Package ‘IBCF.MTME’

May 7, 2026

Title Item Based Collaborative Filtering for Multi-Trait and Multi-Environment Data

Version 1.6-0

Language en-US

Date 2019-03-17

Maintainer Francisco Javier Luna-Vazquez <`frahik@gmail.com`>

Description Implements the item based collaborative filtering (IBCF) method for continues phenotypes in the context of plant breeding where data are collected for various traits that were studied in various environments proposed by Montesinos-López et al. (2017) <doi:10.1534/g3.117.300309>.

Depends R (>= 3.0.0)

License LGPL-3

Encoding UTF-8

LazyData true

Type Package

RoxygenNote 6.1.1

URL `https://github.com/frahik/IBCF.MTME`

BugReports `https://github.com/frahik/IBCF.MTME/issues/new`

NeedsCompilation no

Collate CrossValidation.R IBCF.R IBCF.MTME.R IBCFYear.R methods.R Td2M.R Title.R Wheat_IBCF.R Year_IBCF.R

Imports lsa, tidyr, dplyr

Suggests testthat, knitr, rmarkdown, covr

Author Francisco Javier Luna-Vazquez [aut, cre] (ORCID: <`https://orcid.org/0000-0002-5370-7152`>),
Osval Antonio Montesinos-Lopez [aut] (ORCID: <`https://orcid.org/0000-0002-3973-6547`>),
Abelardo Montesinos-Lopez [aut],
Jose Crossa [aut] (ORCID: <`https://orcid.org/0000-0001-9429-5855`>)

Repository CRAN

Date/Publication 2019-03-23 05:18:07 UTC

Contents

`barplot.IBCFY` *barplot.IBCFY*

Description

Barplot of the results from IBCFY object

Usage

```
## S3 method for class 'IBCFY'
barplot(height, select = "Pearson", ...)
```

Arguments

`height`	`IBCFY object` IBCFY object, result of use the IBCF.Years() function
`select`	`character` By default ('Pearson'), plot the Pearson Correlations of the IBCF Object, else ('MAAPE'), plot the MAAPE of the IBCF Object.
`...`	Further arguments passed to or from other methods.

`CV.RandomPart` *Cross-Validation with Random Partitions*

Description

This method consists of randomly dividing the training data set and the test data set. For each division, the approximation function is adjusted from the training data and calculates the output values for the test data set. The result corresponds to the arithmetic mean of the values obtained for the different divisions.

Usage

```
CV.RandomPart(DataSet, NPartitions = 10, PTesting = 0.35,
  Traits.testing = NULL, Set_seed = NULL)
```

Arguments

`DataSet`	`data.frame` The data set object is a data.frame object that contains 4 columns in the Tidy data format: `$Line` is the Line or genotype identifier, and the name of this column could change. `$Env` is the name of the evaluated environment (s). `$Trait` is the name of the evaluated trait (s). `$Response` Variable response obtained for the row corresponding to line and environment.
`NPartitions`	`integer` Number of Partitions for the Cross-Validation. Is 10 by default.
`PTesting`	`Double` Percentage of Testing for the Cross-Validation. Is 0.35 by default.
`Traits.testing`	`character` By default is null and use all the traits to fit the model, else only part of the traits specified be used to fit the model.
`Set_seed`	`integer` Number of seed for reproducible research. Is NULL by default.

Value

`List` A list object with length of `NPartitions`, every index has a `matrix` $n \times x$, where n is the number of `NLines` and x is the number of `NEnv` × `NTraits`. The values inside is 1 for training and 2 for testing.

Examples

```
## Not run:
  library(IBCF.MTME)
  data('Wheat_IBCF')

  CV.RandomPart(Wheat_IBCF)
  CV.RandomPart(Wheat_IBCF, NPartitions = 10)
  CV.RandomPart(Wheat_IBCF, Traits.testing = 'DH')
  CV.RandomPart(Wheat_IBCF, NPartitions = 10, PTesting = .35)
  CV.RandomPart(Wheat_IBCF, NPartitions = 10, Traits.testing = 'DH')
  CV.RandomPart(Wheat_IBCF, NPartitions = 10, PTesting = .35, Set_seed = 5)
  CV.RandomPart(Wheat_IBCF, NPartitions = 10, PTesting = .35, Traits.testing = 'DH')
```

```
 CV.RandomPart(Wheat_IBCF, NPartitions = 10, PTesting = .35, Traits.testing = 'DH', Set_seed = 5 )

## End(Not run)
```

