Package ‘spherical.lasso’

November 18, 2020

Type Package

Title An R package which calculates the generalized spherical harmonics

Version 1.0

Author Nishan Mudalige, Peter Kim

Maintainer Nishan Mudalige <nishan.mudalige.1@ulaval.ca>

Description An R package to apply a modified version of the LASSO on data over the hyper-sphere.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 4.0)

Imports orthopolynom, pracma, matrixcalc, mefa, Directional, hypergeo, stringr, assertr, Matrix, glmnet

RoxygenNote 7.1.1

NeedsCompilation no

R topics documented:

`spherical.design` *A function to create the appropriate design matrix for data over the hypersphere*

Description

A function which creates the appropriate design matrix for regression analysis. The design matrix returned from this can be used with the spherical LASSO or with Beran’s regression estimator. We use the general spherical harmonics as defined by Dai and Xu in "Approximation Theory and Harmonic Analysis on Spheres and Balls. This package works up to the second order spherical order q-vector. Requires data in p>=3 dimensions.

Usage

```
spherical.design(x.entered, normalize.by.volume)
```

Arguments

x.entered Data matrix. Should be in p>=3 dimensions.

normalize.by.volume
Logical argument. Whether to normalize by the volume of the hyper-sphere. If TRUE, then the value returned with be normalized by the volume element.

Value

Design matrix after the spherical harmonics have been aplied, up to the second order q-vector.

Note

This research was supported by a grant from the Google Cloud Platform.

Author(s)

Nishan Mudalige

References

[1]Dai, F. and Xu, Y. (2013), Approximation Theory and Harmonic Analysis on Spheres and Balls, Springer Science & Business Media.

[2] Beran, R.(1979), Exponential Models for Directional Data, Ann Statist 71162-1178.

Examples

```
## Requires simmd by Kent et. al.
## See Kent, J. T., Ganeiber A.M., Mardia, K.V. (2017), A New Unified Approach for the Simulation of a Wide Class

## Number of simulations to run
# n.to.sim = 3000

## Generate data from a Fisher-Bingham with chosen parameters
# x.mfb = rFisherBingham(nsim = n.to.sim, mu = c(0, 0, 0, 0, 234),
#                        Aplus = c(101, 101, 101, -303, 0))
# x.mfb = as.matrix(x.mfb)

## Create design matrix
# x.d = gen.spherical.design(x.mfb, T)
```

spherical.lasso *A function to evaluate a variation of the LASSO as applied to data over the hypersphere*

Description

A function which applied the LASSO on data over the hypersphere. The von Mises Fisher Kernel from the ‘Directional‘ package is used to evaluate the estimated density for the LASSO.

Usage

```
spherical.lasso(x.entered, q, normalize.by.volume, intercept, extra.info)
```

Arguments

`x.entered`	Data matrix. Should be in p>=3 dimensions.
`q`	Maximum order of the q vectors. Should be either 1 or 2. Note that q=2 generates the design matrix for both q=1 and q=2.
`normalize.by.volume`	Logical argument. Whether to normalize by the volume of the hyper-sphere. If TRUE, then the value returned with be normalized by the volume element.
`intercept`	Logical argument. Whether the LASSO should include an intercept or not. If TRUE, then the model will include an intercept.
`extra.info`	Logical argument. Whether additional information should be returned. If TRUE, then object returned will be a list consisting of the model, the estimated values of the density for each data point and the design matrix.

Value

If ‘extra.info = TRUE‘

`model`	The LASSO model.
`y`	The estimated density evaluated using the von-Mises Fisher kernel density estimator.
`x.design`	The design matrix.

If ‘extra.info = FALSE‘, only the model is returned.

Note

This research was supported by a grant from the Google Cloud Platform.

Author(s)

Nishan Mudalige

References

[1]Dai, F. and Xu, Y. (2013), Approximation Theory and Harmonic Analysis on Spheres and Balls, Springer Science & Business Media.

[2] Beran, R.(1979), Exponential Models for Directional Data, Ann Statist 71162-1178.

Examples

```
## Requires simmd by Kent et. al.
## See Kent, J. T., Ganeiber A.M., Mardia, K.V. (2017), A New Unified Approach for the Simulation of a Wide Class

# set.seed(1)

# p = 4
# n = 100
# int.vec = c(1:100)
# kappa = 5

# mu = mu.vec(p)
# nu.mat = rand.nu.mat(p, int.vec, kappa)
# nu.mat.star = nu.mat/kappa

# x.mfb = rFisherBingham(nsim=n, mu=mu, Aplus=nu.mat, q=dimset(mu,nu.mat))
# x.mfb = as.data.frame(x.mfb)
# model.vmf = gen.spherical.lasso(x.mfb, 2, intercept=F, normalize.by.volume = F, extra.info=T)
```

Index