Package 'sistec'

October 14, 2022

Type Package

Title Tools to Analyze 'Sistec' Datasets

Version 0.2.0

Maintainer Samuel Macêdo <samuelmacedo@recife.ifpe.edu.br>

Description The Brazilian system for diploma registration and validation on technical and superior courses are managing by 'Sistec' platform, see <https://sistec.mec.gov.br/>. This package provides tools for Brazilian institutions to update the student's registration and make data analysis about their situation, retention and drop out.

License GPL (>= 2)

LazyData true

URL https://github.com/r-ifpe/sistec

BugReports https://github.com/r-ifpe/sistec/issues

Depends R (>= 3.6)

Imports dplyr (>= 1.0), openxlsx, rlang, shiny, stringi, stringr, tcltk, utils

RoxygenNote 7.1.0

Suggests testthat

Encoding UTF-8

NeedsCompilation no

Author Samuel Macêdo [aut, cre], Carlos Patrício [aut], Cássio Santos [aut], Clécio Santos [aut], Tiago Spiandorello [ctb]

Repository CRAN

Date/Publication 2020-10-26 22:20:02 UTC

R topics documented:

`aria` *Aria web application*

Description

This is the web application using the sistec package. It was created to ease the work using the package, but you can have the same results reading the files (sistec::read_rfept()), comparing the results with (sistec::compare_sistec()) and write the outputs (sistec::write_output()).

Usage

```
aria(
  output_path = NULL,
  output_folder_name = "ARIA",
  max_file_size = 100,
  options_port = 8888,
  options_launch_browser = TRUE,
  test_mode = TRUE,
  version = "offline"
)
```

Arguments

`output_path`	The folder where you want to save the results.
`output_folder_name`	The folder's name you want to save the results.
`max_file_size`	The maximum file size in megabytes.
`options_port`	The TCP port that the application should listen on, usually 8888.
`options_launch_browser`	If true, the system's default web browser will be launched automatically after the app is started.

test_mode	Use FALSE in production and TRUE when you are testing. In production, when you close the browser ,the app and the R session will be closed. In test, only the app will close when you close the browser.
version	A string. Choose "offline" or "online" version.

Value

A web application.

`aria_aws` *Aria web application*

Description

This is the web application using the sistec package. It was created to ease the work using the package, but you can have the same results reading the files (sistec::read_rfept()), comparing the results with (sistec::compare_sistec()) and write the outputs (sistec::write_output()).

Usage

```
aria_aws(
  output_path = NULL,
  output_folder_name = "ARIA",
  max_file_size = 100,
  test_mode = TRUE
)
```

Arguments

output_path	The folder where you want to save the results.
output_folder_name	The folder's name you want to save the results.
max_file_size	The maximum file size in megabytes.
test_mode	Use FALSE in production and TRUE when you are testing. In production, when you close the browser ,the app and the R session will be closed. In test, only the app will close when you close the browser.

Value

A web application.

`aria_desktop_build` *Build ARIA desktop version*

Description

Build the desktop version of ARIA automatically. You will need innosetup to build the executable. Only works in windows. See Details for more information.

Usage

```
aria_desktop_build(aria_folder = "", download_dependencies = TRUE)
```

Arguments

`aria_folder`	The folder you want to create the executable.
`download_dependencies`	Download all the dependencies: r-portable and packages needed to run the ARIA. Use FALSE if you already have the dependencies in the aria_folder.

Details

To create the executable correctly, follow theses steps:

- Build the sistec package (in Rstudio, you use ctrl + shift + b)
- Create a folder
- Use `aria_desktop_build("your folder")`
- Install innosetup (if you don't have it installed yet)
- Double click the .iss file created in your folder and run it.

Value

Create the setup to build ARIA desktop in your specified folder.

Examples

```
## Not run:
aria_folder <- tempdir()
aria_desktop_build(aria_folder)

## End(Not run)
```

compare_sistec *Comparison between Sistec and a student registration dataset*

Description

A generic function to compare and save the student situation. This function also shows inconsistencies in the datasets. You can pass the folder's file path or a data frame read by `sistec::read_*()` functions. In most cases, there are no link between courses in Sistec and the academic registration. You can pass this relationship using `linked_courses` parameter or using ARIA estimation.

