User Manual for R package RefManageR

Package Version 0.8.1. *http://cran.r-project.org/web/packages/RefManageR/*

Straightforward Bibliography Managament in R Using the RefManageR Package

Mathew W. McLean
Texas A&M University

Abstract

This work introduces the R package **RefManageR**, which provides tools for importing and working with bibliographic references. It extends the `bibentry` class in **R** in a number of useful ways, including providing **R** with previously unavailable support for BibLaTeX. BibLaTeX provides a superset of the functionality of BibTeX, including full Unicode support, no memory limitations, additional fields and entry types, and more sophisticated sorting of references. **RefManageR** provides functions for citing and generating a bibliography with hyperlinks for documents prepared with `RMarkdown` or `RHTML`. Existing `.bib` files can be read into **R** and converted from BibTeX to BibLaTeX and vice versa. References can also be imported via queries to NCBI's Entrez, Zotero libraries, Google Scholar, and CrossRef. Additionally, references can be created by reading PDFs stored on the user's machine with the help of Poppler. Entries stored in the reference manager can be easily searched by any field, by date ranges, and by various formats for name lists (author by last names, translator by full names, etc.). Entries can also be updated, combined, sorted, printed in a number of styles, and exported.

Keywords: **R**, Biblatex, Bibtex, reference management, document generation, Unicode, **cURL**.

1. Introduction

Creating, managing, and processing references can often be a hastle. There are a number of reasons one may want or need to work with bibliographic data in **R** (R Core Team 2013), for example for bibliometrics. The `person` and `bibentry` classes available in the base-priority **utils** package since **R** 2.14.0 provide very useful functionality for working with names and bibliographic information, respectively. An introduction to these classes is available in Hornik, Murdoch, and Zeileis (2012). In this paper, I introduce the **RefManageR** package, which uses these classes as building blocks to greatly simplify working with bibliographies in **R**.

The `bibentry` class is designed to work with references in BibTeX format (Patashnik 1988).

RefManageR provides the `BibEntry` class which also works with BibTeX references, but additionally supports BibLaTeX formatting.

The BibTeX fields stored in a `bibentry` object can be easily accessed using the `` `$` `` operator, but there do not exist functions for conveniently conducting complicated searches. These are provided by the **RefManageR** package using the `` `[` `` operator. With this operator one may search a collection of references by any field or group of fields. BibLaTeX fields for lists of names, such as 'author' and 'editor', can be searched by family name only, full name, or full name with initials. Additionally, dates may be specified by ranges and are compared using the lubridate package (Grolemund and Wickham 2011). Entries may also be indexed by key, created in several different ways using functions `BibEntry` and `as.BibEntry`, and updated using the `` `[<-` `` operator. The `bibentry` class provides a method for the `c` generic for concatenating entries, our package retains this feature, while also providing a `merge` method to remove potential duplicate entries when combining entries from various sources.

Entries may be imported into **R** in a number of ways. A function is provided for reading in `.bib` files in BibLaTeX and BibTeX format. For machines with `Poppler` (`http://poppler.freedesktop.org`) installed, bibliographic metadata can be read from PDFs stored on the user's machine to generate a citation for each PDF. The package also contributes interfaces to the CrossRef, Zotero, and NCBI's Entrez APIs to search and import references from these resources, using the **RCurl** package (Lang 2013a) for the HTTP requests. References can additionally be obtained from a researcher's Google Scholar profile.

The package is equiped with additional printing formats and several bibliography and citation styles. All the bibliography sorting options available in BibLaTeX are available in **RefManageR**. A convenient interface for setting optional arguments for the most commonly used functions similar to the `options` function is used. In case it is necessary to convert between formats, for example when submitting to a journal that does not support BibLaTeX, a function is provided for converting a bibliography with BibLaTeX formatting back to BibTeX.

To our knowledge our package is the first of its kind to provide support for including citations and bibliographys with hyperlinks in `[R]HTML` and `[R]Markdown` documents. Links can point from each citation to their bibliography entry and vice versa, and hyperlinks are also automatically created for values in the BibLaTeX fields 'url', 'doi', and 'eprint'.

The rest of the document proceeds as follows: In Section 2 I show how to create bibliography entries in **R**, import them from local files, and discuss setting package options; Section 3 discusses importing references from the web; in Section 4 I discuss printing, sorting, and exporting references; in Section 5 I show how to search and update `BibEntry` objects; Section 6 introduces using **RefManageR** to cite references and print a bibliography of only cited references; lastly, Section 7 concludes.

2. Creating `BibEntry` Objects and Importing From Files

2.1. The `BibEntry` Function

Similar to the `bibentry` function in **utils**, **RefManageR** provides a function `BibEntry` for creating a `BibEntry` object containing a single reference, which can be combined with other references into a single `BibEntry` object. An entry is specified to the `BibEntry` function via an argument `bibtype` for the entry type, an argument `key` for the entry key and by specifying

other arguments in `field = value` form to the `"..."` argument. Though the 'year' field is still supported in BibLaTeX to allow backwards compatibility with BibTeX, the field 'date' is preferred and allows for a number of different formats for the date, which will be discussed later. The field 'journaltitle' is preferred for specifying journals, though 'journal' remains supported. Below I create and print an entry of type 'Article' with fields 'author', 'title', 'date', 'journaltitle', 'volume', and 'number'. The `print` function for `BibEntry` objects offers a number of features which will be discussed in detail later. Its default settings are chosen to mimic the defaults of BibLaTeX. The `toBiblatex` function can be used to display the entry in its `.bib` file format.

```
bib <- BibEntry(bibtype="Article", key = "barry1996", date = "1996-08",
  title = "A Diagnostic to Assess the Fit of a Variogram to Spatial Data",
  author = "Ronald Barry", journaltitle = "Journal of Statistical Software",
                volume = 1, number = 1)
bib

## [1] R. Barry. "A Diagnostic to Assess the Fit of a Variogram to
## Spatial Data". In: _Journal of Statistical Software_ 1.1 (Aug.
## 1996).

toBiblatex(bib)

## @Article{barry1996,
##   date = {1996-08},
##   title = {A Diagnostic to Assess the Fit of a Variogram to Spatial Data},
##   author = {Ronald Barry},
##   journaltitle = {Journal of Statistical Software},
##   volume = {1},
##   number = {1},
## }
```

BibLaTeX offers a huge amount of additional functionality compared to BibTeX. For the full details, one can see the 253 page user manual (Lehman, Kime, Boruvka, and Wright 2013). BibLaTeX expands the number of automatically recognized entry types and fields offered by BibTeX, allowing for much more detailed bibliographic entries, while still maintaining compatibility with BibTeX. For example, to handle an `arXiv` eprint in BibTeX, one needs to use or create a special BibTeX style, or perhaps use the 'note' and 'year' fields in unintended ways. The below entry is used in an attempt to cite a submitted manuscript of the first author's using the `bibentry` function.

```
bibentry("misc", key = "mclean2013bayesian", author = "M. W. McLean and
         F. Scheipl and G. Hooker and S. Greven and D. Ruppert",
         title = "Bayesian Functional Generalized Additive Models
                 with Sparsely Observed Covariates", year = "Submitted",
         note = "arXiv eprint: 1305.3585")

## McLean MW, Scheipl F, Hooker G, Greven S and Ruppert D
```

```
## (Submitted). "Bayesian Functional Generalized Additive Models with
## Sparsely Observed Covariates." arXiv eprint: 1305.3585.
```

Though arXiv provides suggestions for creating BibTeX entries for their papers (http://arxiv.org/hypertex/bibstyles/), there is a frustrating lack of consistency in how people choose to create BibTeX entries for their arXiv papers. In BibLaTeX, there is greatly expanded support for electronic publications with fields for eprint, eprinttype, eprintclass, urldate, and pubstate. One can cite the same article in BibLaTeX without the need of a special `.bst` file or the note field using

```
@misc{mclean2013bayesian,
  author = {M. W. McLean and F. Scheipl and G. Hooker
                and S. Greven and D. Ruppert},
  title = {Bayesian Functional Generalized Additive Models
                with Sparsely Observed Covariates},
  urldate = {2013-10-06},
  date = {2013},
  eprinttype = {arxiv},
  eprintclass = {stat.ME},
  eprint = {1305.3585},
  pubstate = {submitted},
}
```

The entry can be created using the `BibEntry` function in **RefManageR**

```
BibEntry("misc", key = "mclean2013bayesian", author = "McLean, M. W. and
         Scheipl, F. and Hooker, G. and Greven, S. and Ruppert, D.",
         title = "Bayesian Functional Generalized Additive Models
                with Sparsely Observed Covariates", urldate = "2013-10-06",
         date = "2013", eprinttype = "arxiv", eprintclass = "stat.ME",
         eprint = "1305.3585", pubstate = "submitted")

## [1] McLean, M. W., Scheipl, et al. _Bayesian Functional
## Generalized Additive Models with Sparsely Observed Covariates_.
## 2013. arXiv: 1305.3585 [stat.ME]. (Visited on 10/06/2013).
## Submitted.
```

In BibLaTeX the 'eprint' identifier will automatically become a hyperlink to the paper on `arXiv`.

