

How to submit your article

Articles for the R Journal are short to medium length, no more than 20 pages, on topics of interest to the R community. Articles must not be published or submitted for publication elsewhere. The submission needs to include:

- paper files to build the article including the .Rmd, .tex, .bib, .sty, figures, and the output files .html, .pdf.
- files to reproduce the results presented in the paper, including R scripts, and data.
- motivating letter that describes why your paper is suitable for the consideration by the R Journal.
- if needed, supplementary files that might include additional technical details or examples.

Read the information below on the requirements for files and format and then [CLICK HERE](#) to submit an article, re-submission, or revision.

We are also keen to welcome new reviewers for the R Journal. If you would like to volunteer to review articles, please complete this [form here](#). You should have at least completed an undergraduate degree, and be an active R user.

Creating your article

Rmarkdown to output html and pdf

The [rjtools](#) package has a template to create the files needed for a submission in the form required by the R Journal.

It is strongly recommended that you use the `create_article()` function. Knitting the Rmd will generate both an HTML and PDF format of your paper.

When writing your article you should aim to write without specific reference to HTML and LaTeX codes. If you do need to write code specific to the HTML or PDF output, you can use `knitr:::is_html_output()` and `knitr:::is_latex_output()` to conditionally produce a particular output.

The web articles for the R Journal are based on the distill framework, which provides several styling and arrangement options for your article contents. The [distill R package documentation](#) (generally also applicable to articles written for the R Journal) describes various article elements and Rmarkdown syntax: <https://rstudio.github.io/distill/>.

We are especially looking for creative uses of interactive content in HTML formatted articles. This will require that you make different code chunks evaluated conditionally using `eval=knitr:::is_html_output()` or `eval=knitr:::is_latex_output()` with appropriate re-captionsing, and conditional inline reference links. Interactive plots should be produced with a manageable file size (aim for under 10Mb) which may require small data examples. Figures and tabs for the html format should have alt-text in the chunk specification, to further assist with accessibility.

Converting from `rticles::rjournal_article` RMarkdown

If you currently use the `rticles::rjournal_article` output format for R Markdown, the change involves five steps: (1) change output formats to `rjtools:::rjournal_web_article` to switch to the new style, (2) remove any latex specific functionality, (3) change figure, table references to `\@ref()` and (4) add the `rjournal.cs1` and `rjournal.css` template files to your folder, updating YAML to utilise them, and (5) set the reference to the `.bib` in the YAML.

Traditional latex format

Articles can also be written using the traditional latex template. Using this style you will need to provide `RJwrapper.tex`, `RJwrapper.pdf`, `RJournal.sty`, `your-article.tex` (which contains the content of your paper), `your-article.bib`, `your-article.R` (to reproduce results in your paper), in addition to the motivation letter. You should avoid using any other specialist latex style files. The `rjtools` check functions described below can also be applied to check your files prior to submission.

Checking your article

The `rjtools` package has a number of functions which can help you check that your article is ready to submit. These include:

- `initial_check_article()` does multiple checks on the files in your paper submission folder
 - `check_wrappers()` that the files `RJournal.tex` and `RJournal.pdf` are present
 - `check_filenames()` file names are consistent and as requested
 - `check_unnecessary_files()` that there are no extra files than those necessary and requested
 - `check_cover_letter()` there is a cover letter with the files to be submitted
 - `check_title()` that the title is in title case
 - `check_section()` that section headings are in sentence case
 - `check_spelling()` does a spell-check
 - `check_proposed_pkg()` package is available on CRAN or bioconductor
 - `check_packages_available()` additionally referenced packages are also available on CRAN or Bioconductor

- Each of these can also be run individually.

