

# Package ‘easystats’

July 22, 2024

**Type** Package

**Title** Framework for Easy Statistical Modeling, Visualization, and Reporting

**Version** 0.7.3

**Maintainer** Daniel Lüdecke <d.luedecke@uke.de>

**Description** A meta-package that installs and loads a set of packages from 'easystats' ecosystem in a single step. This collection of packages provide a unifying and consistent framework for statistical modeling, visualization, and reporting. Additionally, it provides articles targeted at instructors for teaching 'easystats', and a dashboard targeted at new R users for easily conducting statistical analysis by accessing summary results, model fit indices, and visualizations with minimal programming.

**License** MIT + file LICENSE

**URL** <https://easystats.github.io/easystats/>

**BugReports** <https://github.com/easystats/easystats/issues>

**Depends** R (>= 3.6)

**Imports** bayestestR, correlation, datawizard, effectsize, insight, modelbased, parameters, performance, report, see, tools, utils

**Suggests** DHARMA, DT, flexdashboard (>= 0.6.2), glmmTMB, htr, jsonlite, knitr, mockery, pak, patchwork, rmarkdown, testthat (>= 3.2.1), withr, xml2

**VignetteBuilder** knitr

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.3.2

**Config/testthat/edition** 3

**Config/Needs/website** easystats/easystatstemplate

**NeedsCompilation** no

**Author** Daniel Lüdecke [aut, cre] (<<https://orcid.org/0000-0002-8895-3206>>, @strengjacke),  
 Dominique Makowski [aut] (<<https://orcid.org/0000-0001-5375-9967>>, @Dom\_Makowski),  
 Mattan S. Ben-Shachar [aut] (<<https://orcid.org/0000-0002-4287-4801>>),  
 Indrajeet Patil [aut] (<<https://orcid.org/0000-0003-1995-6531>>, @patilindrajeets),  
 Brenton M. Wiernik [aut] (<<https://orcid.org/0000-0001-9560-6336>>, @bmwiernik),  
 Etienne Bacher [aut] (<<https://orcid.org/0000-0002-9271-5075>>),  
 Rémi Thériault [aut] (<<https://orcid.org/0000-0003-4315-6788>>, @rempsysc)

**Repository** CRAN

**Date/Publication** 2024-07-22 10:00:01 UTC

## Contents

easystats_packages . . . . .	2
easystats_update . . . . .	3
easystats_zen . . . . .	3
install_latest . . . . .	4
install_suggested . . . . .	5
model_dashboard . . . . .	6
<b>Index</b>	<b>9</b>

---

easystats\_packages      *List all packages in the easystats ecosystem*

---

## Description

List all packages in the easystats ecosystem

## Usage

```
easystats_packages()
```

## Value

A character vector

## Examples

```
easystats_packages()
```

---

easystats_update	<i>Update easystats-packages and its dependencies from CRAN, if necessary.</i>
------------------	--

---

**Description**

Update *easystats*-packages and its dependencies from CRAN, if necessary.

**Usage**

```
easystats_update(which = c("all", "core", "deps"))
```

**Arguments**

which	String, indicates whether easystats-packages (which = "core"), dependencies (which = "deps") or both (which = "all") should be checked for available updates.
-------	---

**Value**

Invisible NULL.

**Examples**

```
# check which local easystats-packages (and their dependencies)
# are out of date and install updates from CRAN
easystats_update()

# update only easystats core-packages
easystats_update("core")
```

---

easystats_zen	<i>Welcome to the easyverse</i>
---------------	---------------------------------

---

**Description**

Welcome to the easyverse

**Usage**

```
easystats_zen()
```

**Value**

A reassuring message.

**Examples**

```
easystats_zen()
```

---

install_latest	<i>Install the easystats suite from R-universe (GitHub) or CRAN</i>
----------------	---

---

**Description**

This function can be used to install all the **easystats** packages, either latest development versions (from R-universe/GitHub) or the current versions from CRAN. If the development versions are installed, packages will be installed from the stable branch (master/main) for each package.

**Usage**

```
install_latest(
  source = c("development", "cran"),
  packages = "all",
  force = FALSE,
  verbose = TRUE
)
```

**Arguments**

source	Character. Either "development" or "cran". If "cran", packages will be installed from the default CRAN mirror returned by <code>getOption("repos")['CRAN']</code> . If "development" (the default), packages are installed from the r-universe repository ( <a href="https://easystats.r-universe.dev/">https://easystats.r-universe.dev/</a> ).
packages	Character vector, indicating which packages to be installed. By default, the option "all" will install all <b>easystats</b> packages.
force	Logical, if FALSE, only those packages with a newer version number will be installed. Use force=TRUE to force installation of all packages, even if the version number for the locally installed packages is identical to the latest development-versions. Only applies when source="development".
verbose	Toggle messages.

**Value**

Invisible NULL.

