Package 'sketchy'

July 23, 2025

Type Package

Title Create Custom Research Compendiums

Version 1.0.5

Maintainer Marcelo Araya-Salas <marcelo.araya@ucr.ac.cr>

Description Provides functions to create and manage research compendiums for data analysis. Research compendiums are a standard and intuitive folder structure for organizing the digital materials of a research project, which can significantly improve reproducibility. The package offers several compendium structure options that fit different research project as well as the ability of duplicating the folder structure of existing projects or implementing custom structures. It also simplifies the use of version control.

License GPL (>= 2)

Imports knitr, stringi, crayon, packrat, utils, git2r, xaringanExtra, rmarkdown, remotes, cli, urlchecker, stringr

Depends R (>= 3.5.0)

LazyData TRUE

URL <https://github.com/maRce10/sketchy>

BugReports <https://github.com/maRce10/sketchy/issues>

NeedsCompilation no

Suggests testthat, formatR

RoxygenNote 7.3.2

Repository CRAN

Language en-US

Author Marcelo Araya-Salas [aut, cre] (ORCID:
<https://orcid.org/0000-0003-3594-619X>),
Andrea Yure Arriaga Madrigal [aut]

Date/Publication 2025-01-16 20:40:07 UTC

Contents

---

add_to_gitignore   *Add entries to gitignore*

Description

add_to_gitignore adds entries to gitignore based on file extension or file size

Usage

```
add_to_gitignore(add.to.gitignore = FALSE, cutoff = NULL, extension = NULL, path = ".")
```

Arguments

add.to.gitignore

Logical to control if files are added to 'gitignore' or just printed on the console.

cutoff   Numeric. Defines the file size (in MB) cutoff used to find files (i.e. only files above the threshold would returned). 99 (MB) is recommended when hosting projects at github as the current file size limit is 100 MB.

extension   Character string to define the file extension of the files to be searched for.

path   Path to the project directory. Default is current directory.

Details

The function can be used to avoid conflicts when working with large files or just avoid adding non-binary files to remote repositories. It mostly aims to simplify spotting/excluding large files. Note that file names can be manually added to the '.gitignore' file using a text editor.

Value

Prints the name of the files matching the searching parameters. If add.to.ignore = TRUE the files matching the search parameters ('cutoff' and/or 'extension') would be added 'gitignore' (a file used by git to exclude files form version control, including adding them to github).

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

See Also

[compendiums](), [make_compendium]()

Examples

```
{
data(compendiums)

make_compendium(name = "my_compendium", path = tempdir(),
 format = "basic", force = TRUE)

# save a file
write.csv(iris, file.path(tempdir(), "my_compendium", "iris.csv"))

# add the file to gitignore
add_to_gitignore(add.to.gitignore = TRUE,
path = file.path(tempdir(), "my_compendium"), extension = "csv")
}
```

---

check_urls   *Check urls in dynamic report files*

Description

check_urls Check urls in dynamic report files (.md, .Rmd & .qmd)

Usage

```
check_urls(path = ".")
```

Arguments

path   Path to the directory containing the files to be checked. Default is current directory.

Details

The function can be used to check if url addresses in dynamic reports are broken. Taken from Nan Xiao's blogpost (https://nanx.me/blog/post/rmarkdown-quarto-link-checker/).

Value

A url_checker_db object with an added class with a custom print method.

Author(s)

Nan Xiao (<me@nanx.me>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3. Xiao, N. (2023). A General-Purpose Link Checker for R Markdown and Quarto Projects. Blog post. https://nanx.me/blog/post/rmarkdown-quarto-link-checker/

See Also

add_to_gitignore, make_compendium

Examples

```
{
data(compendiums)

# make compendiums
make_compendium(name = "my_compendium", path = tempdir(),
format = "basic", force = TRUE)

# check urls in scripts
check_urls(path = file.path(tempdir(), "./scripts"))
}
```

---

compendiums   *List with compendium skeletons*

Description

compendiums is a list containing the format of 14 different project folder skeletons. For each format 3 elements are provided: '$skeleton' (folder structure), '$comments' and '$info' (reference to the original source).

