labeleR: an R package to optimize the generation of collection labels and scientific documents

Julia G. de Aledo [1,2,3] Jimena Mateo-Martín [3] Francisco Rodríguez-Sánchez [4] Ignacio Ramos-Gutiérrez [3,4]

[1] Estación Biológica de Doñana
[2] Universidad Rey Juan Carlos
[3] Universidad Autónoma de Madrid
[4]
Universidad de Sevilla

Key Points:

- R
- Rmarkdown
- herbaria

Corresponding author: Ignacio Ramos-Gutiérrez, `i.ramos.gutierrez@gmail.com`

Abstract

`labeleR` is an R package designed to automate the creation of collection labels and documents for scientific events. It simplifies repetitive and time-consuming tasks, offering a practical alternative to manual or costly tools. With `labeleR`, users can generate a wide variety of customizable PDF documents that can also be automatically emailed. The package provides a set of functions classified into two groups: scientific collections (e.g. labels for herbarium or insects) and scientific events organization (e.g. personal badges, abstract books and certificates of attendance and participation). Starting from a tidy dataset, users can easily customize content, incorporate QR codes, logos, images, and edit their own templates. `labeleR` transforms tedious and repetitive workflows into an efficient, reproducible process, contributing to greater scientific productivity. The package is available under an open-source license and can be freely downloaded from CRAN or the GitHub repository (https://ecologyr.github.io/labeleR/).

Plain language summary

Some journals require a plain language summary. See: https://publications.agu.org/author-resource-center/text-requirements/#abstract

1 Cover letter

`Concise cover letter focused on the question the manuscript attempts to address`

2 Statement of need

The management and design of scientific labels and event documents is a time-consuming task. Large-scale label generation tools for herbarium and scientific collections (used by institutions such as museums or botanical gardens) are often paid and proprietary software (e.g. **? ?**). Microsoft Excel-Word integration through mailing lists is commonly used at a smaller scale, although still involving paid software with limited large database management capacity. Most free alternatives are not open-source, require installing a program with limited customization, and are often only compatible with Windows operating system (e.g. **? ?**), or designed for very specific purposes (e.g. **?** for insects, **?** for lichens or **?** for plant vouchers). Additionally, credentials and certificates for scientific events are either created manually one at a time, through paid online servers, or by hiring an event organization company. To our knowledge, there are no free, customizable tools for the bulk production and distribution of these documents. `labeleR` fills this gap facilitating the creation of scientific collection labels, conference badges, attendance and participation certificates, and abstract books, among others.

3 Package description

The `labeleR` package builds upon the `RMarkdown` ecosystem (**?**) to generate PDF documents from a tidy data frame in R (Figure 1). `labeleR` functions include three types of arguments: (1) R instructions, such as the data object, paths and file name of the rendered document; (2) “fixed” arguments, text that remains constant across output documents (e.g. event name or image path); (3) “variable” arguments, linked to columns in the dataframe, thus changing between documents (e.g. taxonomic names in labels or attendee names in certificates). A QR code can be included either through a fixed argument or a variable argument, without the need for external software. Users can also edit and adapt the default `RMarkdown` templates provided by the package for their own purposes.

4 Documents that can be generated with

4.1 Labels for collections

Appropriate labelling of samples is a fundamental step of the scientific process (i.e., labelling test tubes in laboratories, storing animal or plant materials or displaying collections in museums or botanical gardens). A user-friendly bulk rendering tool is vital for efficiently producing crafted, uniform labels in a reproducible manner. We present three label types: “herbarium” (most complex), “collection” (most aesthetic) and “tinylabels” (compact and simplified, for small insect collections) (Figure 2). These labels can include QR codes (e.g. links to websites, images, or identification codes) without additional tools, making it easy to quickly access and link to external information.