Usage

```
compare_sistec(sistec, rfept, linked_courses = NULL)
```

Arguments

`sistec`	The folder's path to Sistec files or the Sistec data frame read by `sistec::read_sistec()`function.
`rfept`	The folder's path to students reagistration datasets or a data frame read by `sistec::read_*()`functions.
`linked_courses`	By default, the linked courses will be estimate using the data (ARIA estimation). You can specify those links loadind a .xlsx/csv file with linked courses between the rfept and sistec. The columns must be in this order: INICIO, CICLO, CURSO_SISTEC CURSO_RFEPT CAMPUS. The date in INICIO column must be in yyyy.period. Ex.: 2019.1 or 2019.2.

Value

A list of data frames.

Examples

```
# these datasets are not real. It is just for test purpose.

# using ARIA estimation for relation between courses
sistec <- read_sistec(system.file("extdata/examples/sistec",
                                  package = "sistec"))

qacademico <- read_qacademico(system.file("extdata/examples/qacademico",
                                          package = "sistec"))

compare_sistec(sistec, qacademico)

# using linked courses file

linked_courses <- read_linked_courses(system.file("extdata/examples/linked_courses",
                                                  package = "sistec"), "csv")

compare_sistec(sistec, qacademico, linked_courses)
```

read_linked_courses *Read linked courses files*

Description

This function reads a .xlsx/.csv file with the linked courses between Sistec and the academic registration. The columns must be in this order: INICIO, CICLO, CURSO_SISTEC, CURSO_RFEPT and CAMPUS. The date in INICIO column must be in yyyy.period. Ex.: 2019.1 or 2019.2. .

Usage

```
read_linked_courses(path = "", format = "xlsx", encoding = "latin1")
```

Arguments

path	The linked courses file's path.
format	You can choose between xlsx or csv.
encoding	This function uses latin1 by default but you can use UTF-8 if needed.

Value

A data frame.

Examples

```
linked_courses <- read_linked_courses(system.file("extdata/examples/linked_courses",
                                                  package = "sistec"), "csv")

linked_courses
```

read_qacademico *Read Qacademico files*

Description

This function support two kinds of schemas: from the api and the website. See Details if you need help to download the Qacademico data.

Usage

```
read_qacademico(path = "", start = NULL)
```

Arguments

path	The Qacademico file's path.
start	A character with the date to start the comparison. The default is the minimum value found in the data. The date has to be in this format: "yyyy.semester". Ex.: "2019.1" or "2019.2".

Details

To download the student's data, go to your proper account on Qacademico and follow:

- "Relatorio de Alunos" -> "Listagem de Alunos" (choose year and period)
- Click on "visualizar"
- Using F10 shortcut and save in .csv format
- Rename the including year and period (example2020_1.csv)

Be sure that your data has the variables: "Matricula", "Nome", "Situacao Matricula", "Curso", "Cpf", "Instituicao", "Per. Letivo Inicial" and "Cota".

Value

A data frame.

Examples

```
# this dataset is not a real one. It is just for test purpose.
qacademico <- read_qacademico(system.file("extdata/examples/qacademico",
                                          package = "sistec"))

qacademico
```

---

read_rfept *Identify and read academic registration*

Description

The `read_rfept()` is a wrapper around `read_qacademico()` and `read_sigaa()`. Now you just need to specify the folder path and `read_rfept()` identifies if it is a qacademico or sigaa file and then read it.

Usage

```
read_rfept(path = "", start = NULL)
```

Arguments

path	The file's path to Qacademico or Sigaa folder.
start	A character with the date to start the comparison. The default is the minimum value found in the data. The date has to be in this format: "yyyy.semester". Ex.: "2019.1" or "2019.2".

Details

By now, this function only supports qacademico and sigaa-sc.

Value

A data frame.

Examples

```
# these datasets are not a real ones. It is just for test purpose.

qacademico <- read_rfept(system.file("extdata/examples/qacademico", package = "sistec"))

sigaa <- read_rfept(system.file("extdata/examples/sigaa", package = "sistec"))

class(qacademico)
class(sigaa)

# example selecting the period
qacademico_2019_2 <- read_rfept(system.file("extdata/examples/qacademico", package = "sistec"),
                                start = "2019.2")

class(qacademico_2019_2)
```

---

read_sigaa *Read Sigaa files*

Description

This function reads partial and complete Sigaa datasets. These two formats can perform the comparison, but the partial doesn't have information about "Campus" and "Cota". See Details if you need help where to download the Sigaa data.