Usage

```
data(compendiums)
```

Format

A list with 14 compendium formats:

basic basic sketchy format

figures similar to basic, but including output/figures folders

project_template following Kenton White's ProjectTemplate

pakillo following Francisco Rodriguez-Sanchez' template

boettiger following Carl Boettiger's blog

wilson following Wilson et al. (2017) format

small_compendium following Marwick et al (2018) small compendium format

medium_compendium following Marwick et al (2018) medium compendium format

large_compendium following Marwick et al (2018) large compendium format

vertical following Vuorre *et al.* (2018) R package vertical

rrtools following Marwick (2018) (R package rrtools)

rdir following folder structure described on at a r-dir blog post (although seems like it was removed)

workflowr following Blischak *et al.* (2019) R package workflowr

sketchy same skeleton than 'basic' but including a custom Rmarkdown and quarto files for documenting data analyses

References

Blischak, J. D., Carbonetto, P., & Stephens, M. 2019. *Creating and sharing reproducible research code the workflowr way*. F1000Research, 8.

Marwick, B. 2018. *rrtools: Creates a reproducible research compendium*.

Marwick, B., Boettiger, C., & Mullen, L. 2018. *Packaging data analytical work reproducibly using R (and friends)*. The American Statistician, 72(1), 80-88.

Vuorre, Matti, and Matthew J. C. Crump. 2020. *Sharing and Organizing Research Products as R Packages*. PsyArXiv. January 15.

Wilson G, Bryan J, Cranston K, Kitzes J, Nederbragt L. & Teal, T. K.. 2017. *Good enough practices in scientific computing*. PLOS Computational Biology 13(6): e1005510.

---

load_packages   *Install and load packages*

Description

`load_packages` installs and loads packages from different repositories.

Usage

```
load_packages(packages, quite = FALSE, upgrade.deps = FALSE)
```

Arguments

packages   Character vector with the names of the packages to be installed. The vector names indicate the repositories from which packages will be installed. If no name is included CRAN will be used as the default repository. Available repositories are: 'cran', 'github', 'gitlab', 'bitbucket' and 'bioconductor'. Note that for 'github', 'gitlab' and 'bitbucket' the string must include the user name in the form 'user/package'.

quite	Logical argument to control if package startup messages are printed. Default is FALSE (messages are printed).
upgrade.deps	Logical argument to control if package dependencies are upgraded.Default is FALSE.

Details

The function installs and loads packages from different repositories in a single call.

Value

No object is returned.

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

See Also

[compendiums](#), [make_compendium](#)

Examples

```
## Not run:
load_packages(packages = c("kableExtra", bioconductor = "ggtree",
github = "maRce10/Rraven"), quite = TRUE)

## End(Not run)
```

---

make_compendium   *Generate folder structures for research compendiums*

Description

make_compendium generates the folder structure of a research compendium.

Usage

```
make_compendium(name = "research_compendium", path = ".", force = FALSE,
format = "basic", packrat = FALSE,
git = FALSE, clone = NULL, readme = TRUE, Rproj = FALSE)
```

Arguments

name	character string: the research compendium directory name. No special characters should be used. Default is "research_compendium".
path	Path to put the project directory in. Default is current directory.
force	Logical controlling whether existing folders with the same name are used for setting the folder structure. The function will never overwrite existing files or folders.
format	A character vector of length 1 with the name of the built-in compendiums available in the example object 'compendiums' (see [compendiums](#) for available formats). Default is 'basic'. Alternatively, it can be a character vector with 2 or more elements with the names of the folders and subfolders to be included.
packrat	Logical to control if packrat is initialized (packrat::init()) when creating the compendium. Default is FALSE.
git	Logical to control if a git repository is initialized (git2r::init()) when creating the compendium. Default is FALSE.
clone	Path to a directory containing a folder structure to be cloned. Default is NULL. If provided 'format' is ignored. Folders starting with ^\.git|^\.Rproj.user|^\.\.Rcheck will be ignored.
readme	Logical. Controls if a readme file (in Rmd format) is added to the project. The file has predefined fields for documenting objectives and current status of the project. Default is TRUE.
Rproj	Logical. If TRUE a R project is created (i.e. a .Rproj file is saved in the main project directory).

