

# Package: rjdmarkdown (via r-universe)

October 13, 2026

**Type** Package

**Title** 'rmarkdown' Extension for Formatted 'RJDemetra' Outputs

**Version** 0.2.3

**Description** Functions to have nice 'rmarkdown' outputs of the seasonal and trading day adjustment models made with 'RJDemetra'.

**SystemRequirements** Java ( $\geq 8$ )

**Depends** R ( $\geq 3.1.1$ ), RJDemetra

**Imports** knitr, kableExtra, magrittr

**License** EUPL

**URL** <https://github.com/AQLT/rjdmarkdown>

**BugReports** <https://github.com/AQLT/rjdmarkdown/issues>

**Encoding** UTF-8

**VignetteBuilder** knitr

**Suggests** rmarkdown, ggdemetra

**Config/roxygen2/version** 8.1.0

**Config/pak/sysreqs** cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev make libharfbuzz-dev default-jdk libicu-dev libpng-dev libuv1-dev libxml2-dev

**Repository** <https://aqlt.r-universe.dev>

**Date/Publication** 2026-10-13 19:49:39 UTC

**RemoteUrl** <https://github.com/AQLT/rjdmarkdown>

**RemoteRef** HEAD

**RemoteSha** 95e3901542abcf4edcef978ecdece3a46352be53

## Contents

|                     |   |
|---------------------|---|
| create_rmd          | 2 |
| print_decomposition | 5 |
| print_diagnostics   | 6 |
| print_preprocessing | 7 |

|       |   |
|-------|---|
| Index | 9 |
|-------|---|

---

create\_rmd

---

*Create and render 'rmarkdown' file*


---

## Description

Function to create a 'rmarkdown' file with all the output and render it

## Usage

```
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
  ...
)

## S3 method for class 'SA'
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
  ...
)

## S3 method for class 'jSA'
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
```

```

    ...
)

## S3 method for class 'workspace'
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
  ...
)

## S3 method for class 'multiprocessing'
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
  ...,
  workspace
)

## S3 method for class 'sa_item'
create_rmd(
  x,
  output_file,
  output_format = "pdf_document",
  preprocessing_fun = print_preprocessing,
  decomposition_fun = print_decomposition,
  diagnostics_fun = print_diagnostics,
  title = "Seasonal adjustment summary",
  knitr_chunk_opts = list(fig.pos = "h", echo = FALSE, results = "asis", fig.cap =
    "S-I Ratio"),
  ...,
  workspace
)

```

## Arguments

|                                |                                                                                                                                                                  |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x</code>                 | the object to render: it can be a "SA", "jSA", "sa_item", "multiprocessing" or "workspace" object                                                                |
| <code>output_file</code>       | the name of the output 'rmarkdown' file.                                                                                                                         |
| <code>output_format</code>     | the R Markdown output format to convert to: "pdf_document" for a pdf output, "html_document" for a HTML output. See <a href="#">render</a> for more details.     |
| <code>preprocessing_fun</code> | the function used to print the preprocessing. <a href="#">print_preprocessing</a> by default. If <code>preprocessing_fun = NULL</code> the function is not used. |
| <code>decomposition_fun</code> | the function used to print the decomposition <a href="#">print_decomposition</a> by default. If <code>decomposition_fun = NULL</code> the function is not used.  |
| <code>diagnostics_fun</code>   | the function used to print the diagnostics <a href="#">print_diagnostics</a> by default. If <code>diagnostics_fun = NULL</code> the function is not used.        |
| <code>title</code>             | the title of the R Markdown document.                                                                                                                            |
| <code>knitr_chunk_opts</code>  | options for R code chunks. See <a href="#">opts_chunk</a> for more details.                                                                                      |
| <code>...</code>               | other arguments to pass to <a href="#">render</a> .                                                                                                              |
| <code>workspace</code>         | the workspace. Only used when <code>x</code> is a "sa_item" or "multiprocessing".                                                                                |

## Examples

```

ipi <- RJDemetra::ipi_c_eu[, "FR"]
jsa_x13 <- RJDemetra::jx13(ipi)

output_file <- tempfile(fileext = ".Rmd")
create_rmd(jsa_x13, output_file, output_format = "pdf_document")
# To directly open the pdf:
browseURL(sub(".Rmd", ".pdf", output_file, fixed = TRUE))

# To create a pdf from a workspace:
jsa_ts <- jtramoseats(ipi)
wk <- new_workspace()
mp <- new_multiprocessing(wk, "sa1")
add_sa_item(wk, "sa1", jsa_x13, "X13")
add_sa_item(wk, "sa1", jsa_ts, "TramoSeats")

# It's important to compute the workspace to be able
# to import the models
compute(wk)

output_file <- tempfile(fileext = ".Rmd")
create_rmd(wk, output_file,
  output_format = c("pdf_document", "html_document"),
  output_options = list(toc = TRUE,
    number_sections = TRUE))

```

```
# To open the file:
browseURL(sub(".Rmd",".pdf", output_file, fixed = TRUE))
browseURL(sub(".Rmd",".html", output_file, fixed = TRUE))
```