The `bibentry` class supports BibTeX-style crossreferencing, while the `BibEntry` class. Cross references are handled specially when indexing and searching `BibEntry` objects and discussed in Section 5.1. In a similar vain as cross-referencing, BibLaTeX supports an entry type "XData" which is never printed, but may be used to store fields that are shared by several entries. Entries can specify a field 'xdata' containing a comma separated list of keys belonging to XData entries that the entry inherits from. The following example demonstrates its use for

online references available on arXiv, and uses the `c` operator for combining `BibEntry` objects.

```
bib <- BibEntry(bibtype="XData", key = "statME", eprinttype = "arxiv",
                eprintclass = "stat.ME")
bib <- c(bib, BibEntry(bibtype="XData", key = "online2013", year = "2013",
                       urldate = "2013-12-20"))
toBiblatex(bib)

## @XData{statME,
##   eprinttype = {arxiv},
##   eprintclass = {stat.ME},
## }
##
## @XData{online2013,
##   year = {2013},
##   urldate = {2013-12-20},
## }

bib <- c(bib, BibEntry(bibtype="Online", key="mclean2013rlrt",
  author = "Mathew McLean and Giles Hooker and David Ruppert",
  title = "Restricted Likelihood Ratio Tests for Scalar-on-Function Regression",
  eprint = "1310.5811", url = "http://arxiv.org/abs/1310.5811",
  xdata = "statME,online2013"))
bib <- c(bib, BibEntry(bibtype="Online", key="mclean2013bayesian",
  author = paste("Mathew McLean and Fabian Scheipl and Giles Hooker",
                 "and Sonja Greven and David Ruppert"),
  title = paste("Bayesian Functional Generalized Additive Models",
                "for Sparsely Observed Covariates"),
  eprint = "1305.3585", url = "http://arxiv.org/abs/1305.3585",
  xdata = "statME,online2013"))
bib

## XData: online2013
##
## XData: statME
##
## [1] M. McLean, G. Hooker and D. Ruppert. _Restricted Likelihood
## Ratio Tests for Scalar-on-Function Regression_. 2013. arXiv:
## 1310.5811 [stat.ME]. <URL: http://arxiv.org/abs/1310.5811>
## (visited on 12/20/2013).
##
## [2] M. McLean, F. Scheipl, G. Hooker, et al. _Bayesian Functional
## Generalized Additive Models for Sparsely Observed Covariates_.
## 2013. arXiv: 1305.3585 [stat.ME]. <URL:
## http://arxiv.org/abs/1305.3585> (visited on 12/20/2013).
```

The cross-referencing system in BibLaTeX and **RefManageR** is more sophisticated than the symmetric field mapping system used in BibTeX, allowing for less cluttering and duplication of fields. In BibLaTeX, the "InBook" entry type is used for a self-contained work with its own title within a book, as opposed to simply referring to an untitled part of a book as in BibTeX. In the following example, involving an 'InBook' entry inheriting from a 'Book' entry, there is no need to create a 'booktitle' field duplicating the 'title' field in the parent entry to pass on to the child entry, and there is also no need to create an empty 'subtitle' field in the child entry to ensure it does not incorrectly inherit the 'subtitle' of the parent.

```
c(BibEntry("book", key = "parent", title = "The Book Title", year = 2012,
           subtitle = "The Book Subtitle", author = "Book Author",
           publisher = "A publisher"), BibEntry("inbook", key = "child",
           crossref = "parent", title = "The Title of the In Book Entry",
           author = "In Book Author"))

## [1] B. Author. _The Book Title. The Book Subtitle_. A publisher,
## 2012.
##
## [2] I. B. Author. "The Title of the In Book Entry". In: B. Author.
## _The Book Title. The Book Subtitle_. A publisher, 2012.
```

RefManageR recognizes some, but not all, localization keys defined by default in BibLaTeX. A localization key is a special value that BibLaTeX parses for certain fields and replaces with predefined text called the 'localization string' when printing the bibliography. In the example below I use localization keys to specify the roles of editors using the 'editortype' field and refer to portions of a text using the 'bookpagination' field.

```
BibEntry(bibtype="Collection", key = "jaffe", editor = "Phillip Jaff\u00eb",
  title = "Regesta Pontificum Romanorum ab condita ecclesia ad annum post
  Christum natum {MCXCVIII}", date = "1885/1888",
  editora = "S. Loewenfeld and F. Kaltenbrunner and P. Ewald",
  editoratype = "redactor", totalpages = "10", bookpagination = "section")

## [1] P. Jaffë, ed. _Regesta Pontificum Romanorum ab condita
## ecclesia ad annum post Christum natum MCXCVIII_. Red. by S.
## Loewenfeld, F. Kaltenbrunner and P. Ewald. 1885-1888.
```

2.2. Reading `.bib` Files Into R

RefManageR provides the function `ReadBib` for parsing `.bib` files in BibLaTeX or BibTeX format and creating `BibEntry` objects from them. This function is based on the `read.bib` function in package **bibtex** (Francois 2013), which uses code for parsing BibTeX files from Beebe (2004). `ReadBib` expands on `read.bib` by providing BibLaTeX support; by having an argument/option `check`, which can be disabled, and checks that each entry in the file has all the fields required by that BibLaTeX or BibTeX entry type; and also has expanded handling of name list fields to ensure that complicated names are correctly converted to `person` objects.

This last feature is important when searching the `BibEntry` object later; as it would not be possible to properly search by parts of a name (such as family name only) if a name has not been correctly converted to a `person` object. Since it is often not necessary in BibLaTeX to provide all the "required" fields for an entry, it can be useful to be able turn off the check for required fields in **R** when one wants to work with entries that are missing some fields. For example, the sample `.bib` file that comes with the BibLaTeX package, and is also included with **RefManageR** for demonstration purposes has three entries that are missing required fields. The default behaviour is to not add these entries, but this can be changed.

```
file <- system.file("Bib", "biblatexExamples.bib", package = "RefManageR")
bib <- ReadBib(file, check = "error")


## Ignoring entry titled "The Chicago Manual of Style" because A bibentry of bibtype
##  'Manual' has to specify the field: c("author", "editor")
## Ignoring entry titled "CTAN" because A bibentry of bibtype 'Online' has to
## specify the field: c("author", "editor")
## Ignoring entry titled "Computers and Graphics" because A bibentry of bibtype '
## Periodical' has to specify the field: editor


bib <- ReadBib(file, check = FALSE)
print(bib[c("cms", "jcg", "ctan")], .opts = list(check.entries = FALSE))

## [1] _The Chicago Manual of Style. The Essential Guide for Writers,
## Editors, and Publishers_. 15th ed. Chicago, Ill.: University of
## Chicago Press, 2003. ISBN: 0-226-10403-6.
##
## [2] _Computers and Graphics_. 35.4 (2011): _Semantic 3D Media and
## Content_. ISSN: 0097-8493.
##
## [3] _CTAN. The Comprehensive TeX Archive Network_. 2006. <URL:
## http://www.ctan.org> (visited on 10/01/2006).
```

2.3. Creating Citations From PDFs

Using the function `ReadPDFs` and the freely available software Poppler (http://poppler.freedesktop.org), it is possible to create references from PDFs stored on a user's machine. The user specifies a directory containing PDFs (or a single PDF file) which are then read by Poppler and converted to `.txt` files which are read into **R**, parsed into citations, and output as a `BibEntry` object.

The function will first search the text for a Document Object Identifier (DOI), and if one is found, the citation information will be downloaded from CrossRef using their API. This feature will be discussed in more detail in the next section. The function also works especially well with PDFs downloaded from jstor.org by recognizing the format of the cover page that JSTOR generates. This allows for detailed and accurate citations to be obtained. The function also recognizes papers downloaded from http://arXiv.org and parses the arXiv identifier in its current and pre-March 2007 format.

