Friedrich-Alexander-Universität
Medizinische Fakultät

FAU

Junior R-core experiences

Sebastian Meyer

Institute of Medical Informatics, Biometry and Epidemiology

useR! 2022 – International R User Conference

Who am I?

- (Bio-)Statistician, trained at LMU Munich (D), ENSAI Rennes (F), and UZH Zurich (CH) – PhD programme in Epidemiology and Biostatistics
- Since 2016: postdoc at FAU in Erlangen, Germany
- Since 2007 (BSc thesis) doing research in statistical methods for infectious disease `surveillance` data

With regard to base R:

- Learned in 2005 (R 2.1.0)
- 2011: first bug report submitted to R's Bugzilla (a suggestion to improve the Sweave documentation)
- Since January 2020: reviewing/discussing bugs, providing patches
- Also: someone who hadn't planned to join R-core, but then ...

Subect: Sebastian Meyer joins the R core team
From: Martin Mächler
Date: 2021-10-14
To: R-announce@R-project.org

We are very happy to announce that

Sebastian Meyer (http://www.imbe.med.uni-erlangen.de/ma/S.Meyer ;
Twitter @bastistician;
also https://github.com/bastistician/)

has joined the R core team yesterday (Oct 13). He has been an
active contributor notably in handling and fixing R bugzilla issues in
dozens of contributions and also by direct communication,
mainly during the last 2 years but starting considerably earlier.

The R Core team

Acknowledgements

- My top 3 mentors from the R Core Team:
  Kurt Hornik, Tomas Kalibera, Martin Mächler
- Director of IMBE @ FAU:
  Olaf Gefeller
- My 2 kids:
  for the fun of building a simple “collecting” game in R

1. NEWS in R 4.2.0

2. Contributing to Base R

3. Selected Bug Stories

Find the NEWS

- `news()` opens the NEWS from the installed R version in the browser
- https://www.R-project.org/ > What's New? > NEWS (of latest release)
- Revived news summary in the *R Journal*:
  https://journal.R-project.org/ > News and Notes > Changes in R

NEWS AND NOTES 734

Changes in R

by Tomas Kalibera, Sebastian Meyer, Kurt Hornik, Gennadiy Starostin and Luke Tierney

Abstract We present important changes in the development version of R (referred to as R-devel, to become R 4.2) and give a summary of the new search engine interfaced by `RSiteSearch()`. Some statistics on bug tracking activities in 2021 are also provided.

R-devel selected changes

- https://blog.R-project.org/ for blog posts on new core developments

Native UTF-8 support on Windows

(see blog posts by Tomas Kalibera)

- R on (recent) Windows now uses UTF-8 as the native encoding.
- Work with characters not representable in the system codepage.
- New R toolchain bundle for Windows, Rtools42, targeting the new Universal C Runtime (UCRT).
- Default user library now at `C:\Users\username\AppData\Local`
- More encoding-related fixes on Windows (e.g., for Rgui) are in R 4.2.1 (released *today*!), and Windows users of R 4.2.0 are advised to update

Enhancements to HTML Documentation

(see blog post by Deepayan Sarkar and Kurt Hornik)

- HTML5 (validated by `R CMD check --as-cran` via HTML Tidy)
- LaTeX-like math typesetting via KaTeX/MathJax
- Code highlighting via Prism
- Running examples directly from the help page

```
help("melanoma", "lattice", help_type = "html")
```

R 4.1.3

Examples

```
# Time-series plot.  Figure 3.64 from Cleveland.
xyplot(incidence ~ year,
  data = melanoma,
       aspect = "xy",
       panel = function(x, y)
               panel.xyplot(x, y, type="o", pch = 16),
       ylim = c(0, 6),
       xlab = "Year",
       ylab = "Incidence")
```

R 4.2.0

Examples

Run examples

```
# Time-series plot.  Figure 3.64 from Cleveland.
xyplot(incidence ~ year,
  data = melanoma,
    aspect = "xy",
    panel = function(x, y)
        panel.xyplot(x, y, type="o", pch = 16),
    ylim = c(0, 6),
    xlab = "Year",
    ylab = "Incidence")
```

Protection against malformed conditions I

`badcond <- c(TRUE, FALSE)`

Conditions of length > 1 produced a warning since R 1.7.0 (April 2003):

R 4.1.3

```
if (badcond) getRversion()
```

```
[1] '4.1.3'
Warning message:
In if (badcond) getRversion() :
  the condition has length > 1 and only the first element will be used
```

This has now become an error in R 4.2.0 (April 2022):

R 4.2.0

```
if (badcond) getRversion()
```

```
Error in if (badcond) getRversion() : the condition has length > 1
```

Protection against malformed conditions II

`badcond <- c(TRUE, FALSE)`

Logical operators `&&` and `||` silently ignored elements beyond the first:

R 4.1.3

```
badcond && TRUE
```

```
[1] TRUE
```

This now gives a warning but will also become an error in the future:

R 4.2.0

```
badcond && TRUE
```

```
[1] TRUE
Warning message:
In badcond && TRUE : 'length(x) = 2 > 1' in coercion to 'logical(1)'
```

Elementwise comparison is performed by `&` and `|`.