`getMatrixForm` *Tidy data format to Matrix format*

Description

Tidy data format to Matrix format

Usage

```
getMatrixForm(Tidy_DataSet, onlyTrait = FALSE)
```

Arguments

`Tidy_DataSet`	`data.frame` object that contains 4 columns: `$Line`: Line or genotype identifier, and the name of this column could change. `$Env`: Name of the evaluated environment (s). `$Trait`: Name of the evaluated trait (s). `$Response`: Variable response obtained for the row corresponding to line and environment.
`onlyTrait`	`logical` by default is `FALSE`, if is `TRUE` only the column `$Trait` is transformed.

Value

A `data.frame` object with the `$Response` divided by `$Traits` columns.

Examples

```
## Not run:
  data('Wheat_IBCF')
  M <- getMatrixForm(Wheat_IBCF)

## End(Not run)

## Not run:
  data('Year_IBCF')
  M.Y <- getMatrixForm(Year_IBCF, onlyTrait = T)

## End(Not run)
```

`getTidyForm` *Matrix format to Tidy data format*

Description

Matrix format to Tidy data format

Usage

```
getTidyForm(Matrix_DataSet, onlyTrait = FALSE)
```

Arguments

`Matrix_DataSet`	A data.frame object with the response values divided in n environments or traits columns
`onlyTrait`	`logical` by default is `FALSE`, if is `TRUE` only is considered the `$Trait` column.

Value

A `data.frame` object with the `$Response` divided by `$Traits` columns.

Examples

```
## Not run:
  data('Wheat_IBCF')
  M <- getMatrixForm(Wheat_IBCF)
  Tidy <- getTidyForm(M)

## End(Not run)

## Not run:
  data('Year_IBCF')
  M.Y <- getMatrixForm(Year_IBCF, onlyTrait = T)
  Tidy.Y <- getTidyForm(M.Y, onlyTrait = T)

## End(Not run)
```

`IBCF` *IBCF*

Description

Item Based Collaborative Filtering for multi-trait and multi-environment data.

Usage

```
IBCF(object, dec = 4)
```

Arguments

`object`	`list` CrossValidation object, is obtained from CV.RandomPartition function.
`dec`	`integer` Number of decimals to print in the results.

Value

A list with the next components

`NPartitions`	`integer` Number of partitions used for testing data
`predictions_Summary`	`data.frame` A data.frame with the results of the test
`Predictions`	`list` A list with the predicted results for each partition

Examples

```
 ## Not run:
  library(IBCF.MTME)
  data('Wheat_IBCF')

  CV <- CV.RandomPart(Wheat_IBCF)
  IBCF(CV)

## End(Not run)
```

`IBCF.MTME` *IBCF.MTME: Item Based Collaborative Filtering for Multi-Trait and Multi-Environment Data.*

Description

The Item Based Collaborative Filtering for Multi-Trait and Multi-Environment Data (IBCF.MTME) package was developed to implement the item based collaborative filtering (IBCF) method for continues phenotypes in the context of plant breeding where data are collected for various traits that were studied in various environments. It is important to point out that the main difference of this package with the available packages that can implement IBCF is that this package was developed for continuous phenotypes which cannot be implemented in the current packages that can implement IBCF that only work for binary and ordinary phenotypes.

`IBCF.Years` *IBCF.Years*

Description

Item Based Collaborative Filtering for Years data

Usage

```
IBCF.Years(DataSet, colYears = 1, colID = 2, Years.testing = "",
  Traits.testing = "", dec = 4)
```

Arguments

`DataSet`	`data.frame` A data set in Matrix Form.
`colYears`	`string` or `integer` A name or the position of the 'Years' column just in case that is not the first column.
`colID`	`string` or `integer` A name or the position of the 'ID' column, just in case that is not the second column.
`Years.testing`	`vector` A vector with the names of the years to use in test.
`Traits.testing`	`vector` A vector with the names of the traits to use in test.
`dec`	`integer` Number of decimals to print in the results.