If there is no DOI available and the document does not have a JSTOR cover page, it is considerably more difficult to obtain an accurate citation. The function is often able to recover the title, author, and date information. It can parse journal title, volume, and issue information if it is present in an obvious format. Articles with complicated formatting and missing the features discussed in the previous paragraph are not likely to be parsed correctly and the user will have to manually edit the entries, which will be covered in a later section. With the following code, Windows binaries of Poppler are downloaded along with some PDFs to test out the function.

```
tmpdir <- tempdir()
tmpfile <- tempfile(".zip", tmpdir)
download.file("http://dl.dropbox.com/u/3291828/Poppler/poppler.0.22.0_win32.zip",
              tmpfile)
unzip(tmpfile, exdir = tmpdir)
curdir <- getwd()
setwd(file.path(tmpdir, "bin", fsep = "\\"))
download.file("http://www.jstatsoft.org/v56/i11/paper", "jss.pdf",
                mode = "wb")
download.file("http://arxiv.org/pdf/math/0703791",
              destfile = "FIZaop.pdf", mode = "wb")
download.file("http://arxiv.org/pdf/math/0703858",
              destfile = "PBHTaos.pdf", mode = "wb")
download.file("http://biomet.oxfordjournals.org/content/83/4/715.full.pdf",
  destfile = "ADVb.pdf", mode = "wb")
download.file("http://www.jstor.org/stable/pdfplus/25645718.pdf",
              destfile = "jstor.pdf", mode = "wb")
```

```
bib <- ReadPDFs(".")

## Getting Metadata for 5 pdfs...
## Getting 1 BibTeX entries from CrossRef...
## Done


bib

## [1] A. AZZALINI and A. D. VALLE. "The Multivariate Skew-normal
## Distribution". In: _Biometrika_ (1996), pp. 715-726.
##
## [2] R. Chepesiuk. "JSTOR and Electronic Archiving". In: _American
## Libraries_ 31.11 (2000), pp. 46-48. JSTOR: 25645718. <URL:
## http://www.jstor.org/stable/25645718>.
##
## [3] S. Fang, P. Imkeller and T. Zhang. "Global flows for
## stochastic differential equations without global Lipschitz
## conditions". In: _The Annals of Probability_ 35.1 (Jan. 2007), pp.
```

```
## 180-205. DOI: 10.1214/009117906000000412. <URL:
## http://dx.doi.org/10.1214/009117906000000412>.
##
## [4] S. Luo, Y. Chen, X. Su, et al. _Meta-analysis_. 2014.
##
## [5] D. Paul, E. Bair, T. Hastie, et al. _"Pre-conditioning" For
## Feature Selection And Regression In High-dimensional Problems_.
## Apr. 16, 2013. arXiv: math/0703858v1. <URL:
## http://arxiv.org/abs/math/0703858v1>.
```

Note that entry [4] is not complete. To clean up use `setwd(curdir)` and `unlink(tmpdir)`.

2.4. Conversion of Other Object Types to Class `BibEntry`

The `as.BibEntry` function will convert objects of several other data types to `BibEntry` if they have the proper format. Acceptable formats include a named `character vector` with entries for 'bibtype', 'key' and other fields for a single reference; a `list` of named `character vectors` for multiple references; `bibentry` objects; or a `data frame` with one row per entry, and columns for each field, including a column for 'bibtype'. For `data frames`, the row names should provide the 'keys'.

The following example uses the function to create a `Bibentry` object containing entries for every installed **R** package. It makes use of the `installed.packages` and `citation` functions in the **utils** package.

```
pkg.names <- rownames(installed.packages())
pkg.bib <- lapply(pkg.names, function(pkg){
  refs <- as.BibEntry(citation(pkg))
  if (length(refs))
    names(refs) <- make.unique(rep(pkg, length(refs)))
  refs
})
```

Keys may be extracted from `BibEntry` objects using either the `names` method or the `` `$` `` operator with `name` argument (the value to the right of the '$' sign) equal to 'key'. The extra step involving the `` `names<-` `` method for `BibEntry` objects assigns a unique key to each entry, as the `citation` function does not provide a key for each entry and may return more than one reference for a single package. At this point, because of the use of `lapply`, `pkg.bib` is a `list` of `BibEntry` objects, instead of a single `BibEntry` object. One way to rectify this is using the internal function `MakeCitationList`. Additionally, the `` `names<-` `` method can be used to assign keys. Entries in our `BibEntry` object of packages may be referred to using the package name/key because of the special features of the `` `[` `` operator for `BibEntry` objects, which will be discussed in detail in a later section.

```
pkg.bib <- RefManageR:::MakeCitationList(pkg.bib)
pkg.bib["boot"]
```

```
## [1] A. Canty and B. D. Ripley. _boot: Bootstrap R (S-Plus)
## Functions_. R package version 1.3-9. 2013.

pkg.bib[key = "boot"]

## [1] A. Canty and B. D. Ripley. _boot: Bootstrap R (S-Plus)
## Functions_. R package version 1.3-9. 2013.
##
## [2] A. C. Davison and D. V. Hinkley. _Bootstrap Methods and Their
## Applications_. ISBN 0-521-57391-2. Cambridge: Cambridge University
## Press, 1997. <URL: http://statwww.epfl.ch/davison/BMA/>.
```

Using `pkg.bib["boot"]` matches the entry with key exactly "boot". Using `pkg.bib[key = "boot"]`, a (partial) match occurs for any entry whose 'key' contains the `string` "boot", due to the default settings of the `` `[` `` operator discussed in Section 5. The `BibEntry` class also has methods `as.data.frame` and `unlist` to convert `BibEntry` objects to a data frame and unlist'ed vector, respectively. The function `as.data.frame` will create a `data frame` from a `BibEntry` object with each row corresponding to a unique entry and one column for every field present in the `BibEntry` object, including a column called 'bibtype' for the type of entry. `NA` values indicate that the field is not present in that entry (row of the `data frame`). The row names will be the 'key's of the entries.

2.5. Setting Package Options

The package contains a convenience function `BibOptions` for changing packages options. This function behaves similarly to the `options` function in **base**. This allows the user to set default values for several arguments to the most commonly used functions, so the user does not have to specify them each call. Options may be specified in `name = value` pairs or as a `list`, and current values may be extracted by specifying a `character vector` of option names. Discussion of most of the options is left to later sections of the document when introducing the functions the options affect. One option already encountered, is whether to check to ensure that each entry has values for all the fields required by BibLaTeX for its entry type. As mentioned, though entries have required fields in BibLaTeX, they are not really required as the package will work and generate a citation in any reasonable situation with missing fields. The option is named `check.entries`, and the default value is `"error"`. With this setting, an error is thrown when an attempt is made to use an entry with missing fields and a new entry is not created when an attempt is made to create an entry with missing fields. The value `"warn"` results in a warning being thrown when an entry with missing fields is encountered, but execution will not be stopped. Lastly, the value `FALSE` turns off checking of entries entirely. In the following example, use of the `BibOptions` function is demonstrated and it is shown what happens for the different settings of the `check.entries` option.

```
BibOptions("check.entries")

## $check.entries
## [1] "error"
```

```
BibEntry(bibtype = "Online", key = "ctan", date = "2006",
 title = "The Comprehensive TeX Archive Network", url = "http://www.ctan.org")

## Error: A bibentry of bibtype 'Online' has to specify the field: c("author"
## , "editor")


old.opt.val <- BibOptions(check.entries = FALSE)
BibEntry(bibtype = "Online", key = "ctan", date = "2006",
 title = "The Comprehensive TeX Archive Network", url = "http://www.ctan.org")

## [1] _The Comprehensive TeX Archive Network_. 2006. <URL:
## http://www.ctan.org>.

BibOptions(old.opt.val)  # restore the old value of the option
```

The default values of all options can be restored using `BibOptions(restore.defaults = TRUE)`. A list of all options and their current values can be obtained by calling the function with no arguments, i.e. `BibOptions()`.

3. Importing Citations From the Web

3.1. NCBI's Entrez

The National Center for Biotechnology Information's Entrez Global Query Cross-Database Search provides access to a large number of databases related to health sciences. **RefManageR** provides an interface to Entrez which allows for searching for references and parsing them to `BibEntry` objects. Additionally, users may look up references given a set of PubMed ID's, search for ID's for references already stored in a `BibEntry` object, and search for related works to references already in **R**. The full documentation for Entrez is available in Sayers (2009).

The first **RefManageR** function to be discussed is `ReadPubMed`, which uses the ESearch E-Utility. Among other features, ESearch is used to search any of Entrez's 38 databases (not just PubMed) using a query string and returns a list of entries in the database that match the query by their IDs. These IDs are then used to retrieve bibliographic information using another call to Entrez which is parsed into a `BibEntry` object and returned by `ReadPubMed`. The next example does a simple search for some of Raymond J. Carroll's publications.

```
rjc.pm <- ReadPubMed("raymond j. carroll", database = "PubMed")
rjc.pm[[1L]]

## [1] Y. Cho, N. D. Turner, L. A. Davidson, et al. "Colon cancer
## cell apoptosis is induced by combined exposure to the n-3 fatty
## acid docosahexaenoic acid and butyrate through promoter
## methylation". In: _Experimental biology and medicine (Maywood,
## N.J.)_ (2014). DOI: 10.1177/1535370213514927. PMID: 24495951.
```

The `"..."` argument of `ReadPubMed` can be used to pass additional optional arguments to ESearch. Among them are `retmax` to specify the maximum number of entries to return, `retstart` to specify the index of the first result to return, and `field` to search only a particular field of the entries for a match. For controlling the date of the matches there are options, `datetype` which gives the type of date to consider when searching by date; for example `datetype = "pdat"` specifies to search by publication date and `datetype = "mdat"` specifies to search by modification date. The `mindate` and `maxdate` options specify the minimum and maximum dates that the search results should be restricted to. Dates should be in the format "YYYY", "YYYY/MM", or "YYYY/MM/DD". Our next query returns one entry published in 2009 in the Journal of Statistical Software

```
ReadPubMed("journal of statistical software", field = "journal", retmax = 1,
    mindate = 2009, maxdate = 2009)

## [1] S. Holmes, A. Kapelner and P. P. Lee. "An Interactive Java
## Statistical Image Segmentation System: GemIdent". In: _Journal of
## statistical software_ 30.10 (2009). PMID: 21614138.
```

