Package: gcTensor (via r-universe)

August 31, 2024

Type Package

Title Generalized Coupled Tensor Factorization

Version 1.0.0

Suggests testthat

Depends R (>= 4.1.0)

Imports rTensor, einsum

Description Multiple matrices/tensors can be specified and decomposed simultaneously by Probabilistic Latent Tensor Factorisation (PLTF). See the reference section of GitHub README.md <https://github.com/rikenbit/gcTensor>, for details of the method.

License MIT + file LICENSE

URL https://github.com/rikenbit/gcTensor

Repository https://rikenbit.r-universe.dev

RemoteUrl https://github.com/rikenbit/gctensor

RemoteRef HEAD

RemoteSha 8a5c05b74e73493fe6a81443b2811da3f41f2368

Contents

`gcTensor-package` *Generalized Coupled Tensor Factorization*

Description

Multiple matrices/tensors can be specified and decomposed simultaneously by Probabilistic Latent Tensor Factorisation (PLTF). See the reference section of GitHub README.md <https://github.com/rikenbit/gcTensor>, for details of the method.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.

Author(s)

Koki Tsuyuzaki [aut, cre]

Maintainer: Koki Tsuyuzaki <k.t.the-answer@hotmail.co.jp>

References

Y. Kenan Yilmaz, et. al., (2011). Generalised Coupled Tensor Factorisation, NIPS

Beyza Ermis, et. al., (2015). Link prediction in heterogeneous data via generalized coupled tensor factorization, Data Mining and Knowledge Discovery

Examples

```
ls("package:gcTensor")
```

`GCTF` *Generalised Coupled Tensor Factorisation (GCTF)*

Description

The input data is assumed to be a list containing multiple matrices. GCTF decomposes N matrices (Xs) to M low-dimensional factor matices (Zs).

Usage

```
GCTF(X, R, M=NULL, pseudocount=.Machine$double.eps, initZ=NULL, fix=NULL, Ranks, Beta=1,
    num.iter=30, thr=1E-10, verbose=FALSE)
```

Arguments

`X`	A list containing N data matrices.
`R`	Coupling matrix which has N-rows and M-columns. N is the number of data matrices and M is the number of factor matrices decomposed by GCTF algorithm. If i-th data (X_i) has j-th factor matrix (Z_j), 1 is filled in R[i,j], otherwise 0.
`M`	A list containing N mask matrices. If in n-th data matrix, i-th row/j-th column is missing value, 0 is filled, otherwise 0. Default value is NULL, which means all the values are filled with 1 (No missing value).
`pseudocount`	The pseudo count to avoid zero division, when the element is zero (Default: Machine Epsilon).
`initZ`	A M-length list, which is the initial values of factor matrix Z. If not specified, random matrices are generated and used (Default: NULL).
`fix`	Whether each factor matrix Z is updated in each iteration step (Default: NULL, which means all Zs are updated).
`Ranks`	A M-length list, which is the correspondence between the dimension of data matrices and the lower dimension of factor matrices.
`Beta`	The parameter of Beta-divergence. Beta=0, 1, and 2 each mean Euclid Distance, KL-divergence, and Itakura-Saito divergence between the data matrices and reconstructed matrices by factor matrices (Default: 1).
`num.iter`	The number of interation step (Default: 30).
`thr`	When error change rate is lower than thr, the iteration is terminated (Default: 1E-10).
`verbose`	If verbose == TRUE, Error change rate is generated in console window (Default: FALSE).

Value

U : A matrix which has N-rows and J-columns (J < N, M). V : A matrix which has M-rows and J-columns (J < N, M). J : The number of dimension (J < N, M). RecError : The reconstruction error between data tensor and reconstructed tensor from U and V. TrainRecError : The reconstruction error calculated by training set (observed values specified by M). TestRecError : The reconstruction error calculated by test set (missing values specified by M). TrainRecError : The reconstruction error calculated by training set (observed values specified by M). TestRecError : The reconstruction error calculated by test set (missing values specified by M). RelChange : The relative change of the error. Trial : All the results of the trials to estimate the rank. Runtime : The number of the trials to estimate the rank. RankMethod : The rank estimation method.

Author(s)

Koki Tsuyuzaki

References

Y. Kenan Yilmaz, et. al., (2011). Generalised Coupled Tensor Factorisation, NIPS

Beyza Ermis, et. al., (2015). Link prediction in heterogeneous data via generalized coupled tensor factorization, Data Mining and Knowledge Discovery

Examples

```
if(interactive()){
# Simulation Datasets
set.seed(123)

# I times J times K
X1 <- rand_tensor(modes = c(4, 5, 6))
X1 <- X1@data^2
names(dim(X1)) <- c("I", "J", "K")

# I times P
X2 <- matrix(runif(4 * 7), nrow=4, ncol=7)
names(dim(X2)) <- c("I", "M")

# J times Q
X3 <- matrix(runif(5 * 8), nrow=5, ncol=8)
names(dim(X3)) <- c("J", "N")

# Coupled Tensor/Matrix
X <- list(X1 = X1, X2 = X2, X3 = X3)

# Coupling matrix R (CP)
R_CP <- rbind(
    c(1,1,1,0,0),
    c(1,0,0,1,0),
    c(0,1,0,0,1)
)
rownames(R_CP) <- paste0("X", seq(3))
colnames(R_CP) <- LETTERS[seq(5)]

# Size of Factor matrices (CP)
Ranks_CP <- list(
    A=list(I=4, r=3),
    B=list(J=5, r=3),
    C=list(K=6, r=3),
    D=list(M=7, r=3),
    E=list(N=8, r=3))

# Coupling matrix R (Tucker)
R_Tucker <- rbind(
    c(1,1,1,1,0,0),
    c(1,0,0,0,1,0),
    c(0,1,0,0,0,1)
)
rownames(R_Tucker) <- paste0("X", seq(3))
colnames(R_Tucker) <- LETTERS[seq(6)]

# Size of Factor matrices (Tucker)
Ranks_Tucker <- list(
    A=list(I=4, p=3),
    B=list(J=5, q=4),
    C=list(K=6, r=3),
```

```
    D=list(p=3, q=4, r=3),
    E=list(M=7, p=3),
    F=list(N=8, q=4))

# CP
out.CP_EUC <- GCTF(X, R_CP, Ranks=Ranks_CP, Beta=0, verbose=TRUE)
out.CP_KL <- GCTF(X, R_CP, Ranks=Ranks_CP, Beta=1, verbose=TRUE)
out.CP_IS <- GCTF(X, R_CP, Ranks=Ranks_CP, Beta=2, verbose=TRUE)

# Tucker
out.Tucker_EUC <- GCTF(X, R_Tucker, Ranks=Ranks_Tucker, Beta=0, verbose=TRUE)
out.Tucker_KL <- GCTF(X, R_Tucker, Ranks=Ranks_Tucker, Beta=1, verbose=TRUE)
out.Tucker_IS <- GCTF(X, R_Tucker, Ranks=Ranks_Tucker, Beta=2, verbose=TRUE)
}
```

Index