`simplify2array()`

Background

- Used by `sapply()` and `mapply()`
- Simplifies a list of **length-one** values *to a vector*:

```
month.name
```

```
 [1] "January"   "February"  "March"     "April"     "May"
 [6] "June"      "July"      "August"    "September" "October"
[11] "November"  "December"
```

```
sapply(month.name, nchar) |> summary()
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.000   4.750   6.500   6.167   8.000   9.000
```

- Does *not* simplify a list of **length-zero** values.

`sapply()` simplifies to an array *except* for common lengths 0 and 1.

`simplify2array(x, except = NULL)`

New argument

- Default `except = c(0L, 1L)` corresponds to `sapply` behaviour.
- Use `except = NULL` to always simplify to array (if possible):

```
lapply(month.name, nchar) |> simplify2array()
```

```
 [1] 7 8 5 5 3 4 4 6 9 7 8 8
```

```
lapply(month.name, nchar) |> simplify2array(except = NULL)
```

```
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]    7    8    5    5    3    4    4    6    9     7     8     8
```

```
list(integer(0)) |> simplify2array()
```

```
[[1]]
integer(0)
```

```
list(integer(0)) |> simplify2array(except = NULL)
```

```
     [,1]
```

Sys.setLanguage()

- Change the language of messages during an R session.
- Useful for internet searches.

```
months[1]
```

```
Fehler in months[1] : Objekt des Typs 'closure' ist nicht indizierbar
```

```
Sys.setLanguage("en") -> .olang
months[1]
```

```
Error in months[1] : object of type 'closure' is not subsettable
```

```
Sys.setLanguage(.olang)
```

More classed errors

R 4.1.3

```
tryCatch(months[1], error = identity)
```

```
<simpleError in months[1]: object of type 'closure' is not subsettable>
```

R 4.2.0

```
tryCatch(months[1], error = identity)
```

```
<notSubsettableError in months[1]: object of type 'closure' is not subset
```

This allows more specific handling of particular types of errors:

```
globalCallingHandlers(notSubsettableError = function (e)
    cat("OH NO, NOT AGAIN!\n"))
months[1]
```

```
OH NO, NOT AGAIN!
Error in months[1] : object of type 'closure' is not subsettable
```

Other changes

- New graphics engine features (see talk and blog post by Paul Murrell)
- New function `hashtab()` to create mutable hash tables that efficiently map keys to values. Keys can be simple strings but also other objects.
- `matrix(x, n, m)` now warns *more* when `length(x) > n * m`:

```
m <- matrix(1:2, 1, 1)
```

```
Warning message:
In matrix(1:2, 1, 1) :
data length differs from size of matrix: [2 != 1 x 1]
```

- Native pipe gains placeholder symbol (for a single named argument):

```
mtcars |> lm(mpg ~ disp, data = _) -> fit
```

- Relaunch of the search engine behind `RSiteSearch()`: https://search.R-project.org

1. NEWS in R 4.2.0

2. Contributing to Base R

3. Selected Bug Stories

How to get involved in base R development

- Subscribe to the R-devel mailing list (and read the posting guide): https://www.R-project.org/mail.html
- Report or analyze bugs on R's Bugzilla, https://bugs.R-project.org/, after reading the posting guide: https://www.R-project.org/bugs.html
- Good starting point: blog post "R Can Use Your Help: Reviewing Bug Reports" (by Tomas Kalibera and Luke Tierney, October 2019)
- That blog post was my motivation to kick off.
- Analyzing/Fixing bugs is the best place to start as you are faced with concrete problems to solve and parts of the code base to study.

Insights from the useR! 2020 panel discussion

Recording available at: https://youtu.be/X_eDHNVceCU

- There is no formal procedure to join the R Core Team.
- Make periodic contributions to the base R code until R-core finds that the contributor no longer needs mentoring.
- Distinguishing feature of R-core: write access to the code base
- Most of the time is spent by studying the existing code, analysing bug reports, and debugging breakage in packages.
- “R-core people do what they want to do and feel necessary to do. Important thing is they are qualified and do serious work, at least periodically.”
- It is OK not be an expert in low-level programming (look at me), but R-core does need some that are.