The `GetPubMedRelated` function uses the ELink E-Utility to find related articles to a set of articles or IDs. Either a character vector of IDs or a `BibEntry` object containing entries with 'eprinttype' field equal to "pubmed" and pubmed ID's stored in the 'eprint' field (the format expected by BibLaTeX and also returned by the `ReadPubMed` function) should be specified for the `id` argument. ELink can perform in two distinct ways given a set of IDs, either search for related articles for each ID in the set separately, or use the entire set at once to find articles that are related to every article specified by the set of IDs. The latter type of behaviour is requested in `GetPubMedRelated` by specifying `batch.mode = TRUE` as an argument in the call. In the below example I find related entries to the articles returned by the previous query for publications by RJC.

```
GetPubMedRelated(rjc.pm, batch.mode = TRUE, max.results = 1)

## [1] J. Fan and Y. Wu. "Semiparametric estimation of covariance
## matrices for longitudinal data". In: _Journal of the American
## Statistical Association_ 103.484 (2008), pp. 1520-1533. DOI:
## 10.1198/016214508000000742. PMID: 19180247.
```

Entrez returns a similarity score with each returned citation giving a measure of how similar the returned entry is to the specified IDs. These scores can be returned in the outputted `BibEntry` object in a field called 'score' by specifying `return.sim.scores = TRUE` in the call. Additionally, the IDs in the call that were used to determine the relation can be included in the output in a field called 'PMIDrelated' if the argument `return.related.ids` is `TRUE`. In the next example, `batch.mode = FALSE` is used and one related article is returned for each of two entries in `rjc.pm`.

```
BibOptions(check.entries = FALSE)
ids <- rjc.pm$eprint[3:4]
ids
```

```
## $guenther2014healthy
## [1] "24453128"
##
## $li2013selecting
## [1] "24376287"


related <- GetPubMedRelated(ids, batch.mode = FALSE, max.results = c(1, 1),
                      return.sim.scores = TRUE, return.related.ids = TRUE)
toBiblatex(related)


## @Article{guenther2008evaluation,
## title = {Evaluation of the Healthy Eating Index-2005},
## author = {Patricia M Guenther and Jill Reedy and Susan M Krebs-Smith and
## Bryce B Reeve},
## year = {2008},
## journal = {Journal of the American Dietetic Association},
## volume = {108},
## number = {11},
## pages = {1854-64},
## eprint = {18954575},
## doi = {10.1016/j.jada.2008.08.011},
## eprinttype = {pubmed},
## score = {54583749},
## pmidrelated = {24453128},
## }
##
## @Article{seghouane2007criterion,
## title = {The AIC criterion and symmetrizing the Kullback-Leibler
## divergence},
## author = {Abd-Krim Seghouane and Shun-Ichi Amari},
## year = {2007},
## journal = {IEEE transactions on neural networks / a publication of the
## IEEE Neural Networks Council},
## volume = {18},
## number = {1},
## pages = {97-106},
## eprint = {17278464},
## doi = {10.1109/TNN.2006.882813},
## eprinttype = {pubmed},
## score = {20610997},
## pmidrelated = {24376287},
## }
```

The `LookupPubMedID` function is provided by **RefManageR** to use Entrez's Ecitmatch to search for PubMed IDs for entries in an existing `BibEntry` object. In the following, I read in a

BibTeX file of references to RJC papers from Google Scholar and search for PubMed ID's for the first ten entries. If the search is successful and an ID is found, the corresponding entry is updated so that the 'eprinttype' field is assigned the value "pubmed" and the 'eprint' field is assigned the ID.

```
file.name <- system.file("Bib", "RJC.bib", package = "RefManageR")
bib <- ReadBib(file.name)
bib <- LookupPubMedID(bib, seq_len(10))

## Success for entries: 1, 2, 8, 9


bib[eprinttype = "pubmed"][[1L]]  # print entry for first located ID

## [1] N. Serban, A. M. Staicu and R. J. Carroll. "Multilevel
## Cross-Dependent Binary Longitudinal Data". In: _Biometrics_ 69.4
## (2013), pp. 903-913. PMID: 24131242.
```

Finally, the `GetPubMedByID` function uses Entrez's Efetch to obtain bibliography data given a vector of PubMed ID's. The just obtained PubMed IDs can be used to get the BibTeX entry from Entrez and compare it with the one already in our bibliography from Google Scholar.

```
GetPubMedByID(unlist(bib$eprint)[1L])

## [1] N. Serban, A. Staicu and R. J. Carroll. "Multilevel
## cross-dependent binary longitudinal data". In: _Biometrics_ 69.4
## (2013), pp. 903-13. DOI: 10.1111/biom.12083. PMID: 24131242.
```

If one wishes to use other NCBI E-Utilities and does not wish to work with `BibEntry` or `bibentry` objects, see the **rentrez** package (Winter 2012).

3.2. Zotero

`Zotero` is free, open source software for collecting and sharing bibliographic information. `Zotero` can automatically retrieve bibliographic metadata that has been embedded in web-pages using `ContextObjects in Spans` (`COinS`), and is thus a very convenient way to collect bibliographic information when browsing, for example, journal websites. The **RefManageR** package contains functions for querying existing `Zotero` libraries and converting the results to a `BibEntry` object and also for uploaded an existing `BibEntry` object to a Zotero library. To use the Zotero API, one needs a Zotero account, a userID and an API key for the library one wishes to access. The userID and API key for personal libraries may be found by logging in and visiting the page `https://www.zotero.org/settings/keys`. The following call to `ReadZotero` searches for the first two references with the word 'Bayesian' in the title contained in the library specified by the `'key'` parameter.

```
ReadZotero(user = '1648676', .params = list(q = 'bayesian',
                                key = '7lhgvcwVq60CDi7E68FyE3br', limit = 2))

## [1] P. Müller and R. Mitra. "Bayesian Nonparametric Inference -
## Why and How". In: _Bayesian Analysis_ 8.2 (Jun. 2013), pp.
## 269-302. ISSN: 1936-0975. DOI: 10.1214/13-BA811. <URL:
## http://projecteuclid.org/euclid.ba/1369407550> (visited on
## 10/24/2013).
##
## [2] K. Sriram, R. Ramamoorthi and P. Ghosh. "Posterior Consistency
## of Bayesian Quantile Regression Based on the Misspecified
## Asymmetric Laplace Density". In: _Bayesian Analysis_ 8.2 (Jun.
## 2013), pp. 479-504. ISSN: 1936-0975. DOI: 10.1214/13-BA817. <URL:
## http://projecteuclid.org/euclid.ba/1369407561> (visited on
## 10/24/2013).
```

3.3. Google Scholar

A function is provided for downloading citations from a public Google Scholar profile. This function is partially based on the function `get_publications` in the **scholar** package (Keirstead 2013), but provides additional functionality and processes the results into a `BibEntry` object. The function requires the Google Scholar ID of the researcher of interest. A user can obtain this ID by navigating to the researcher's Google Scholar profile and copying the value of the `user` parameter in the URL. The profile must be public for the function to work. The function assumes that each entry is either of type 'Article' or type 'Book'. If any numbers are available with the entry relating to journal volume, number, or pages; then the entry will be classified as type 'Article'. Otherwise, the type will be 'Book'. The code that follows will return the Raymond J. Carroll's three most recent papers indexed by Google Scholar.

```
## RJC's Google Scholar profile is at:
## http://scholar.google.com/citations?user=CJOHNoQAAAAJ
rjc.bib <- ReadGS(scholar.id = 'CJOHNoQAAAAJ', sort.by.date = TRUE,
                  limit = 3)
rjc.bib

## [1] Y. Cho, N. D. Turner, L. A. Davidson, et al. "Colon cancer
## cell apoptosis is induced by combined exposure to the n-3 fatty
## acid docosahexaenoic acid and butyrate through promoter
## methylation". In: _Experimental Biology and Medicine_
## 1535370213514927 (2014).
##
## [2] P. M. Guenther, S. I. Kirkpatrick, J. Reedy, et al. "The
## Healthy Eating Index-2010 Is a Valid and Reliable Measure of Diet
## Quality According to the 2010 Dietary Guidelines for Americans".
## In: _The Journal of nutrition, jn._ 113 (2014).
##
```

```
## [3] M. P. Little, A. G. Kukush, S. V. Masiuk, et al. "Impact of
## Uncertainties in Exposure Assessment on Estimates of Thyroid
## Cancer Risk among Ukrainian Children and Adolescents Exposed from
## the Chernobyl Accident". In: _PLOS ONE_ 9.1 (2014).
```