Details

The function takes predefined folder structures to generate the directory skeleton of a research compendium.

Value

A folder skeleton for a research compendium. In addition the structure of the compendium is printed in the console. If the compendium format includes a "manuscript" or "doc(s)" folder the function saves a manuscript template in Rmarkdown format ("manuscript.Rmd"), a BibTex file ("example_library.bib", for showing how to add citations) and APA citation style file ("apa.csl") inside that folder.

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

Marwick, B., Boettiger, C., & Mullen, L. (2018). Packaging Data Analytical Work Reproducibly Using R (and Friends). American Statistician, 72(1), 80-88.

Alston, J., & Rick, J. (2020). A Beginners Guide to Conducting Reproducible Research.

See Also

[compendiums](#), [print_skeleton](#)

Examples

```
{
data(compendiums)

make_compendium(name = "mycompendium", path = tempdir(), format = "basic",
force = TRUE)
}
```

---

open_wd   *Open working directory*

Description

open_wd opens the working directory in the default file browser.

Usage

```
open_wd(path = ".", verbose = TRUE)
```

Arguments

path	Directory path to be opened. By default it's the working directory.
verbose	Logical to control whether the 'path' is printed in the console. Default is TRUE.

Details

The function opens the working directory using the default file browser and prints the working directory in the R console. This function aims to simplify the manipulation of files and folders in a project.

Value

Opens the working directory using the default file browser.

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

See Also

spot_unused_files

Examples

```
{
open_wd()
}
```

---

print_skeleton   *Print folder structures*

Description

print_skeleton prints the folder structure of a research compendium.

Usage

```
print_skeleton(path = ".", comments = NULL, folders = NULL)
```

Arguments

path	path to the directory to be printed. Default is current directory.
comments	A character string with the comments to be added to each folder in the graphical representation of the folder skeleton printed on the console.
folders	A character vector including the name of the sub-directories of the project.

Details

The function prints the folder structure of an existing project.

Value

The folder skeleton is printed in the console.

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

See Also

compendiums, make_compendium

Examples

```
{
data(compendiums)

make_compendium(name = "my_other_compendium", path = tempdir(), format = "basic")

print_skeleton(path = file.path(tempdir(), "mycompendium"))
}
```

---

spot_unused_files   *Spot/remove unused files*

Description

spot_unused_files

Usage

```
spot_unused_files(
  path = ".",
 file.extensions = c("png", "jpg", "jpeg", "gif", "bmp", "tiff", "tif", "csv", "xls",
    "xlsx", "txt"),
  script.extensions = c("R", "Rmd", "qmd"),
  archive = FALSE,
  ignore.folder = "./docs"
)
```

Arguments

path   A character string with the path to the directory to be analyzed. Default is current directory.

file.extensions
                     A character vector with the file extensions to be considered. Default is c("png", "jpg", "jpeg", "gif", "bmp", "tiff", "tif", "csv", "xls", "xlsx", "txt").

script.extensions
                     A character vector with the script extensions to be considered. Default is c("R", "Rmd", "qmd").

| archive | A logical value indicating whether to archive the unused files. If TRUE the spotted files will be move into the folder "./unused_files". Default is FALSE. |
| ignore.folder | A character string with the path or paths to the directory(ies) to be ignored. Default is "./docs". |

Details

This function is used to spot/remove unused files in a project directory. It is useful to keep the project directory clean and organized. It is recommended to first run the function with a the argument archive = FALSE to spot which files are being spotted and then run archive = TRUE if they need to be removed.

Value

Returns a data frame with 2 columns: file.name (self explanatory) and folder (where the file is found).

Author(s)

Marcelo Araya-Salas (<marcelo.araya@ucr.ac.cr>)

References

Araya-Salas, M., Arriaga, A. (2023), sketchy: research compendiums for data analysis in R. R package version 1.0.3.

See Also

add_to_gitignore, make_compendium

Examples

```
{
}
```

Index