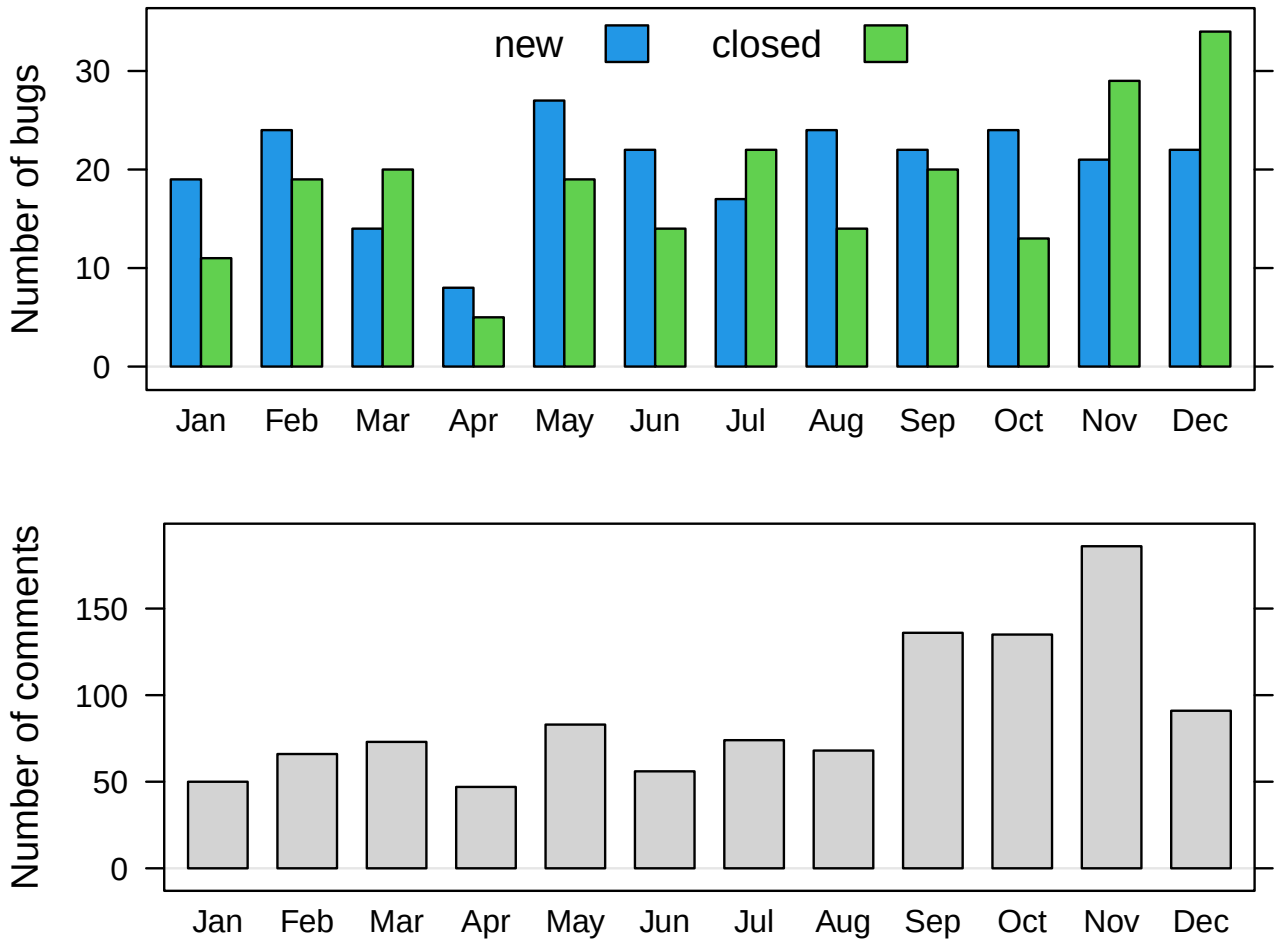
R Bugzilla statistics for 2021

- 244 new bug reports or enhancement requests
- Top 5 categories: Low-level, Language, Documentation, Misc, Wishlist
- 214 bugs closed (14% on weekends)
- 1065 comments on reports (17% on weekends)
- 115 contributors

Special thanks to: Elin Waring, Suharto Anggono, Michael Chirico, Henrik Bengtsson, Lionel Henry, Kevin Ushey, Andrew Simmons, and Duncan Murdoch