The function also stores the number of citations of each result. Each `BibEntry` will store the number of citations in a field `'cites'`, which is ignored when generating a bibliography by BibLaTeX or BibTeX without additional effort to handle a custom entry field. The following code will obtain the second author's three most cited works according to Google Scholar and prints the citation count and entry type for each entry.

```
## RJC's Google Scholar profile is at:
## http://scholar.google.com/citations?user=CJOHNoQAAAAJ
rjc.bib <- ReadGS(scholar.id = 'CJOHNoQAAAAJ', sort.by.date = FALSE,
                  limit = 3)
rjc.bib

## [1] R. J. Carroll and D. Ruppert. _Transformation and weighting in
## regression_. CRC Press, 1988.
##
## [2] R. J. Carroll, D. Ruppert, L. A. Stefanski, et al.
## _Measurement error in nonlinear models: a modern perspective_. CRC
## press, 2012.
##
## [3] D. Ruppert, M. P. Wand and R. J. Carroll. _Semiparametric
## regression_. Cambridge University Press, 2003.

cbind(rjc.bib$cites, rjc.bib$bibtype)

##                            [,1]   [,2]
## carroll2012measurement    "2495" "Book"
## ruppert2003semiparametric "1931" "Book"
## carroll1988transformation "1416" "Book"
```

A shortcoming of this approach, is that long author lists, long titles, or long journal/publisher info can all lead to incomplete information being returned for those fields for the offending entries. In this case, the `ReadGS` function will either not include entry or provide a add the entry with a warning depending on the value of the `check.entries` argument.

```
## RJC's Google Scholar profile is at:
## http://scholar.google.com/citations?user=CJOHNoQAAAAJ
rjc.bib <- ReadGS(scholar.id = 'CJOHNoQAAAAJ', sort.by.date = FALSE,
                  limit = 10, check.entries = 'error')

## Incomplete author information for entry "Structure of dietary measurement error:
## results of the OPEN biomarker study" it will NOT be added
```

```
rjc.bib2 <- ReadGS(scholar.id = 'CJOHNoQAAAAJ', sort.by.date = FALSE,
                   limit = 10, check.entries = 'warn')

## Incomplete author information for entry "Structure of dietary measurement error:
## results of the OPEN biomarker study" adding anyway


length(rjc.bib) == length(rjc.bib2)

## [1] FALSE

## the offending entry.  RJC is missing because list of authors was too long
print(rjc.bib2[title='dietary measurement error'],
      .opts = list(max.names = 99, bib.style = 'alphabetic'))

## [Kip+03] V. Kipnis, A. F. Subar, D. Midthune, L. S. Freedman, R.
## Ballard-Barbash and and. "Structure of dietary measurement error:
## results of the OPEN biomarker study". In: _American Journal of
## Epidemiology_ 158.1 (2003), pp. 14-21.
```

3.4. CrossRef

The function `ReadCrossRef` uses the CrossRef Metadata Search API (http://search.crossref.org/help/api) to import references based on a search of CrossRef's nearly 60 million records. Given a search term and possibly a search year, the function receives BibTeX entries as JSON objects using the **RJSONIO** package (Lang 2013b), which are saved to a temporary file and then read back into **R** using the `ReadBib` function to be returned as a `BibEntry` object.

```
ReadCrossRef(query = 'rj carroll measurement error', limit = 3,
             sort = "relevance", min.relevance = 80, verbose = FALSE)

## [1] R. J. Carroll, D. Ruppert and L. A. Stefanski. "Measurement
## Error in Nonlinear Models".  (1995). DOI:
## 10.1007/978-1-4899-4477-1. <URL:
## http://dx.doi.org/10.1007/978-1-4899-4477-1>.
##
## [2] R. J. Carroll, D. Ruppert and L. A. Stefanski. "Response
## Variable Error". In: _Measurement Error in Nonlinear Models_
## (1995), p. 229-242. DOI: 10.1007/978-1-4899-4477-1_13. <URL:
## http://dx.doi.org/10.1007/978-1-4899-4477-1_13>.
##
## [3] D. Ruppert, M. P. Wand and R. J. Carroll. "Measurement Error".
## In: _Semiparametric Regression_ (2003), p. 268-275. DOI:
## 10.1017/cbo9780511755453.017. <URL:
## http://dx.doi.org/10.1017/cbo9780511755453.017>.
```

Although false negatives are rare, the CrossRef Metadata Search can be prone to false positives. For this reason, it is important to specify the `min.relevance` argument. Each reference returned by CrossRef comes with a relevancy score which is CrossRef's determination of how likely the reference is to be a match for the supplied query. The maximum possible value is 100, so for the most strict possible matching, specify `min.relevance = 100`. If the argument `verbose` is `TRUE`, then a message is printed with the relevancy score and full citation for each reference with a relevancy score greater than `min.reference` in addition to returning the references in a `BibEntry` object.

4. Sorting, Printing, Opening, and Outputting to File

4.1. Printing

A number of BibLaTeX bibliography styles are available in **RefManageR** for formatting and displaying citations. The styles currently implemented are "numeric" (the default), "authortitle", "authoryear", "alphabetic", and "draft". The "authoryear" style always begins with the family name of the first author and follows the list of authors with the year of publication in parentheses. The other four styles all use the same format, differing only in the label they print before each entry. Style "numeric" prints the numeric index of each entry in the bibliography, style "authortitle" uses no label, style "alphabetic" creates a label using the family names of the authors and the last two digits of the publication year, and style "draft" uses the entry key as the label.

Entries may be printed as plain text, HTML, BibTeX format, BibLaTeX format, as **R** code, Markdown, or as a mixture of BibTeX and plain text commonly used for citations. For an example of the "authoryear" style

```
file.name <- system.file("Bib", "biblatexExamples.bib", package = "RefManageR")
bib <- ReadBib(file.name, check = FALSE)
print(bib[author = "Nietzsche"], .opts = list(bib.style = "authoryear"))

## Nietzsche, F. (1988a). _Sämtliche Werke. Kritische
## Studienausgabe_. Ed. by G. Colli and M. Montinari. 2nd ed. Vol.
## 15. 15 vols. München and Berlin and New York: Deutscher
## Taschenbuch-Verlag and Walter de Gruyter.
##
## --- (1988b). _Sämtliche Werke. Kritische Studienausgabe_. Vol. 1.:
## _Die Geburt der Tragödie. Unzeitgemäße Betrachtungen I-IV.
## Nachgelassene Schriften 1870-1973_. Ed. by G. Colli and M.
## Montinari. 2nd ed. München and Berlin and New York: Deutscher
## Taschenbuch-Verlag and Walter de Gruyter.
##
## --- (1988c). "Unzeitgemässe Betrachtungen. Zweites Stück. Vom
## Nutzen und Nachtheil der Historie für das Leben". In: F.
## Nietzsche.  _Sämtliche Werke. Kritische Studienausgabe_. Vol. 1.:
## _Die Geburt der Tragödie. Unzeitgemäße Betrachtungen I-IV.
```

```
## Nachgelassene Schriften 1870-1973_. Ed. by G. Colli and M.
## Montinari. München and Berlin and New York: Deutscher
## Taschenbuch-Verlag and Walter de Gruyter, pp. 243-334.
```

The package has a number of options similar to those available in BibLaTeX, including `dashed` to control the use of dashes for duplicate authors as in the above example, `max.names` to control the number of names in name list fields that will be printed before they are truncated with "et al.", and `first.inits` to control whether given names are truncated to first initials or full names are used. These options can be set using the `BibOptions` function or passed as options to the `.opts` argument of the `print` method. There is also a package option, `no.print.fields` for supressing the printing of certain fields.

```
old.opts <- BibOptions(bib.style = "alphabetic", max.names = 2,
                       first.inits = FALSE)
bib[bibtype = "report"]

## [CC78] Willy W. Chiu and We Min Chow. _A Hybrid Hierarchical Model
## of a Multiple Virtual Storage (MVS) Operating System_. Research
## rep. RC-6947. IBM, 1978.
##
## [PFT99] Jitendra Padhye, Victor Firoiu, et al. _A Stochastic Model
## of TCP Reno Congestion Avoidance and Control_. Tech. rep. 99-02.
## Amherst, Mass.: University of Massachusetts, 1999.

BibOptions(old.opts)  # reset to original values
print(bib[[19]], .opts = list(style = "html", no.print.fields = "url",
      bib.style = "authortitle"))

## <p><cite>Spiegelberg, H.
## &ldquo;&ldquo;Intention&rdquo; und &ldquo;Intentionalität&rdquo; in
## der Scholastik, bei Brentano und Husserl&rdquo;.
## In: <EM>Studia Philosophica</EM> 29 (1969), pp. 189-216.</cite></p>
```

The user can create a custom BibLaTeX or BibTeX bibliography style using the `bibstyle` fuction in the `tools` package. To do this involves creating an environment containing functions for formatting entries of each type with signatures such as `formatArticle(paper)` and `formatBook(paper)`.

