Package 'golfr'

March 15, 2025

Type Package

Title Group Assignment Tool

Version 0.1.0

Description An efficient algorithm to generate group assignments for classroom settings while minimizing repeated pairings across multiple rounds.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Hannah Kim [aut],
Collin Nolte [cre]

Maintainer Collin Nolte <noltecollin@grinnell.edu>

Repository CRAN

Date/Publication 2025-03-15 17:10:01 UTC

Contents

`GenerateData` *Generates test data frame of the students with unique ID (uppercase letters)*

Description

Generates test data frame of the students with unique ID (uppercase letters)

Usage

```
GenerateData(num_students)
```

Arguments

`num_students`	a positive integer.

Value

a data frame

Examples

```
# five students
GenerateData(5)
```

`groupassign` *Assign Groups and Update Interaction Matrix*

Description

Assign Groups and Update Interaction Matrix

Usage

```
groupassign(student_data, students_per_group, iterations)
```

Arguments

`student_data`	A data frame containing student identifiers. The column should be named 'Student'.
`students_per_group`	An integer specifying the number of students per group.
`iterations`	An integer defining how many rounds of group assignments should be performed.

Value

A matrix representing the interaction history of students, where each cell indicates the number of times two students have been grouped together.

Examples

```
n_students <- 4
student_data <- GenerateData(n_students)
students_per_group <- 2
iterations <- 3

matrices_df <- groupassign(student_data, students_per_group, iterations)
matrices_df
```

---

initmat *Generates initial matrix*

Description

Generates initial matrix

Usage

```
initmat(students)
```

Arguments

students	a column of a dataset that has students' unique IDs

Value

matrix

Examples

```
# Initial matrix 5x5
data <- GenerateData(5)
initmat(data$Student)
```

MakeGroups *Assign every student into groups with set number of students per group*

Description

Assign every student into groups with set number of students per group, with set number of iterations (overlap not considered)

Usage

```
MakeGroups(data, students_per_group, iterations, initial_matrix)
```

Arguments

`data`	a data frame
`students_per_group`	a positive integer
`iterations`	a positive integer
`initial_matrix`	matrix generated with 'initmat()'

Value

data frame

Examples

```
# Assign 9 students into 3 groups of 3, with 3 iterations
data <- GenerateData(9)
M <- initmat(data$Student)
MakeGroups(data, 3, 3, M)
```

updatemat *Update Interaction Matrix with Group Assignments*

Description

This function updates an existing interaction matrix based on new group assignments. It increments the matrix values to track how often students have been grouped together.

Usage

```
updatemat(initialmat, group_assignments, students)
```

Arguments

initialmat	A square matrix where rows and columns represent students, and cell values track the number of times two students have been grouped together.
group_assignments	A vector indicating the group assignments for each student.
students	A vector containing student identifiers corresponding to the matrix row and column names.

Value

An updated matrix reflecting the new group assignments.

Examples

```
n_students <- 4
student_data <- GenerateData(n_students)
initial_matrix <- initmat(student_data$Student)
group_assignments <- c(1, 1, 2, 2)  # Example group assignments

updated_matrix <- updatemat(initial_matrix, group_assignments, student_data$Student)
updated_matrix
```

Index