

Package ‘gsrs’

May 8, 2026

Type Package

Title A Group-Specific Recommendation System

Version 0.1.1

Author Yifei Zhang, Xuan Bi

Maintainer Yifei Zhang <yz3925@columbia.edu>

Description A group-specific recommendation system to use dependency information from users and items which share similar characteristics under the singular value decomposition framework. Refer to paper A Group-Specific Recommender System <[doi:10.1080/01621459.2016.1219261](https://doi.org/10.1080/01621459.2016.1219261)> for the details.

License GPL

Imports MASS,foreach,doParallel

Encoding UTF-8

RoxygenNote 7.1.1

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2021-08-02 08:10:05 UTC

Contents

gssvd	2
Index	5

gssvd

*Train the group-specific model and test model performance***Description**

This `gssvd()` function uses ratings dataset to train a group-specific recommender system, tests the performance, and output the key matrix for prediction. To make the training process run in parallel, `doParallel` package is recommended to use. For more details regarding how the simulated dataset created, please refer to <http://dx.doi.org/10.1080/01621459.2016.1219261>.

Usage

```
gssvd(
  train,
  test,
  B = 10,
  C = 10,
  K,
  tol_1 = 0.001,
  tol_2 = 1e-05,
  lambda = 2,
  max_iter = 100,
  verbose = 0,
  user_group = NULL,
  item_group = NULL
)
```

Arguments

<code>train</code>	Train set, a matrix with three columns (userID, movieID, ratings)
<code>test</code>	Test set, a matrix with three columns (userID, movieID, ratings)
<code>B</code>	Number of user groups, 10 by default, don't need to specify if <code>user_group</code> parameter is not NULL
<code>C</code>	Number of item groups, 10 by default, don't need to specify if <code>item_group</code> parameter is not NULL
<code>K</code>	Number of latent factors
<code>tol_1</code>	The stopping criterion for outer loop in the proposed algorithm, 1e-3 by default
<code>tol_2</code>	The stopping criterion for sub-loops, 1e-5 by default
<code>lambda</code>	Value of penalty term in ridge regression for ALS, 2 by default
<code>max_iter</code>	Maximum number of iterations in the training process, 100 by default
<code>verbose</code>	Boolean, if print out the detailed intermediate computations in the training process, 0 by default
<code>user_group</code>	Optional parameter, should be a n-dim vector, n is total number of users, each element in the vector represents the group ID for that user (We will use missing pattern if not specified)

`item_group` Optional parameter, should be a m-dim vector, m is total number of items, each element in the vector represents the group ID for that item (We will use missing pattern if not specified)

Value

Return the list of result, including matrix P, Q, S, T and RMSE of test set (RMSE_Test)

Author(s)

Yifei Zhang, Xuan Bi

References

Xuan Bi, Annie Qu, Junhui Wang & Xiaotong Shen A Group-Specific Recommender System, Journal of the American Statistical Association, 112:519, 1344-1353 DOI: 10.1080/01621459.2016.1219261. Please contact the author should you encounter any problems A fast version written in Matlab is available at <https://sites.google.com/site/xuanbigts/software>.

Examples

```
## Training model on the simulated data file
library(doParallel)
registerDoParallel(cores=2)
# CRAN limits the number of cores available to packages to 2,
# you can use cores = detectCores()-1 in the real work setting.
getDoParWorkers()
example_data_path = system.file("extdata", "sim_data.txt", package="gsrs")
ratings = read.table(example_data_path, sep=":", header = FALSE)[1:100,]
# Initialization Parameters
K=3
B=10
C=10
lambda = 2
max_iter = 1 # usually more than 10;
tol_1=1e-1
tol_2=1e-1
# Train Test Split
N=dim(ratings)[1]
test_rate = 0.3
train.row=which(rank(ratings[, 1]) <= floor((1 - test_rate) * N))
test.row=which(rank(ratings[, 1]) > floor((1 - test_rate) * N))
train.data=ratings[train.row,1:3]
test.data=ratings[test.row,1:3]
# Call gssvd function
a = gssvd(train=train.data, test=test.data, B=B, C=C, K=K,
lambda=lambda, max_iter=max_iter, verbose=1)
stopImplicitCluster()
# Output the result
a$RMSE_Test
head(a$P)
head(a$Q)
```

```
head(a$S)  
head(a$T)
```

Index

gssvd, [2](#)