A downside of BibLaTeX is that the majority of academic journals do not support its use, having long ago written a custom `bst` file for generating citations which can only be used by BibTeX. For this reason **RefManageR** provides a `toBibtex` method returning a character vector with entries converted from BibLaTeX to BibTeX format. Entries of a type that are not supported by BibTeX will be converted to a type that is, e.g., entries of type 'report' are converted to type 'techreport'. Other conversions include replacing the 'date' field with a properly formatted 'year' field (if year is not already present) and converting the 'journaltitle' field to 'journal'. Since the cross-referencing system in BibTeX is more limited than the one supported by BibLaTeX, an attempt is made to ensure the cross-referencing will still work as

expected in BibTeX. All fields not normally supported by BibTeX are dropped unless they are specified in the argument `extra.fields`. The argument `note.replace.field` can be used to specify fields to add to the 'note' field in entries that are missing it. As already demonstrated, the `toBiblatex` function will convert a `BibEntry` object to a character vector contains lines of the corresponding BibLaTeX-formatted bibliography. No fields are converted or dropped by this function; in this way it is very similar to the `toBibtex` method for `bibentry` objects.

```
ref <- BibEntry("thesis", key = "schieplthesis", date = "2011-03-17", url =
"http://edoc.ub.uni-muenchen.de/13028/", urldate = "2014-03-06", title =
"Bayesian Regularization and Model Choice for Structured Additive Regression",
type = "phdthesis", institution = "LMU Munich", author = "Fabian Scheipl")
toBiblatex(ref)

## @Thesis{schieplthesis,
## date = {2011-03-17},
## url = {http://edoc.ub.uni-muenchen.de/13028/},
## urldate = {2014-03-06},
## title = {Bayesian Regularization and Model Choice for Structured Additive
## Regression},
## type = {phdthesis},
## institution = {LMU Munich},
## author = {Fabian Scheipl},
## }

toBibtex(ref, note.replace.field = "urldate")

## @PhdThesis{schieplthesis,
## url = {http://edoc.ub.uni-muenchen.de/13028/},
## title = {Bayesian Regularization and Model Choice for Structured Additive
## Regression},
## author = {Fabian Scheipl},
## year = {2011},
## month = {mar},
## school = {LMU Munich},
## note = {Last visited on 03/06/2014},
## }
```

The function `WriteBib`, based on the function `write.bib` in the package **bibtex** (Francois 2013), is provided for writing a `BibEntry` object to a `bib` file in BibLaTeX or BibTeX format using `toBiblatex` and `toBibtex`, respectively, depending on the value of the `biblatex` logical argument to `WriteBib`. In the next example I write the previous thesis reference to a file in BibTeX format, and for demonstration purposes only, read back in the `.bib` file using `read.bib` in package **bibtex** so that a `bibentry` object is created instead of a `BibEntry` one.

```
tmpfile <- tempfile(fileext = ".bib")
WriteBib(ref, file = tmpfile, biblatex = FALSE, verbose = FALSE)
library(bibtex)
read.bib(tmpfile)

## Scheipl F (2011). _Bayesian Regularization and Model Choice for
## Structured Additive Regression_. PhD thesis, LMU Munich. Last
## visited on 03/06/2014, <URL:
## http://edoc.ub.uni-muenchen.de/13028/>.

unlink(tmpfile)
```

4.2. Sorting

Nine different methods are available for sorting citations stored in a `BibEntry` object, corresponding to the ones predefined in BibLaTeX. Depending on the `bib.style` option, the default sorting method is `"nty"` to sort by 'name' ('n'), then 'title' ('t'), then 'year'/'date' ('y'). Other possibilities are "debug" to sort by 'key', "none" for no sorting, "nyt", "nyvt", "anyt", "anyvt", "ynt", and "ydnt"; where the 'a' stands for sorting by alphabetic label, 'v' stands for sorting by 'volume', and 'yd' for sorting by 'year'/'date' in descending order.

All sorting methods first consider the field 'presort', if available. Entries with no 'presort' field are assigned 'presort' value "mm". Next the 'sortkey' field is used. When sorting by name, the 'sortname' field is used first. If it is not present, the 'author' field is used, if that is not present 'editor' is used, and if that is not present 'translator' is used. When sorting by 'title', first 'sorttitle' is considered. Similarly, when sorting by 'year', 'sortyear' is first considered. When sorting by 'volume', if the field is present, it is padded to four digits with leading zeros; otherwise, the string "0000" is used. When sorting by alphabetic label, first 'shorthand' is considered, then 'label', then 'shortauthor', 'shorteditor', 'author', 'editor', and 'translator'. Refer to Lehman *et al.* (Sections 3.1.2.1 and 3.5 and Appendix C.2 2013) for further details.

4.3. Opening Connections to References

Using the `open` method for `BibEntry` objects, it is possible to open a connection to a copy of an entry in the bibliography. This will work for entries that have a value for the 'file' field (specifying the path to a local copy of the reference), the 'doi' field, the 'url' field, or the 'eprint' field when the 'eprinttype' field is equal to `"jstor"`, `"arxiv"`, or `"pubmed"`. Which of those fields are used and in which order they are checked for can be specified using the `open.field` argument. The viewer to use can be specified (as a path) using the `viewer` argument. By default the value `getOptions("pdfviewer")` is used when opening values stored in the 'file' field, and the value of `getOptions("browser")` is used to open values in the other fields. Recalling our bibliography of installed packages, the 'url' field can be used to open the resource for the **base** package. Additionally, a path to the **R** Language Definition manual can be given in the 'file' field, so that that can be opened instead when requested.

```
open(pkg.bib[["base"]])  # will use the 'url' field
pkg.bib[["base"]]$file <- file.path(R.home("doc/manual"), "R-lang.pdf")
open(pkg.bib[["base"]], open.field = c("file", "url"))
```

5. Searching and Manipulating `BibEntry` Objects

5.1. Extraction Operators - Searching and Indexing

The extraction operator `` `[` ``, has been defined for `BibEntry` objects to allow for easily searching a database of references saved in a `BibEntry` object. A different interface providing the same functionality is the function `SearchBib`. Search options can be changed by set variables in the BibOptions object or alternatively specified directly as arguments to the function `SearchBib`. BibLaTeX date fields ('date', 'year', 'origdate', 'urldate', 'eventdate') and name lists ('author', 'editor', 'editora', 'editorb', 'editorc', 'translator', 'commentator', 'annotator', 'introduction', 'foreword', 'afterword', 'bookauthor', and 'holder') are handled specially as outlined below. Other fields can be searched using either exact string matching or regular expressions, with or without ignoring case (controlled via options `use.regex` and `ignore.case`, respectively).

Indices and search terms can be specified in a number of ways. Similar to the default extraction operator for list objects, a vector of numeric indices or logical values can be given. Additionally, a character vector of 'key' values can be specified. To search by field, a query can be specified with comma delimited `field=search.term` pairs, with `search.term` potentially being a vector with length greater than one to match multiple terms for `field` (think "OR"). Each `field = search.term` pair will have to match to declare a match for that entry (think "AND"). Multiple queries ("OR") can be handled (involving different fields) by enclosing each separate query preferably in a `list` or alternatively, `c`. For example, $\text{list}(field_{11} = \textbf{search.term}_{\textbf{11}}, field_{12} = \textbf{search.term}_{\textbf{12}}), \text{list}(field_{21} = \textbf{search.term}_{\textbf{21}})$. If `c` is used instead of `list`, then the search terms *must* have length one. Examples will be provided shortly after discussing the special handling of date and name fields. An '!' at the beginning of a search term can be used to negate a match (obviously, this can also be done using regular expressions).

Valid values for date fields in BibLaTeX have the form `yyyy`, `yyyy-mm`, `yyyy-mm-dd`, and can be intervals of the form `yyyy/yyyy`, `yyyy-mm/yyyy-mm`, `yyyy-mm-dd/yyyy-mm-dd`. The second date can be omitted in the interval to allow for open-ended end dates, e.g. `yyyy/`. When searching using a date field, the search string should have one of these formats. Additionally, the search string can be an interval with no start date, e.g. `date = "/1980-06"` to return all entries published before June, 1980. The **lubridate** package (Grolemund and Wickham 2011) is used to compare date fields, dates specified as intervals are converted to class `Interval` and non-interval dates are converted to class `POSIXct`. The format `yyyy-mm` *is* currently supported despite not being supported in base **R** or **lubridate**. For compatibility with BibTeX, BibLaTeX and **RefManageR** support the fields `year` and `month`, which are used if the `date` field is missing. Whether to ignore month and day values, if available, and only compare based on the year portion of the date field, is controlled by the option `match.date`, which supports two values "exact" or "year.only".

When searching name list fields, the search term is expected to have the same format as used in

a `.bib` file, e.g., `"Doe, Jr., John and Jane {Doe Smith}"`. Names can be matched based on family names only, by family name and given name initials, or by full name, depending on the value of the option `match.author`.