4.1.1 Herbarium labels

Herbarium labels are one of the documents with more variable parameters. Note that the `family.column` content will always be capitalized, and the `taxon.column` one in italics, recommended to be used as originally defined, while the rest ca be interchangeable. The QR can stand for a free text (and therefore remain identical in all labels), or be a column name, and the codes will be rendered with the individual information of each row. Four different labels will fit in each of the A4 pdf pages.

```
create_herbarium_label(
  data = herbarium.table,
  path = "labeleR_output",
  filename = "herbarium_labels",
  qr = "QR_code",
  title ="Magical flora of the British Isles" ,
  subtitle = "Project: Eliminating plant blindness in Hogwarts students",
  family.column = "Family",
  taxon.column = "Taxon",
  author.column = "Author",
  det.column = "det",
  date.det.column = "Det_date",
  location.column = "Location",
  area.description.column = "Area_description",
  latitude.column = "Latitude",
  longitude.column = "Longitude",
  elevation.column = "Elevation",
  field1.column = "life_form",
  field2.column = "Observations",
  field3.column = "Height",
  collector.column = "Collector",
  collection.column = "Collection_number",
  assistants.column = "Assistants",
  date.column = "Date"
)
```

4.1.2 Collection labels

They count with five variable parameters, which are not recommended to be too long, along with the possibility of including a QR code (fixed or variable) and an image (logo or picture). Field 1 will be always capitalized, and Field 2 italicized. Any field can be left blank. The user may manually fix the backgroud and text colors to their preference, using HTML color codes. Eight different labels will fit in each of the A4 pdf pages.

```
create_collection_label(
  data = collection.table,
  path = "labeleR_output",
  filename = "labels",
  qr = "QR_code",
  field1.column = "field1",
  field2.column = "field2",
  field3.column = "field3",
  field4.column = "field6",
  field5.column = "field7",
  system.file("rmarkdown/pictures/Hogwarts_BnW.png", package = "labeleR"),
  bgcolor = "D0ECC1",  #White is "FFFFFF",
  textcolor = "1E3F20" #Black is "000000"
)
```

4.1.3 Tiny labels

This type of labels is a simplified version of the collection label, including just five variable fields and the possibility of including a QR code. It is recommended to write short texts in the variable arguments and in the QR, as they might become difficult to read. 16 different labels will fit in each of the A4 pdf pages.

```
create_tiny_label(
  data = tiny.table,
  qr = "QR_code",
  path = "labeleR_output",
  filename = "tinylabels",
  field1.column ="field2",
  field2.column ="field1",
  field3.column ="field3",
  field4.column ="field4",
  field5.column ="field5"
)
```

4.2 Documents for scientific events

Scientific events often host a high number of participants, and require the creation of different documentation, such as abstract books, personal identification badges and certificates for attendees and participants. Bulk rendering significantly decreases the amount of time invested in the creation of these documents. Moreover, to deliver attendance and participation certificates automatically, those `labeleR` functions allow users to automatically send individual documents to email addresses stored in a column.

4.2.1 Abstract book

Abstract books result in a single pdf document with multiple pages. Each abstract will appear on a different page, following the same order as in the dataframe rows. If other order of appearance is desired, it is necessary to first arrange the columns in the original dataframe. Each page will include four variable fields (title, author names, affiliations and the abstract texts). The output document can include a table of contents with the titles and page numbers of all abstracts. Additionally, is possible to insert a custom front page that appearing at the beginning of the document.

```
create_abstractbook(
data=abstract.table,
path = "labeleR_output",
filename = "congress_abstractbook",
title.column = "abstract_title",
authors.column = "authors",
affiliation.column = "affiliation",
text.column = "abstract_text",
title.cex = 20,
authors.cex = 15,
affiliations.cex = 14,
text.cex = 12,
frontpage = "Congress_frontpage.pdf"
)
```

4.2.2 Badges

Badges can be used for personal accreditation in congresses, courses, meetings, etc. They have only two variable fields (name and affiliation), and can include two top logos or images. Accreditation badges include a dot line in the bottom for individual hand-edition once printed.

```
create_badge(
  data = badges.table,
  path = "labeleR_output",
  filename = "badges",
  event = "INTERNATIONAL CONFERENCE OF MUGGLEOLOGY",
  name.column = "List",
  affiliation.column = "Affiliation",
  rpic = system.file("rmarkdown/pictures/Hogwartslogo.png", package = "labeleR"),
  lpic = system.file("rmarkdown/pictures/MinMagic.png", package = "labeleR")
)
```