File format for submission

New submission

To submit a new article to the R Journal, you will need to complete. You will need to provide these details:

- Your name.
- An email address for the corresponding author. Try to ensure it is an address that will be available in the long term, through the review process and after the paper is published so that readers can contact you about your work if desired.
- An alternate email address, in case email to the primary address doesn't respond to editors requests.
- Names of all authors.
- Article title.
- Article type.
- Article keywords/tags. You should be prepared to select at least one or two from the list and you are free to add an additional words or phrases that accurately describe your paper.
- **Zip file containing:**
 - finished paper files .pdf and .html.
 - paper files to build the article including the .Rmd, .tex, .bib, .sty, figures.
 - file to reproduce the results, including R scripts, and data.
 - motivating letter that describes why your paper is suitable for the consideration by the R Journal.
 - reproducibility (supplementary) files in order that results reported in your paper can be reproduced, plus any additional files that contain additional technical details for your work or extra examples.
- **A list of the reproducibility and supplementary files**, exactly as they are placed in your paper directory, to be entered comma-delimited like mypaper.Rmd, data/mydata.rds, scripts/mycode.R, appendix.pdf. This list is used to automatically build the .zip file that is distributed along with your paper as supplementary materials, when it is published. Thus it is important that you provide the comma-delimited file names to streamline the build of your paper. If you have a lot of data and code files you so that listing them all is prohibitive, you can zip them into one file to list. Note that, figure files are not considered supplementary materials for the R Journal.

You may have problems submitting if size of your submission zip file is more than 10Mb. If you have large files, for example, data files, you can make them available on an alternative site and provide the links in the paper. If you have problems, please contact the Editor-in-Chief at r-journal@r-project.org.

Your paper will be checked for reproducibility by running the code provided. Please ensure that the examples can be run in a timely manner, perhaps by reducing the example data size. If necessary, intermediate output files, can be provided to shorten the run time.

What happens after you submit a paper

Once you have submitted a paper, the editorial board will go through the following steps.

1. Decide if your submission is technically in the correct format. In particular:
 - are all the requested files there and of the correct format?
 - does the reproducibility code run?
 - is all code used open source and available via CRAN or BioConductor?
 - are all the data files (if any) in an open, non-proprietary format?
2. If the technical test is passed, it is decided whether the submission is amenable for peer review:
 - is the topic within scope?
 - is the paper properly organized?
 - is the writing style suitable for review?
3. If the submission passes the second test, it will be sent out for review. This process might take several months. You will receive one or more reviews, possibly with notes from one of the (associate) editors, and a *verdict*. There are four possible verdicts:
 - **Accepted** Your paper will be published in the R Journal. You will be asked to upload a final version that will be proof-read and you will receive a print proof before publication.
 - **Minor revisions** The paper can be accepted conditional on a number of smaller revisions in the manuscript and/or the code. You are expected to fix these issues within three months, and resubmit along with a letter where you describe, point-by-point, how the comments of the reviewers have been addressed. The paper will not be reviewed again by peer reviewers, but the editor will check whether the points have been sufficiently addressed.
 - **Major revisions** The manuscript and/or the code need severe updates before the manuscript can be accepted for publication. You are expected to fix the issues within three months, and resubmit along with a letter where you describe, point-by-point how the comments of the reviewers have been addressed. The paper will be sent back to the reviewers for a second round of review, which may result in any verdict.
 - **Reject** Your paper is found to be unsuited for publication in the R Journal. No resubmission is expected.



Re-submission and revisions to accepted articles are received through the same site where new submissions are received (link below). When providing a re-submission or revision, you will need to **enter the identifier of the original submission** in the last field of the submission form. This also needs to be done if your paper has previously been rejected, and you have made appropriate corrections for it to be considered again.

Note on special issue articles

A special issue article can be considered to be a refereed journal article and generally carries more weight than an unrefereed conference proceedings.

Conference committees will designate a “Conference Editor in Chief” (CEiC) who will be primarily responsible for putting the special issue together. For this pilot to be successful, it is important that putting together the special issue does not significantly increase the workload of the regular editors or the administrative team that helps to build the issues. It will be the responsibility of the CEiC to identify associate editors if needed, assign reviewers, accept and reject papers, and assemble the special issue. It will also be the responsibility of the CEiC to make sure that the issue reflects the diversity of the community, and that the R Consortium and R Community Code of Conduct is followed during the process.