Entries containing valid `crossref` and `xdata` fields are expanded prior to searching, so that when a match is found for a field and value that a child entry inherits from its parent, the result is both the parent and child being returned. If a match is found in a child entry and not in the parent, only the child entry is returned, but the returned entry will contain any fields it inherits from its parent. Any xdata entries that the child references will also be returned. Examples follow.

```
file.name <- system.file("Bib", "biblatexExamples.bib", package = "RefManageR")
bib <- ReadBib(file.name, check = FALSE)
# by default match.author = 'family.only' and ignore.case = TRUE inbook
# entry inheriting editor field from parent
bib[editor = "westfahl"]

## [1] G. Westfahl, ed. _Space and Beyond. The Frontier Theme in
## Science Fiction_. Westport, Conn. and London: Greenwood, 2000.
##
## [2] G. Westfahl. "The True Frontier. Confronting and Avoiding the
## Realities of Space in American Science Fiction Films". In: _Space
## and Beyond. The Frontier Theme in Science Fiction_. Ed. by G.
## Westfahl. Westport, Conn. and London: Greenwood, 2000, pp. 55-65.

# no match with parent entry, the returned child has inherited fields
bib[author = "westfahl"]

## [1] G. Westfahl. "The True Frontier. Confronting and Avoiding the
## Realities of Space in American Science Fiction Films". In: _Space
## and Beyond. The Frontier Theme in Science Fiction_. Ed. by G.
## Westfahl. Westport, Conn. and London: Greenwood, 2000, pp. 55-65.
```

```
# Entries published in Zürich (in bib file Z{\'u}ich) OR entries written by
# Aristotle and published before 1930
bib[list(location = "Zürich"), list(author = "Aristotle", year = "/1930")]

## [1] Aristotle. _De Anima_. Ed. by R. D. Hicks. Cambridge:
## Cambridge University Press, 1907.
##
## [2] Aristotle. _Physics_. Trans.  by P. H. Wicksteed and F. M.
## Cornford. New York: G. P. Putnam, 1929.
##
## [3] Aristotle. _The Rhetoric of Aristotle with a commentary by the
## late Edward Meredith Cope_. Ed. by E. M. Cope. With a comment. by
## E. M. Cope. Vol. 3. 3 vols. Cambridge University Press, 1877.
```

```
##
## [4] Homer. _Die Ilias_. Trans.  by W. Schadewaldt. With an intro.
## by J. Latacz. 3rd ed. Düsseldorf and Zürich: Artemis \& Winkler,
## 2004.

length(bib[author = "!knuth"])

## [1] 85
```

The list extraction operator, `[[`, is used for extacting `BibEntry` objects by position (an integer) or the entry key (a string). Unlike the default operator, a vector of indices may be given to extract more than one entry at a time.

As with `bibentry` objects, the `$` operator for `BibEntry` objects is used to return a list containing the value of a particular field for all entries, with a value of `NULL` returned for entries that do not have the specified field. A list of all entry types or keys for the `BibEntry` object, `bib`, can be obtained using `bib$bibtype` and `bib$key`, respectively.

5.2. Assignment Operators

List assignment, `[[<-` is used for replacing one entry in a `BibEntry` object with another. The below example uses a bibliography of just under 500 works of Raymond J. Carroll indexed on Google Scholar. It contains a number of errors, some of which are corrected below to help demonstrate the use of the package. I make use of the `logical` option `return.ind` to have the search return a `numeric vector` of indices as opposed to a `BibEntry` object.

```
file.name <- system.file("Bib", "RJC.bib", package = "RefManageR")
bib <- ReadBib(file.name)
## length(bib)
length(bib) == length(bib[author = "Carroll"])

## [1] FALSE

# which entries are missing RJC?
ind <- SearchBib(bib, author = "!Carroll", .opts = list(return.ind = TRUE))
bib[ind]$author

## $z2010oracle
## [1] "J G M AR TI NE Z" "R J C AR RO LL"
##
## $caroll2006measurement
## [1] "R J Caroll"      "D Ruppert"       "L A Stefanski"   "C M Crainiceanu"
##
## $ll1996measurement
## [1] "R J C AR RO LL"
##
## $wu1989estimation
```

```
## [1] "M C Wu"     "K R Bailey"
##
## $carroll1989covariance
## [1] "R J Caroll"
```

Clearly, one paper is incorrectly attributed to RJC and the other four have spelling errors. We thus drop that entry and correct the spelling on the other four entries.

```
bib <- bib[-ind[4L]]
bib[author="!Carroll"]$author <- c("Martinez, J. G. and Carroll, R. J.",
 "Carroll, R. J. and Ruppert, D. and Stefanski, L. A. and Crainiceanu, C. M.",
 "Carroll, R. J.", "Carroll, R. J.")
length(bib) == length(bib[author="Carroll"])

## [1] TRUE
```

I can update different fields of multiple entries using the operator `` `[<-` `` as follows.

```
BibOptions(sorting = "none", bib.style = "alphabetic")
bib[seq_len(3)]

## [SSC13] N. Serban, A. M. Staicu and R. J. Carroll. "Multilevel
## Cross-Dependent Binary Longitudinal Data". In: _Biometrics_ 69.4
## (2013), pp. 903-913.
##
## [Jen+13] E. M. Jennings, J. S. Morris, R. J. Carroll, et al.
## "Bayesian methods for expression-based integration of various
## types of genomics data". In: _EURASIP Journal on Bioinformatics
## and Systems Biology_ 2013.1 (2013), pp. 1-11.
##
## [Gar+13] T. P. Garcia, S. Müller, R. J. Carroll, et al.
## "Identification of important regressor groups, subgroups and
## individuals via regularization methods: application to gut
## microbiome data". In: _Bioinformatics, btt_ 608 (2013).

bib[seq_len(3)] <- list(c(date="2013-12"), ## add month to Serban et al.
        c(url="http://bsb.eurasipjournals.com/content/2013/1/13",
          urldate = "2014-02-02"), ## add URL and urldate to Jennings et al.
        c(doi="10.1093/bioinformatics/btt608",
          journal = "Bioinformatics")) ## add DOI and correct journal
bib[seq_len(3)]

## [SSC13] N. Serban, A. M. Staicu and R. J. Carroll. "Multilevel
## Cross-Dependent Binary Longitudinal Data". In: _Biometrics_ 69.4
## (Dec. 2013), pp. 903-913.
##
```

```
## [Jen+13] E. M. Jennings, J. S. Morris, R. J. Carroll, et al.
## "Bayesian methods for expression-based integration of various
## types of genomics data". In: _EURASIP Journal on Bioinformatics
## and Systems Biology_ 2013.1 (2013), pp. 1-11. <URL:
## http://bsb.eurasipjournals.com/content/2013/1/13> (visited on
## 02/02/2014).
##
## [Gar+13] T. P. Garcia, S. Müller, R. J. Carroll, et al.
## "Identification of important regressor groups, subgroups and
## individuals via regularization methods: application to gut
## microbiome data". In: _Bioinformatics_ 608 (2013). DOI:
## 10.1093/bioinformatics/btt608.
```

Notice that I set `sorting = "none"` above. Sorting of the entries is done by default when printing, and after sorting the print order of entries is unlikely to correspond to the index order in the `BibEntry` object.

A `BibEntry` object may be used as the replacement value. A field may be removed by specifying its value be set to the empty string `''`.

```
bib2 <- bib[seq_len(3)]
bib2[2:3] <- bib[5:6]
# Note the Sarkar et al. entry is arXiv preprint with incorrect journal field
bib2

## [SSC13] N. Serban, A. M. Staicu and R. J. Carroll. "Multilevel
## Cross-Dependent Binary Longitudinal Data". In: _Biometrics_ 69.4
## (Dec. 2013), pp. 903-913.
##
## [TDC13] C. D. Tekwe, A. R. Dabney and R. J. Carroll. "Application
## of Survival Analysis Methodology to the Quantitative Analysis of
## LC-MS Proteomics Data". In: _AMINO ACIDS_ 45.3 (2013), pp.
## 609-609.
##
## [Sar+13] A. Sarkar, D. Pati, B. K. Mallick, et al. "Adaptive
## Posterior Convergence Rates in Bayesian Density Deconvolution with
## Supersmooth Errors". In: _arXiv preprint arXiv:_ 1308 (2013).

# Change type, remove journal, correct arXiv information
bib2[3] <- c(journal='', eprinttype = "arxiv", eprint = "1308.5427",
             eprintclass = "math.ST", pubstate = "submitted", bibtype = "Misc")
bib2[3]

## [Sar+13] A. Sarkar, D. Pati, B. K. Mallick, et al. _Adaptive
## Posterior Convergence Rates in Bayesian Density Deconvolution with
## Supersmooth Errors_. 2013. arXiv: 1308.5427 [math.ST]. Submitted.
```

5.3. Merging

The combine function, `c`, is available for concatenating multiple `BibEntry` objects, and has been inherited from the `bibentry` class. Of course, this does not perform any checking for duplicate entries. For this, there is the `base` package generics `anyDuplicated`, `duplicated`, and `unique`, which check vectors for duplicate elements. However, if `BibEntry` objects have been compiled from a number of different sources, these functions may be too strict, declaring entries distinct even if only one field has a small difference between the two entries. For this reason, an additional operator `'+'` is supplied along with a wrapper function `merge`, that compares entries only based on the fields specified by the user. Given `BibEntry` objects `bib1` and `bib2`, `bib1 + bib2` will return `bib1` appended with all entries of bib2 that have been determined not be duplicates of entries already in `bib1` by comparing all fields in `BibOptions()$merge.fields.to.check`, which can include `bibtype` and `key`. The function also checks if there are any duplicate keys in the result, and will force them to be unique if duplicates are detected using `make.unique`.