4.2.3 Attendance certificates

Attendance certificates the only variable parameter is the name of the attendees. It allows to include a signature as an image, implying that the signer does not have to sign them individually. This certificate is available both in English and Spanish.

```
create_attendance_certificate(
  data = attendance.table,
  path = "labeleR_output",
  filename = "attendance_certificates",
  language = "English" ,
  name.column = "Names",
  type = "class",
  title = "Potions (year 1992-1993)",
  date = "23/06/1993",
  hours = "200",
  freetext = "taught by Professor S. Snape",
  signer = "A.P.W.B. Dumbledore",
  signer.role = "School Headmaster",
  rpic = system.file("rmarkdown/pictures/Hogwartslogo.png", package = "labeleR"),
  lpic = system.file("rmarkdown/pictures/Hogwartslogo.png", package = "labeleR"),
  signature.pic = system.file("rmarkdown/pictures/dumbledore.png", package = "labeleR")
)
```

4.2.4 Participation certificates

Participation certificates include multiple variable parameters (such as speaker, affiliation, title, etc.). These documents can be rendered in English and in Spanish.

```
create_participation_certificate(
  data = participation.table,
  path = "labeleR_output",
  filename = "participation_certificates",
  language = "English",
  name.column = "Name",
  affiliation.column = "House",
  comm.type.column = "Comm.type",
  title.column = "Title",
  date.column = "Date",
  type = "online",
  event = "seminar",
  freetext = "organized by Hogwarts School of Magic and Wizardry",
  signer = "A.P.W.B. Dumbledore",
  signer.role = "School Headmaster",
  rpic = system.file("rmarkdown/pictures/Hogwartslogo.png", package = "labeleR"),
  lpic = system.file("rmarkdown/pictures/MinMagic.png", package = "labeleR"),
  signature.pic = system.file("rmarkdown/pictures/dumbledore.png", package = "labeleR")
)
```

5 Further applications

The `labeleR` philosophy is quite simple: creating multiple documents with a common design from a dataset containing the required information. It offers a modular structure that allows for customization and extension for new applications. For instance, the newly added `create_multichoice` function generates multichoice tests randomizing the order of questions and possible answers from a given table (question bank). New developments will happen in the GitHub repository (https://github.com/EcologyR/labeleR) and eventually pushed to CRAN. User feedback and code contributions are welcome in the same repository to keep `labeleR` as an open and dynamic tool.

6 Figure legends

7 Data Accessibilty Statement

8 Competing Interests Statement

9 Author Contributions section

10 Acknowledgements

This package has been developed collaboratively between Sevilla and Madrid, with continuous feedback from colleagues in both locations. We acknowledge their input, support and collaboration. We are especially grateful to Manuel Molina for his ideas at 'labeleR initial stages. We would like to emphasize that the original idea has been built horizontally among early career researchers. Our work has been possible thanks to the support of the institutions and projects that employ us, and especially by the software-developing workshop (Cádiz, La Muela, 2023) organized by AEET and funded by US-1381388 grant from Universidad de Sevilla/Junta de Andalucía/FEDER-UE.

J.G.A. was supported by Next Generation EU Investigo contract (URJC-AI-17) and ANTENNA Biodiversa+ and European Commission (PCI2023-146022-2) postdoctoral contract. J.M.M. was supported by the Comunidad de Madrid and Universidad Autónoma de Madrid doctoral grant PEJ-2020-AI/AMB-17551 and research assistant contract PIPF-2022/ECO-24251. F.R.-S. was supported by VI PPIT-US and grants US-1381388 from Universidad de Sevilla/Junta de Andalucía/FEDER-UE and CNS2022-135839 funded by MICIU/AEI/10.13039/501100011033 and by European Union NextGenerationEU/PRTR. I.R.G. was supported by a doctoral grant at Universidad Autónoma de Madrid and a postdoctoral position at Universidad de Sevilla (CNS2022-135839).