Package 'tensorordinal'

March 20, 2020

Type Package

Title Tensor Noise Reduction and Completion Based on Ordinal Observations

Version 0.1.0

Author Chanwoo Lee <chanwoo.lee@wisc.edu>, Miaoyan Wang <miaoyan.wang@wisc.edu>

Maintainer Chanwoo Lee <chanwoo.lee@wisc.edu>

Imports pracma, MASS, rTensor, methods

Description A set of tools for noise reduction and completion from ordinal tensor data with possibly missing values. The algorithms can be found in Lee and Wang (2020) <arXiv:2002.06524>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

NeedsCompilation no

Repository CRAN

Date/Publication 2020-03-20 13:00:02 UTC

R topics documented:

bic *Bayesian Information Criterion (BIC) value*

Description

Compute Bayesian Information Criterion (BIC) given a parameter tensor, an observed tensor, the dimension, and the rank

Usage

```
bic(ttnsr,theta,omega,d,r)
```

Arguments

ttnsr	an observed tensor
theta	a continuous-valued tensor (latent parameters)
omega	the cut-off points
d	dimension of the tensor
r	rank of the tensor

Value

BIC value at given inputs

estimation *Estimation of tensor entries from the cumulative model*

Description

Estimate the ordinal-valued tensor entries given latent parameters and a type of estimations

Usage

```
estimation(theta,omega,type = c("mode","mean","median"))
```

Arguments

theta	a continuous-valued tensor (latent parameters)
omega	the cut-off points
type	type of estimations: "mode" specifies argmax based label estimation "mean" specifies mean based label estimation "median" specifies median based label estimation

Value

an estimated ordinal tensor given latent parameters and a type of estimations

References

Lee, C., & Wang, M. (2020). Tensor denoising and completion based on ordinal observations. *arXiv preprint arXiv:2002.06524*.

Examples

```
indices <- c(10,20,30)
arr <- array(runif(prod(indices),-2,2),dim = indices)
b <- c(-1.5,0,1.5)
r_predict <- estimation(arr,b,type = "mode");r_predict
```

`fit_continuous` *Fitting the Tucker model to a tensor*

Description

Fit the Tucker model to a tensor with possibly missing values

Usage

```
fit_continuous(ttnsr,r,alpha = TRUE)
```

Arguments

`ttnsr`	an observed tensor
`r`	a rank to be fitted (Tucker rank)
`alpha`	a signal level `alpha = TRUE` if the signal level is unknown

Value

a list containing the following:

`C` - an estimated core tensor

`A` - estimated factor matrices

`iteration` - the number of iterations

`cost` - log-likelihood value at each iteration

Examples

```
# Latent parameters
library(rTensor)
alpha = 10
A_1 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
A_2 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
A_3 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
C = as.tensor(array(runif(2^3,min=-1,max=1),dim = c(2,2,2)))
theta = ttm(ttm(ttm(C,A_1,1),A_2,2),A_3,3)@data
theta = alpha*theta/max(abs(theta))
adj = mean(theta)
theta = theta-adj
omega = c(-0.2,0.2)+adj

# Observed tensor
ttnsr <- realization(theta,omega)@data

# Estimation of parameters
continuous_est = fit_continuous(ttnsr,c(2,2,2),alpha = 10)
```

`fit_ordinal` *Fitting the cumulative logistic model to an ordinal tensor*

Description

Fit the cumulative logistic model to an ordinal data tensor

Usage

```
fit_ordinal(ttnsr,r,omega=TRUE,alpha = TRUE)
```

Arguments

`ttnsr`	an observed tensor
`r`	a rank to be fitted (Tucker rank)
`omega`	the cut-off points if known, `omega = TRUE` if unknown
`alpha`	a signal level `alpha = TRUE` if the signal level is unknown

Value

a list containing the following:

`C` - an estimated core tensor

`A` - estimated factor matrices

`theta` - an estimated latent parameter tensor

`iteration` - the number of iterations

`cost` - log-likelihood value at each iteration

`omega` - estimated cut-off points

References

Lee, C., & Wang, M. (2020). Tensor denoising and completion based on ordinal observations. *arXiv preprint arXiv:2002.06524*.

Examples

```
# Latent parameters
library(rTensor)
alpha = 10
A_1 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
A_2 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
A_3 = matrix(runif(15*2,min=-1,max=1),nrow = 15)
C = as.tensor(array(runif(2^3,min=-1,max=1),dim = c(2,2,2)))
theta = ttm(ttm(ttm(C,A_1,1),A_2,2),A_3,3)@data
theta = alpha*theta/max(abs(theta))
adj = mean(theta)
theta = theta-adj
omega = c(-0.2,0.2)+adj

# Observed tensor
ttnsr <- realization(theta,omega)@data

# Estimation of parameters
ordinal_est = fit_ordinal(ttnsr,c(2,2,2),omega = TRUE,alpha = 10)
```

---

likelihood *Log-likelihood function (cost function)*

Description

Return log-likelihood function (cost function) value evaluated at a given parameter tensor, an observed tensor, and cut-off points

Usage

```
likelihood(ttnsr,theta,omega,type = c("ordinal","Gaussian"))
```

Arguments

ttnsr	an observed tensor data
theta	a continuous-valued tensor (latent parameters)
omega	the cut-off points
type	types of log-likelihood function "ordinal" specifies log-likelihood function based on the cumulative logistic model "Gaussian" specifies log-likelihood function based on the Gaussian model

Value

log-likelihood value at given inputs

---

realization *An ordinal tensor randomly simulated from the cumulative model*

Description

Simulate an ordinal tensor from the cumulative logistic model with the parameter tensor and the cut-off points

Usage

```
realization(theta,omega)
```

Arguments

theta	a continuous-valued tensor (latent parameters)
omega	the cut-off points

Value

an ordinal tensor randomly simulated from the cumulative logistic model

References

Lee, C., & Wang, M. (2020). Tensor denoising and completion based on ordinal observations. *arXiv preprint arXiv:2002.06524*.

Examples

```
indices <- c(10,20,30)
arr <- array(runif(prod(indices)),dim = indices)
b <- qnorm((1:3)/4)
r_sample <- realization(arr,b);r_sample
```

Index