6. Using RefManageR in Dynamic Documents

One may print citations from a `BibEntry` object and generate a bibliography for all citations. This is especially useful for inclusion in `RMarkdown` or `RHTML` documents. In those two situations, which can be specified by setting `style = "markdown"` and `style = "html"`, respectively, in the `BibOptions` function, hyperlinks will also automatically be generated. These hyperlinks will either link to an external copy of the reference (using the same mechanism as the `open` method discussed in Section 4.3), or point from the citation to the bibliography entry and vice versa. This behaviour is controlled by the package option `hyperlink`. Owing to the features of the `` `[` `` operator for `BibEntry` objects, there is no need to restrict to only citing entries using their keys.

In addition to the main function `Cite`, the functions `Citet`, `Citep`, `TextCite`, and `AutoCite` are provided for convenience and mimic the output of the corresponding LaTeX commands in the `natbib` (Daly 2010) and `biblatex` LaTeX packages. Using the option `cite.style`, one can specify 'alphabetic', 'numeric', or 'authoryear' citations. Additionally, for the 'numeric' style, one may specify the option `super` to include the numeric citations as superscripts. The punctuation to use for enclosing the citations, separating multiple citations, etc. can be set using the `bibpunct` option. These options can be set in `BibOptions` or specified as a list to the `.opts` argument to any of the citation functions. A function `NoCite` is provided to include a reference in the bibliography without citing it. The bibliography is printed using the `PrintBibliography` function. It is very similar to the `print` method for the `BibEntry` class, except that it will only print the references that have been cited with one of the citation functions. A demonstration follows. Note that we change around the `cite.style` and `bib.style` to show the different styles, but one would normally set these at the start of the document and usually keep them set to be equal so that the labels in the citations and bibliography match.

```
BibOptions(check.entries = FALSE, bib.style = "authoryear", style = "text")
file.name <- system.file("Bib", "biblatexExamples.bib", package = "RefManageR")
bib <- ReadBib(file.name)
```

`Citet(bib, "loh")` produces Loh (1992), a "textual" citation using an entry key. It is possible to cite in parentheses by 'year' using `Citep(bib, year = "1899", .opts = list(cite.style = "alphabetic"))` [Wil99]. Next, three works by Averroes are cited `AutoCite(bib, author = "averroes", .opts = list(super = TRUE, cite.style = "numeric"))` [1;2;3]. There is some support for resolving ambiguous citations; consider `Citet(bib, author = "Baez")` Baez and Lauda (2004a); Baez and Lauda (2004b). Finally, the bibliography is printed using `PrintBibliography`.

```
PrintBibliography(bib, .opts = list(bib.style = "alphabetic"))

## [Ave69] Averroes. _Drei Abhandlungen über die Conjunction des
## separaten Intellects mit dem Menschen. Von Averroes (Vater und
## Sohn), aus dem Arabischen übersetzt von Samuel Ibn Tibbon_. Ed. by
## J. Hercz. Trans.  by J. Hercz. Berlin: S.~Hermann, 1869.
##
## [Ave82] Averroes. _The Epistle on the Possibility of Conjunction
## with the Active Intellect by Ibn Rushd with the Commentary of
## Moses Narboni_. Ed. by K. P. Bland. Trans.  by K. P. Bland.
## Moreshet: Studies in Jewish History, Literature and Thought 7. New
## York: Jewish Theological Seminary of America, 1982.
##
## [Ave92] Averroes. _Des Averroës Abhandlung: "Über die Möglichkeit
## der Conjunktion" oder "Über den materiellen Intellekt"_. Ed. by L.
## Hannes. Trans.  by L. Hannes. With annots. by L. Hannes. Halle an
## der Saale: C.~A. Kaemmerer, 1892.
##
## [BL04a] J. C. Baez and A. D. Lauda. "Higher-Dimensional Algebra V:
## 2-Groups". Version 3. In: _Theory and Applications of Categories_
## 12 (2004), pp. 423-491. arXiv: math/0307200v3.
##
## [BL04b] J. C. Baez and A. D. Lauda. _Higher-Dimensional Algebra V:
## 2-Groups_. Oct. 27, 2004. arXiv: math/0307200v3.
##
## [Loh92] N. C. Loh. "High-Resolution Micromachined Interferometric
## Accelerometer". MA Thesis. Cambridge, Mass.: Massachusetts
## Institute of Technology, 1992.
##
## [Wil99] O. Wilde. _The Importance of Being Earnest: A Trivial
## Comedy for Serious People_. English and American drama of the
## Nineteenth Century. Leonard Smithers and Company, 1899. Google
## Books: 4HIWAAAAYAAJ.
```

Typically, when using `knitr`, one would load **RefManageR**, load the bibliography, and set package options in a chunk at the start of the document using option `include = FALSE` and then include citations and print the bibliography with options `echo = FALSE` and `results = "asis"`. To see demonstrations of these functions use in `RMarkdown` and `RHTML` documents and the hyperlinking features, see the package `vignettes` as well as the examples at `?Cite`.

7. Conclusion

The **RefManageR** package provides **R** with considerable extra resources for working with bibliographic data; alleviating much of the difficulty of managing references from several different sources. Functions have been introduced for importing references from a number of online resources and additionally for conveniently editing entries and creating new ones. By implementing many of the features of BibLaTeX, several shortcomings of working with BibTeX format are removed. Conversion between different formats, bibliography styles, and between BibLaTeX and BibTeX is made easy with the package. The user is able to be less dependent on remembering entry keys when writing a document and is able to make complicated searches using a simple syntax with the `` `[` `` operator. As more and more researchers become aware of the benefits of working with `Markdown`, the citation, hyperlinking, and printing capabilities of **RefManageR** will be a useful tool.

Future work on **RefManageR** will include allowing for additional citation and bibliography styles, making it easier for users to define custom styles, and creating support for `Pandoc` (http://johnmacfarlane.net/pandoc/) style citations. Additionally, more work may be needed to ensure that searching and merging can be done very quickly for extremely large bibliographies for certain applications. I also wish to explore creating a revamped version of the `citEntry` function in package **utils** to allow package developers to include citations in BibLaTeX format in their packages.

Acknowledgements

The author was supported in part by a postdoctoral award from the Texas A&M Institute for Applied Mathematics and Computational Science, and in part by a grant from the National Cancer Institute (R37-CA057030, R. J. Carroll, P.I.). He would also like to thank R. J. Carroll for helpful comments on the manuscript and for having so many articles to reference in examples.

References

Beebe NHF (2004). "bibparse." URL http://ftp.math.utah.edu/pub/bibparse/.

Daly PW (2010). *Natural Science Citations and References*. URL http://ctan.mirrorcatalogs.com/macros/latex/contrib/natbib/natbib.pdf.

Francois R (2013). *bibtex: bibtex parser*. R package version 0.3-6, URL http://CRAN.R-project.org/package=bibtex.

Grolemund G, Wickham H (2011). "Dates and Times Made Easy with lubridate." *Journal of Statistical Software*, **40**(3), 1–25. URL http://www.jstatsoft.org/v40/i03/.

Hornik K, Murdoch D, Zeileis A (2012). "Who Did What? The Roles of R Package Authors and How to Refer to Them." *The R Journal*, **4**(1). URL http://journal.r-project.org/archive/2012-1/RJournal_2012-1.pdf.

Keirstead J (2013). *scholar: Analyse citation data from Google Scholar*. R package version 0.1.1, URL http://CRAN.R-project.org/package=scholar.

Lang DT (2013a). *RCurl: General network (HTTP/FTP/...) client interface for R*. R package version 1.95-4.1, URL http://CRAN.R-project.org/package=RCurl.

Lang DT (2013b). *RJSONIO: Serialize R objects to JSON, JavaScript Object Notation*. R package version 1.0-3, URL http://CRAN.R-project.org/package=RJSONIO.

Lehman P, Kime P, Boruvka A, Wright J (2013). *The biblatex Package*. URL http://ctan.mirrorcatalogs.com/macros/latex/contrib/biblatex/doc/biblatex.pdf.

Patashnik O (1988). *BibTeXing*. URL http://ctan.sharelatex.com/tex-archive/biblio/bibtex/base/btxdoc.pdf.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Sayers E (2009). "Entrez Programming Utilities Help." URL http://www.ncbi.nlm.nih.gov/books/NBK25499/.

Winter D (2012). *rentrez: Entrez in R*. R package version 0.1.1, URL http://CRAN.R-project.org/package=rentrez.

Affiliation:

Mathew W. McLean
Institute for Applied Mathematics and Computational Science
Texas A&M University
3143 TAMU
College Station, TX, 77843
E-mail: mmclean@stat.tamu.edu
URL: http://stat.tamu.edu/~mmclean