---

print\_decomposition     *Print the decomposition*

---

## Description

Function to print the decomposition model

## Usage

```
print_decomposition(
  x,
  format = knitr::opts_knit$get("rmarkdown.pandoc.to"),
  plot = TRUE,
  digits = 3,
  decimal.mark = getOption("OutDec"),
  booktabs = TRUE,
  ...
)
```

## Arguments

|              |                                                                                                          |
|--------------|----------------------------------------------------------------------------------------------------------|
| x            | the object to print.                                                                                     |
| format       | output format: "latex" or "html".                                                                        |
| plot         | boolean indicating whether to plot or not the S-I Ratio.                                                 |
| digits       | number of digits after the decimal point.                                                                |
| decimal.mark | the character to be used to indicate the numeric decimal point.                                          |
| booktabs     | boolean indicating whether to use or not the booktabs package (when format = "latex").                   |
| ...          | arguments passed to <a href="#">plot.decomposition_X11</a> or <a href="#">plot.decomposition_SEATS</a> . |

## Examples

```
ipi <- RJDemetra::ipi_c_eu[, "FR"]

jsa_x13 <- RJDemetra::jx13(ipi)
print_decomposition(jsa_x13, format = "latex")

sa_ts <- RJDemetra::jtramoseats(ipi)
print_decomposition(sa_ts, format = "html")
```

---

|                   |                              |
|-------------------|------------------------------|
| print_diagnostics | <i>Print the diagnostics</i> |
|-------------------|------------------------------|

---

## Description

Function to print diagnostics tests on the quality of the pre-processing and the decomposition

## Usage

```
print_diagnostics(
  x,
  format = knitr::opts_knit$get("rmarkdown.pandoc.to"),
  signif.stars = TRUE,
  tests = c("mean", "skewness", "kurtosis", "ljung box",
    "ljung box (residuals at seasonal lags)", "ljung box (squared residuals)",
    "qs test on sa", "qs test on i", "f-test on sa (seasonal dummies)",
    "f-test on i (seasonal dummies)", "Residual seasonality (entire series)",
    "Residual seasonality (last 3 years)", "f-test on sa (td)", "f-test on i (td)"),
  digits = 3,
  decimal.mark = getOption("OutDec"),
  booktabs = TRUE,
  ...
)
```

## Arguments

|              |                                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| x            | the object to print.                                                                                                                      |
| format       | output format: "latex" or "html".                                                                                                         |
| signif.stars | logical; if TRUE, p-values are additionally encoded visually as 'significance stars' in order to help scanning of long coefficient tables |
| tests        | characters containing the names of the tests to print.                                                                                    |
| digits       | number of digits after the decimal point.                                                                                                 |
| decimal.mark | the character to be used to indicate the numeric decimal point.                                                                           |
| booktabs     | boolean indicating whether to use or not the booktabs package (when format = "latex").                                                    |
| ...          | unused arguments.                                                                                                                         |

## Examples

```
ipi <- RJDemetra::ipi_c_eu[, "FR"]

jsa_x13 <- RJDemetra::jx13(ipi)
print_diagnostics(jsa_x13, format = "latex")
```

```
sa_ts <- RJDemetra::tramoseats(ipi)
print_diagnostics(sa_ts, format = "html")
```

---

print\_preprocessing     *Print the pre-processing model*

---

## Description

Function to print the pre-processing model

## Usage

```
print_preprocessing(
  x,
  format = knitr::opts_knit$get("rmarkdown.pandoc.to"),
  signif.stars = TRUE,
  digits = 3,
  decimal.mark = getOption("OutDec"),
  booktabs = TRUE,
  summary = TRUE,
  likelihood = TRUE,
  arima = TRUE,
  regression = TRUE,
  ...
)
```

## Arguments

|              |                                                                                                                                           |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| x            | the object to print.                                                                                                                      |
| format       | output format: "latex" or "html".                                                                                                         |
| signif.stars | logical; if TRUE, p-values are additionally encoded visually as ‘significance stars’ in order to help scanning of long coefficient tables |
| digits       | number of digits after the decimal point.                                                                                                 |
| decimal.mark | the character to be used to indicate the numeric decimal point.                                                                           |
| booktabs     | boolean indicating whether to use or not the booktabs package (when format = "latex").                                                    |
| summary      | boolean indicating whether to use or not the summary section.                                                                             |
| likelihood   | boolean indicating whether to use or not the likelihood section.                                                                          |
| arima        | boolean indicating whether to use or not the arima section.                                                                               |
| regression   | boolean indicating whether to use or not the regression section.                                                                          |
| ...          | unused.                                                                                                                                   |

**Examples**

```
ipi <- RJDemetra::ipi_c_eu[, "FR"]

sa_x13 <- RJDemetra::jx13(ipi)
print_preprocessing(sa_x13, format = "latex")

sa_ts <- RJDemetra::tramoseats(ipi)
print_preprocessing(sa_ts, format = "html")
```

# Index

`create_rmd`, [2](#)

`opts_chunk`, [4](#)

`plot.decomposition_SEATS`, [5](#)

`plot.decomposition_X11`, [5](#)

`print_decomposition`, [4](#), [5](#)

`print_diagnostics`, [4](#), [6](#)

`print_preprocessing`, [4](#), [7](#)

`render`, [4](#)