**Examples**

```
# install latest development-version of easystats packages from
# the r-universe repository, but only those packages that have newer
# versions available
install_latest()
```

```
# install all latest development-version of easystats packages from
# the r-universe repository, no matter whether local installations
# are up to date or not.
install_latest(force = TRUE)
```

---

install\_suggested      *Download all suggested packages*

---

## Description

In `easystats`, we have a 0-dependency policy, which makes our packages fairly light and fast to install. However, we rely on many many (many) packages for testing (at least all the packages for functions that we support) and some specific features. These "soft dependencies" can be downloaded at once using this function. This will allow you to fully utilize all of `easystats`' functionalities without errors.

## Usage

```
install_suggested(package = "easystats")

show_suggested(package = "easystats")

show_reverse_dependencies(package = "easystats")
```

## Arguments

package	If NULL or "easystats" (the default), all suggested packages for all 'easystats' packages will be installed. If specific packages are specified, only suggested packages for those packages will be installed.
---------	--

## Details

To reduce the dependency load, 'easystats' packages by default will not download all internally needed packages. It will ask the user to download them only if they are needed. The current function can help install all packages a given 'easystats' package might need. For example, `install_suggested("see")`. `show_suggested()` is a convenient helper to show the current list of suggested packages for each 'easystats' package.

## Value

Useful only for its side-effect of installing the needed packages.

**Examples**

```
# download all suggested packages
if (FALSE) {
  install_suggested("easystats")
}

# listing all reverse dependencies of easystats packages
show_reverse_dependencies()

# listing all soft/weak dependencies of easystats packages
show_suggested()
```

---

model_dashboard	<i>Generate a regression model summary from easystats</i>
-----------------	---

---

**Description**

A dashboard containing the following details for the entered regression model:

- tabular summary of parameter estimates
- dot-and-whisker plot for parameter estimates
- tabular summary of indices for the quality of model fit
- collection of models for checking model assumptions
- text report
- model information table

**Usage**

```
model_dashboard(
  model,
  check_model_args = NULL,
  parameters_args = NULL,
  performance_args = NULL,
  output_file = "easydashboard.html",
  output_dir = getwd(),
  rmd_dir = system.file("templates/easydashboard.Rmd", package = "easystats"),
  quiet = FALSE,
  browse_html = interactive()
)
```

**Arguments**

model            A regression model object.

check_model_args	A list of named arguments that are passed down to <code>performance::check_model()</code> . For further documentation and details about the arguments, see <a href="#">this website</a> . See also 'Examples'.
parameters_args	A list of named arguments that are passed down to <code>parameters::model_parameters()</code> . For further documentation and details about the arguments, see <a href="#">this website</a> . See also 'Examples'.
performance_args	A list of named arguments that are passed down to <code>performance::model_performance()</code> . For further documentation and details about the arguments, see <a href="#">this website</a> . See also 'Examples'.
output_file	A string specifying the file name in <code>rmarkdown::render()</code> . Default is "easydashboard.html".
output_dir	A string specifying the path to the output directory for report in <code>rmarkdown::render()</code> . Default is to use the working directory.
rmd_dir	A string specifying the path to the directory containing the RMarkdown template file. By default, package uses the template shipped with the package installation ( <code>inst/templates/easydashboard.Rmd</code> ).
quiet	An option to suppress printing during rendering from knitr, pandoc command line and others. To only suppress printing of the last "Output created: " message, you can set <code>rmarkdown.render.message</code> to FALSE
browse_html	A logical deciding if the rendered HTML should be opened in the browser. Defaults to <code>interactive()</code> .

**Value**

An HTML dashboard.

**Troubleshooting**

For models with many observations, or for more complex models in general, generating the model assumptions plot might become very slow. One reason is that the underlying graphic engine becomes slow for plotting many data points. In such cases, setting the argument `check_model_args = list(show_dots = FALSE)` might help. Furthermore, look at other arguments of `?performance::check_model`, which can be set using `check_model_args`, to increase performance (in particular the `check`-argument can help, to skip some unnecessary checks).

**Examples**

```
# define a regression model
mod <- lm(wt ~ mpg, mtcars)

# with default options
model_dashboard(mod)

# customizing 'parameters' output: standardize coefficients
model_dashboard(mod, parameters_args = list(standardize = "refit"))
```

```
# customizing 'performance' output: only show selected performance metrics
model_dashboard(mod, performance_args = list(metrics = c("AIC", "RMSE")))

# customizing output of model assumptions plot: don't show dots (faster plot)
model_dashboard(mod, check_model_args = list(show_dots = FALSE))
```

# Index

`easystats_packages`, 2  
`easystats_update`, 3  
`easystats_zen`, 3

`install_latest`, 4  
`install_suggested`, 5  
`interactive()`, 7

`model_dashboard`, 6

`parameters::model_parameters()`, 7  
`performance::check_model()`, 7  
`performance::model_performance()`, 7

`rmarkdown::render()`, 7

`show_reverse_dependencies`  
  (`install_suggested`), 5  
`show_suggested` (`install_suggested`), 5