Usage

```
read_sigaa(path = "", start = NULL)
```

Arguments

path	The Sigaa file's path.
start	A character with the date to start the comparison. The default is the minimum value found in the data. The date has to be in this format: "yyyy.semester". Ex.: "2019.1" or "2019.2".

Details

To download the partial Sigaa's data, go to your proper account on Sigaa and follow:

- Access the panel "Consultas" inside Sigaa module.
- Generate the report "Consulta geral discentes".
- Select the check box "Trazer informações em forma de relatório" e "Gerar csv".
- Select the filter "Campus" and other filters you desire.
- Click on "Buscar" and download the file.

Be sure that your data has the variables: "Matricula", "Nome", "Status, "Curso" and "CPF".

For the complete dataset, you have to download directly from the Sigaa database. Be sure that your data has the variables: "Matricula", "Nome", "Situacao Matricula", "Curso", "Cpf", "Instituicao", "ano_ingresso", "semestre_ingresso" and "Cota".

Value

A data frame.

Examples

```
# this dataset is not a real one. It is just for test purpose.
sigaa <- read_sigaa(system.file("extdata/examples/sigaa",
                                package = "sistec"))

sigaa
```

`read_sistec` *Read sistec files*

Description

The package provides support if your data comes from setec or web. You just need to pass the folder's path were are your files. See Details if you need help to download the data from Sistec.

Usage

```
read_sistec(path = "", start = NULL)
```

Arguments

`path`	The sistec file's path.
`start`	A character with the date to start the comparison. The default is the minimum value found in the data. The date has to be in this format: "yyyy.semester". Ex.: "2019.1" or "2019.2".

Details

You can download the Sistec's student registration using your proper account on Sistec. Be sure that your data has these variables:

- On setec: "Nome Aluno", "Numero Cpf", "Co Ciclo Matricula", "Situacao Matricula", "No Curso", "Dt Data Inicio" and "Unidade Ensino".
- On web: "NO_ALUNO", "NU_CPF", "CO_CICLO_MATRICULA", "NO_STATUS_MATRICULA", "NO_CICLO_MATRICULA", "DT_DATA_INICIO" and "CO_UNIDADE_ENSINO".

Tip: To take every student for your institution/campus using web, search by student name and use " ".

Value

A data frame.

Examples

```
# this dataset is not a real one. It is just for test purpose.
sistec <- read_sistec(system.file("extdata/examples/sistec",
                                  package = "sistec"))

sistec

# example selecting the period
sistec_2019_2 <- read_sistec(system.file("extdata/examples/sistec", package = "sistec"),
                                 start = "2019.2")

sistec_2019_2
```

---

sistec_app *Sistec web application*

Description

This is the web application using the sistec package. It was created to ease the work using the package, but you can have the same results reading the files (sistec::read_*()), comparing the results (sistec::compare_sistec()) and write the outputs (sistec::write_output())

Usage

```
sistec_app(
  output_path = NULL,
  output_folder_name = "Comparações",
  max_file_size = 100,
  options_port = 8888,
  options_launch_browser = TRUE,
  test_mode = TRUE
)
```

Arguments

output_path	The folder where you want to save the results.
output_folder_name	The folder's name you want to save the results.
max_file_size	The maximum file size in megabytes.
options_port	The TCP port that the application should listen on, usually 8888.
options_launch_browser	If true, the system's default web browser will be launched automatically after the app is started.
test_mode	Use FALSE in production and TRUE when you are testing. In production, when you close the browser ,the app and the R session will be closed. In test, only the app will close when you close the browser.

Value

A web application.

---

write_output *Save the comparison results*

Description

You can use this function to save the results separated by campus. The results will be saved in .xlsx format.

Usage

```
write_output(x, output_path = NULL, output_folder_name = "ARIA")
```

Arguments

x	A list returned by `compare_sistec()`.
output_path	The folder where you want to save the results.
output_folder_name	The folder's name you want to save the results.

Value

None.

Index