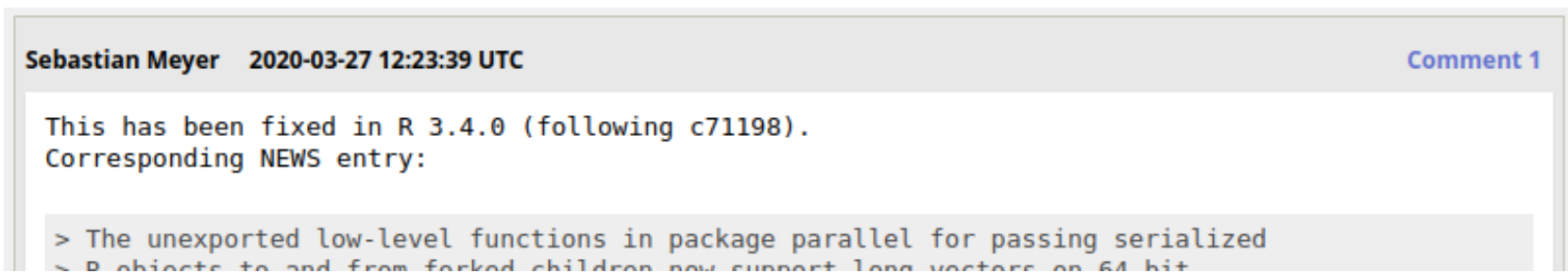
Will we beat the number of 214 closed bugs this year? We are at 97.

Monthly R Bugzilla activity in 2021



Plenty of fixed bugs not formally closed

- Reviewing open bugs showed many old ones that no longer occur
- Exercise: find out as of which R version the bug is gone
  - Scan the R NEWS for a related entry
  - Find the relevant part in R's code base and use `svn annotate` (`vc-annotate` in Emacs) or one of the mirrors on GitHub (wch/r-source, r-devel/r-svn) for browsing
  - Install older versions of R and test when the bug disappeared
- Inform R-core, for example (from Bug 16040):



Sebastian Meyer **2020-03-27 12:23:39 UTC** Comment 1

```
This has been fixed in R 3.4.0 (following c71198).
Corresponding NEWS entry:

> The unexported low-level functions in package parallel for passing serialized
> R objects to and from forked children now support long vectors on 64-bit
```

Bug 16100 - Error in logLik.nls

History

- Verbose report in December 2014, with severity “enhancement”
- Still reproducible in 2020, no comments in Bugzilla so far

Sebastian Meyer **2020-03-27 16:56:39 UTC** Comment 1

```
I can confirm that logLik.nls is wrong for weighted least squares fits and it seems to have been
wrong all the time since R 2.3.0, where support for weights in nls() was added.

The error can be seen by fitting a simple linear model with nls() and comparing the results with
lm(). Below is an example using the cars dataset with random weights. A patch follows.
```

- Essentially: missing division by N and doubly weighted RSS term
- Needed careful code *and* stats review (thus time); it helps to provide patches and test cases; sometimes a reminder (with more discussion)

Sebastian Meyer **2020-06-29 14:06:09 UTC** Comment 3

```
Created attachment 2642 [details]
Fix logLik.nls() calculation for weighted fits, including for zero weights

Bumping this as I think this bug is more than just an "enhancement" for a statistics-focussed
programming language. The result of logLik.nls is wrong for weighted fits. Today I recognized
```

Bug 16100 - Error in logLik.nls

Outcomes


```
                    Changes in version 4.0.3

BUG FIXES

    o   Fix incorrect calculation in logLik.nls() PR#16100, patch from Sebastian Meyer.
```

Subject: R: NLS log likelihood
Date: 2021-12-14
[. . .] I previously published a paper where the main analysis compared the BICs of a series of non-linear least squares models, which I ran in R. When someone else tried to replicate my analysis, they produced different results, and I discovered that the calculation of the log likelihood (and therefore BIC) of NLS models had changed since I ran that analysis [. . .]

Bug 17880 - `nlme::lme()` silently ignores offset

(nlme is currently maintained by R-core)

- `lme()` never supported `offset()` terms and never claimed it would.
- However, silently ignoring them is dangerous, because users will assume offsets work when they don't.
- The following was `TRUE` in nlme $\leq$ 3.1-155

```
library(nlme)
all.equal(coef(lme(weight ~ Diet + offset(Time), BodyWeight)),
          coef(lme(weight ~ Diet,                BodyWeight)))
```

but now gives

```
Error in lme.formula(weight ~ Diet + offset(Time), BodyWeight) :
  offset() terms are not supported
```

"a collaborator send me some code that ran glmmPQL with an offset term. This worked for my collaborator with R 4.1.2 and MASS 7.3-55 and nlme 3.1-155. I have R 4.2.0, MASS 7.3.57, nlme 3.1-157. But I get [. . .]"

R Can Use Your Help

1. Testing before release: bugs are found by using R in ways that aren't yet or cannot be covered by automated tests.
2. Reviewing bug reports: identify outdated or invalid reports, add a minimal reproducible example if one is missing.

Get started tomorrow at the *Bug BBQ*:

https://contributor.r-project.org/events/bug-bbq

Thanks to all contributors!