Value

A list with the next components

`Year.testing`	`vector` a vector with the Years used for the testing data
`Traits.testing`	`vector` a vector with the Traits used for the testing data
`Data_Obs_Pred`	`data.frame` Contains the values observed and predicted (the predicted values has '.1' after the name)
`predictions_Summary`	`data.frame` Contains the summary of the correlation of the predictions and the MAAPE

Examples

```
## Not run:
  library(IBCF.MTME)
  data('Year_IBCF')
  DataSet <- getMatrixForm(Year_IBCF, onlyTrait = TRUE)
  IBCF.Years(DataSet , Years.testing = c('2015', '2016'), Traits.testing = c('T5', 'T6'))


## End(Not run)
```

`plot.IBCF` *Plot IBCF graph*

Description

Plot from IBCF object

Usage

```
## S3 method for class 'IBCF'
plot(x, select = "Pearson", ...)
```

Arguments

`x`	`IBCF object` IBCF object, result of use the IBCF() function
`select`	`character` By default ('Pearson'), plot the Pearson Correlations of the IBCF Object, else ('MAAPE'), plot the MAAPE of the IBCF Object.
`...`	Further arguments passed to or from other methods.

`print.IBCF` *Print IBCF information object*

Description

Print IBCF information object

Usage

```
## S3 method for class 'IBCF'
print(x, ...)
```

Arguments

`x`	IBCF object
`...`	Further arguments passed to or from other methods.

Value

printeable object

`print.IBCFY` *Print IBCFY information object*

Description

Print IBCFY information object

Usage

```
## S3 method for class 'IBCFY'
print(x, ...)
```

Arguments

`x`	IBCFY object
`...`	Further arguments passed to or from other methods.

Value

printeable object

`summary.IBCF` *Summary*

Description

Summary of IBCF object

Usage

```
## S3 method for class 'IBCF'
summary(object, information = "compact", digits = 4,
  ...)
```

Arguments

`object`	`IBCF object` IBCF object, result of use the IBCF() function
`information`	`string` ...
`digits`	`numeric` ...
`...`	Further arguments passed to or from other methods.

`summary.IBCFY` *Summary*

Description

Summary of IBCFY object

Usage

```
## S3 method for class 'IBCFY'
summary(object, digits = 4, ...)
```

Arguments

`object`	`IBCFY object` IBCFY object, result of use the IBCF.Years() function
`digits`	`numeric` Number of digits of the output.
`...`	Further arguments passed to or from other methods.

`Wheat_IBCF` *Wheat Data*

Description

The package includes a data set based on a portion of the data used in the study of Montesinos-Lopez, O. A.; Montesinos-Lopez, A.; Crossa, J.; Toledo, F. H.; Montesinos-Lopez, J. C.; Singh, P. & Salinas-Ruiz, J. (2017). A Bayesian Poisson-lognormal Model for Count Data for Multiple-Trait Multiple-Environment Genomic-Enabled Prediction. G3: Genes|Genomes|Genetics 7(5):1595–1606. http://doi.org/10.1534/g3.117.039974. The data set consists of 250 wheat lines evaluated in 3 environments and 4 distinct traits, i.e. 3000 observations.

Usage

```
data(Wheat_IBCF)
```

Format

a `TidyData` format, 3000 row per 4 columns.

Author(s)

Montesinos-Lopez, O. A.

`Year_IBCF` *Year_IBCF Data*

Description

Dataset based on simulated data with the next code:

```
set.seed(2)
A <- matrix(0.65,ncol=12,nrow=12)
diag(A) <- 1
Sdv <- diag(c(0.9^0.5,0.8^0.5,0.9^0.5,0.8^0.5,0.86^0.5,0.7^0.5,0.9^0.5,0.8^0.5,0.9^0.5,0.7^0.5,0.7^0

Sigma <- Sdv
No.Lines <- 60
Z <- rmvnorm(No.Lines,mean=c(5,5.5,6,5.5,7,6.5,6.0,7,6.6,8,6.3,8),sigma=Sigma)
Years <- c(rep(2014,20),rep(2015,20),rep(2016,20))
Gids <- c(1:No.Lines)

Data.Final <- data.frame(cbind(Years,Gids,Z))

colnames(Data.Final) <- c("Years","Gids","T1","T2","T3","T4","T5","T6","T7","T8","T9","T10","T11","T
head(Data.Final)
Year_IBCF <- getTidyForm(Data.Final, onlyTrait = T)
```

Usage

```
data(Year_IBCF)
```

Format

a `TidyData` format, 750 row per 4 columns.

Author(s)

Montesinos-Lopez, O. A.

Index