Package 'GradeR'

December 17, 2018

Title Functions to make exams, plot grades, ...

Version 0.1.3

Description

Depends R (>= 3.5.1)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Author Jordan Adamson [aut, cre]

Maintainer Jordan Adamson <jordan.m.adamson@gmail.com>

Published 2018-12-17

Date 2018-12-17

URL <https://github.com/Jadamso/GradeR>

Imports stargazer,
RndTexExams

Suggests exams,
ProfessR

RoxygenNote 6.1.1

R topics documented:

`grade_plot` *Calculate Grades*

Description

Calculate Grades

Usage

```
grade_plot(pdfname, score, cuts, breaks = seq(50, 100, by = 1),
  ymin = NA, ymax = NA)
```

Arguments

`pdfname`	name of pdf file
`score`	matrix of class scores
`cuts`	cuttoff points for letter grades

`groups2student` *Randomly Assign Groups to Student Discussants*

Description

Randomly Assign Groups to Student Discussants

Usage

```
groups2student(g = 9, n = 6, w = 14, v = 40)
```

Arguments

`g`	number of groups
`n`	number of people per group
`w`	number of weeks
`v`	number of students

i.n.cases.correct.answers *i.n.cases.correct.answers*

Description

i.n.cases.correct.answers

Usage

```
i.n.cases.correct.answers(n.cases.correct.answers, q.answers, i.cases)
```

Arguments

```
i.answers
```

i.test.fun *i.test.fun*

Description

i.test.fun

Usage

```
i.test.fun(i.test, f.out, n.test, n.question, bank = TRUE, Qend = TRUE,
  latex.dir.out = "latexOut", do.randomize.questions = TRUE,
  do.randomize.answers = TRUE, list.in, l.def, verbose = FALSE)
```

Arguments

```
do.randomize.questions
```

```
verbose
```

latexout *latexout*

Description

latexout

Usage

```
latexout(my.tex.file, f.temp.tex, bank, exam.class,
  str.pattern.end.mchoice, my.last.part, qtext, Qend = TRUE)
```

Arguments

```
Qend=TRUE
```

letter_calc *Calculate Letter Grades*

Description

Calculate Letter Grades

Usage

```
letter_calc(scores, cuts, percent = FALSE)
```

Arguments

weight_df	number of groups
n	number of people per group
w	number of weeks
v	number of students

my.test.comment *helper commands to rte.analyze.tex.file to eliminate coded out lines*

Description

helper commands to rte.analyze.tex.file to eliminate coded out lines

Usage

```
my.test.comment(str.in)
```

Arguments

str.in	

Value

logical TRUE if line coded out

q.answers.cases.fun *q.answers.cases.fun*

Description

q.answers.cases.fun

Usage

```
q.answers.cases.fun(case.now, q.answers, str.pattern.correct,
  str.pattern.choice)
```

Arguments

```
str.pattern.choice
```

random.matrix *Randomly Assign Student Discussants*

Description

Randomly Assign Student Discussants

Randomly Assign Student Discussants w/o duplicates

Usage

```
random.matrix(p, v)

random.matrix.no.dup(p, v)
```

Arguments

p	number of times each person presents
v	number of students

rand_fun *Batch Latex Export*

Description

Batch Latex Export

Usage

```
rand_fun(lfile, latex.dir.in, latex.dir.out, n.test = 1,
  do.randomize.questions = TRUE, do.randomize.answers = TRUE)
```

Arguments

`do.randomize.answers`

rte.analyze.tex.file *import and analyze a latex file*

Description

import and analyze a latex file

Usage

```
rte.analyze.tex.file(f.in, bank = TRUE)
```

Arguments

`f.in`	character string for input file

Value

exam

rte.build.rdn.test *rte.build.rdn.test*

Description

rte.build.rdn.test

Usage

```
rte.build.rdn.test(list.in, f.out, n.test, n.question, bank = TRUE,
  latex.dir.out = "latexOut", do.randomize.questions = TRUE,
  do.randomize.answers = TRUE)
```

Arguments

do.randomize.answers=TRUE

score_calc *Calculate Grades*

Description

Calculate Grades

Usage

```
score_calc(weight_df, verbose = TRUE)
```

Arguments

weight_df	number of groups
verbose	print output

students2weeks.format *Formatting Table for Export*

Description

Formatting Table for Export

Usage

```
students2weeks.format(discussant_table, students, write_file = NA)
```

Arguments

discussant_table	table to format
students	matrix of students
write_file	write table to csv? default NA

Value

a vector

---

students2weeks.matrix *Randomly Assign Student Discussants to Weeks*

Description

Randomly Assign Student Discussants to Weeks

Randomly Assign Student Discussants to Weeks without duplicates

Usage

```
students2weeks.matrix(students, w = 14, p = NA, v = NA)

students2weeks.matrix.no.dup(students, w = 14, p = NA, v = NA)
```

Arguments

students	matrix of students
w	number of weeks
p	number of times each person presents
v	number of students

---

students2weeks.matrix2 *Randomly Assign Student Discussants*

Description

Randomly Assign Student Discussants

Randomly Assign Student Discussants

Usage

```
students2weeks.matrix2(v, w, p = NA, a = 2)

students2weeks.matrix2.no.dup(v, w, p = NA, a = 2)
```

Arguments

v	number of students
w	number of weeks
p	number of times each person presents
a	

Value

a vector

---

students2weeks.print *Write A Latex Table for Each Week*

Description

Write A Latex Table for Each Week

Usage

```
students2weeks.print(discussants, w_start = min(discussants, na.rm = T),
  w_end = max(discussants, na.rm = T), rdir)
```

Arguments

discussants	list/matrix of discussants
w_start, w_end	number of weeks, set w_start=2 if noone reads first week
rdir	directory to write discussants to

Value

a vector

Index