSE,
                       plot_skeletal_sheet = FALSE,
                       plot_r_project = FALSE,
                       plot_r_max = FALSE,
                       add = TRUE)
}
```

---

tube_A *Data*

Description

A tube with 204 elliptical cross-sections.

Usage

```
tube_A
```

Format

A list containing the information of an e-tube with 204 elliptical cross-sections

Source

Generated and stored in the package's 'data/' folder.

tube_B *Data*

Description

A tube with 204 elliptical cross-sections.

Usage

```
tube_B
```

Format

A list containing the information of an e-tube with 204 elliptical cross-sections

Source

Generated and stored in the package's 'data/' folder.

Index