Package ‘grouper’

July 22, 2025

Type Package

Title Optimal Assignment of Students to Groups

Version 0.3.1

Description Integer programming models to assign students to groups by maximising diversity within groups, or by maximising preference scores for topics.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests knitr, ompr.roi, rmarkdown, ROI.plugin.glpk,

VignetteBuilder knitr

Imports cluster, dplyr, magrittr, ompr, rlang, yaml

Depends R (>= 3.5)

NeedsCompilation no

Author Vik Gopal [aut, cre],
Kevin Lam [aut],
Ju Xue [ctb],
National University of Singapore [cph]

Maintainer Vik Gopal <vik.gopal@nus.edu.sg>

Repository CRAN

Date/Publication 2025-07-09 10:20:06 UTC

Contents

`assign_groups` *Assigns model result to the original data frame.*

Description

From the result of `ompr::solve_model()`, this function attaches the derived groupings to the original dataframe comprising students.

Usage

```
assign_groups(
  model_result,
  assignment = c("diversity", "preference"),
  dframe,
  params_list,
  group_names
)
```

Arguments

`model_result` The output solution objection.

`assignment` Character string indicating the type of model that this dataset is for. The argument is either ’preference’ or ’diversity’. Partial matching is fine.

`dframe` The original dataframe used in `extract_student_info()`.

`params_list` The list of parameters from the YAML file, i.e. the output of `extract_params_yaml()`. This is only required for the preference-based assignment.

`group_names` A character string. It denotes the column name in the original dataframe containing the self-formed groups. Note that we need the string here, not the integer position, since we are going to join with it.

Value

A data frame with the group assignments attached to the original group composition dataframe.

`dba_gc_ex001` *DBA Group Composition Data Example 001*

Description

An example dataset to use with the diversity-based assignment model.

Usage

```
dba_gc_ex001
```

Format

`dba_gc_ex001`**:**

A data frame with 4 rows and 4 columns.

- id: the student id of each students, simply the integers 1 to 4.
- major: the primary major of each student.
- skill: the skill level of each student.
- groups: the self-formed groups submitted by each student. In this case, student is in his/her own group.

Source

This dataset was constructed by hand.

`extract_params_yaml` *Extract parameters from a YAML file*

Description

The remaining parameters for the models are retrieved from a YAML file, so as not to clutter the argument list for `extract_student_info()`.

Usage

```
extract_params_yaml(fname, assignment = c("diversity", "preference"))
```

Arguments

`fname`	A YAML file containing the remaining parameters.
`assignment`	Character string indicating the type of model that this dataset is for. The argument is either 'preference' or 'diversity'. Partial matching is fine.

Value

For the diversity+skill-based assignment, this function returns a list containing:

- n_topics: the number of topics
- R: the optimally desired number of repetitions per topic
- nmin: the minimum number of students per topic,
- nmax: the maximum number of students per topic,
- rmin: the minimum number of repetitions per topic,
- rmax: the maximum number of repetitions per topic.

For the preference-based assignment, this function returns a list containing:

- n_topics: the number of topics
- R: the optimally desired number of repetitions per topic
- nmin: the minimum number of students per topic,
- nmax: the maximum number of students per topic,
- rmin: the minimum number of repetitions per topic,
- rmax: the maximum number of repetitions per topic.

extract_student_info *Extract student information*

Description

Converts a dataframe with information on students to a list of parameters. This list forms one half of the inputs to prepare_model(). The other half comes from extract_params_yaml.

Usage

```
extract_student_info(
  dframe,
  assignment = c("diversity", "preference"),
  self_formed_groups,
  demographic_cols,
  skills,
  pref_mat
)
```

Arguments

`dframe` A dataframe with one row for each student. The columns could possibly contain demographic variables, an overall skill measure, and a column indicating self-formed groups. It is best to have an id column to identify each student.

`assignment` Character string indicating the type of model that this dataset is for. The argument is either 'preference' or 'diversity'. Partial matching is fine.

`self_formed_groups`
An integer column that identifies the self-formed groups, submitted by students.

`demographic_cols`
A set of integers indicating the columns corresponding to demographic information, e.g. major, year of study, gender, etc. This argument is only used by the diversity-based assignment.

`skills` A numeric measure of overall skill level (higher means more skilled). This argument is only used by the diversity-based assignment. This argument can be set to NULL. If this is done, then the model used only maximises the diversity.

`pref_mat` The preference matrix with dimensions equal to the num of groups x B*T, where T is the number of topics and B is the number of sub-groups per topic. This argument is only used in the preference-based assignment. See the Details section for more information.

Details

For the diversity-based assignment, the demographic variables are converted into an NxN dissimilarity matrix. By default, the dissimilarity metric used is the Gower distance `cluster::daisy()`.

For the preference-based assignment, the preference matrix indicates the preference that each group has for the project topics. For this model, each topic has possibly B sub-groups. The number of columns of this matrix must be B*T. Suppose there are T=3 topics and B=2 sub-groups per topic. Then the order of the sub-topics should be:

T1S1, T2S1, T3S1, T1S2, T2S2, and T3S2.

Note that higher values in the preference matrix reflect a greater preference for a particular topic-subtopic combination, since the objective function is set to be maximised.

Value

For the diversity-based assignment model, this function returns a list containing:

- N: number of students
- G: number of self-formed groups
- m: a (student x groups) matrix, indicating group membership for each student.
- d: dissimilarity matrix, NxN
- s: skills vector for each individual student (possibly NULL)

For the preference-based assignment model, this function returns a list containing:

- N: number of students
- G: number of self-formed groups

- m: a (student x groups) matrix, indicating group membership for each student.
- n: a vector of length G, with the number of students in each self-formed group.
- p: The preference matrix from the input argument.

`pba_gc_ex002` *PBA Group Composition Data Example 002*

Description

An example dataset to use with the preference-based assignment model.

Usage

```
pba_gc_ex002
```

Format

`pba_gc_ex002`**:**

A data frame with 8 rows and 2 columns.

- id: the student id of each students, simply the integers 1 to 8.
- grouping: the self-formed groups submitted by each student. In this case, each self-formed group is of size 2.

Source

This dataset was constructed by hand.

`pba_prefmat_ex002` *PBA Group Preference Data Example 002*

Description

An example dataset to use with the preference-based assignment model.

Usage

```
pba_prefmat_ex002
```

Format

`pba_prefmat_ex002`**:**

A matrix with 4 rows and 4 columns

Each row represents the preferences of each self-formed group in the dataset `pba_gc_ex002`.

Source

This dataset was constructed by hand.

prepare_model *Initialise optimisation model*

Description

Initialise optimisation model

Usage

```
prepare_model(
  df_list,
  yaml_list,
  assignment = c("diversity", "preference"),
  w1 = 0.5,
  w2 = 0.5
)
```

Arguments

df_list	The output list from extract_student_info().
yaml_list	The output list from extract_params_yaml().
assignment	Character string indicating the type of model that this dataset is for. The argument is either 'preference' or 'diversity'. Partial matching is fine.
w1, w2	Numeric values between 0 and 1. Should sum to 1. These weights correspond to the importance given to the diversity- and skill-based portions in the objective function.

Value

An ompr model.

Index