Reading Human Fertility Database and Human Mortality Database data into `R`

Tim Riffe

Department of Demography, University of California, Berkeley

June 1, 2015

Abstract

The features and usage of the `HMDHFDplus` package are demonstrated for reading data from the Human Mortality Database, the Human Fertility Database, and other similarly formatted sources directly from the database Websites into `R`.

1   Motivation

The Human Fertility Database (2015) and Human Mortality Database (2015) are two widely used data sources for the comparative and historical study of fertility and mortality. Both databases offer the option to download data in bulk in a few different formats. After a bulk download, users can then set up local databases in a variety of convenient ways, such as that described by Minton (2015) for `R` (R Development Core Team 2012) users. Another option for `R` users is to read data directly into an interactive session from the respective database websites. This is handy for small examples, lightweight reproducibility, and rapid prototyping. The `HMDHFDplus` package provides easy direct access to the databases using a simple standard set of arguments. Issues such as authentication and fixing column classes are handled automatically. Analogous functions are also made available for selected databases using similar formatting standards. At this time, these databases include the Japanese Mortality Database (2015), the Canadian Human Mortality Database (2015), and the Human Fertility Collection (2015).[1] This report outlines the basic features and provides usage examples.

---

[1] The Human Life-Table Database (2015) may also be incorporated in the future.

2   Installation

The `HMDHFDplus` package is hosted on and can be installed directly from `github`.[2] Two external dependencies help `R` handle `html` parsing and database authentication, and these must be installed first in order for `HMDHFDplus` to properly install. These two dependencies are `cURL` and `XML`, and their installation unfortunately depends on one's operating system at this time.

In `Linux` and similar systems, open the Terminal and run:

```
sudo apt-get install libcurl
sudo apt-get install libxml2-dev
```

When these external dependencies are properly installed, you should install `R`'s `XML` and `RCurl` packages in the usual way. Then install `HMDHFDplus` using:

```
library(devtools)
install_github("timriffe/TR1/TR1/HMDHFDplus")
```

For `Windows` and `Mac`, installation may be simpler yet.[3] Simply be sure to install the most recent version of `R`, then install the `devtools` package, then run the above two lines of `R` code.

Please consult the `README`[4] file on the `github` repository for more details.

3   Usage and examples

Load the package using:

```
library(HMDHFDplus)
```

The two main functions of interest are `readHMDweb()` and `readHFDweb()`, and both functions have the same essential arguments. These two functions only require the user to supply country codes, data item names, and database authentication parameters.[5] It helps to be familiar with HMD and HFD file naming conventions. To retrieve the population codes used in any of these databases, run:

```
getHFDcountries()
getHFCcountries()
getHMDcountries()
getJMDprefectures()
getCHMDprovinces()
```

---

[2]A package snapshot is included with this article, but I encourage users to use the current build on `github`, as it may contain updates and bug fixes. The package is under the following url: `http://github.com/timriffe/TR1/tree/master/TR1/HMDHFDplus`. This report, as well as a useful `README` document, can be found under `http://github.com/timriffe/TR1`.

[3]Not all versions of Windows and Mac have been tested.

[4]The `README` is at the foot of the main repository page: `https://github.com/timriffe/TR1`

[5]The HFD also allows users to extract data from former updates, which may be useful for strict replication purposes. In order to make use of this feature, users must note the 8-digit date code associated with the specific country series update. By default, `readHFDweb()` extracts the most recent update.

This returns vectors of the standard numerical or letter codes used to identify population units.

The functions used for reading data into R from the Web use a common set of required arguments. In interactive R sessions, the following will prompt the user to enter a username and password into the console (no quotes) each time the function is run:

```
# for HMD:
USmales <- readHMDweb(CNTRY = "USA", item = "mltper_1x1")
# for HFD (will need to re-enter username and password)
USfert <- readHFDweb(CNTRY = "USA", item = "asfrRR")
```

Manually entering a username and password can become tedious for larger data-grabs, so these can also be given explicitly in the arguments, like so:

```
USmales <- readHMDweb(CNTRY = "USA", item = "mltper_1x1", username =
"myusername", password = "mypassword")
USfert <- readHFDweb(CNTRY = "USA", item = "asfrRR", username =
"myusername", password = "mypassword")
```

There is a security trade-off in this case, because the username and password may inadvertently be saved within your R script. I suggest two alternatives in this case. First, in an interactive R session, define your username once at the beginning of the script, but without saving them as text within the script, like so:

```
pw <- userInput()
us <- userinput()
USmales <- readHMDweb(CNTRY = "USA", item = "mltper_1x1", username = us,
password = pw)
```

The two objects `pw` and `us` can in this case be recycled throughout the following R session. Second, For more frequent users, I recommend defining your HMD and HFD passwords in the `.Rprofile` file, such that they are defined and ready to use at the start of R sessions, but are not saved in your potentially-shared code. The above HMD code will return data such as the following: `data.frame`:

```
head(USmales)
  Year Age      mx      qx   ax     lx   dx    Lx      Tx    ex OpenInterval
1 1933   0 0.06859 0.06515 0.23 100000 6515 94978 5916978 59.17        FALSE
2 1933   1 0.01004 0.00999 0.50  93485  934 93018 5822000 62.28        FALSE
3 1933   2 0.00467 0.00466 0.50  92551  431 92336 5728982 61.90        FALSE
4 1933   3 0.00333 0.00333 0.50  92120  307 91967 5636646 61.19        FALSE
5 1933   4 0.00254 0.00253 0.50  91814  233 91697 5544679 60.39        FALSE
6 1933   5 0.00209 0.00209 0.50  91581  191 91485 5452982 59.54        FALSE
```

This `data.frame` differs from the original HMD `mltper_1x1` file in that the `Age` column is integer, and a new `OpenInterval` column has been added, which contains the value `TRUE` for age 110. HFD `Age` and `Cohort` columns are modified in a similar way for more intuitive and immediate use of these columns as integers. Likewise, abridged ages, such as `"5-9"` are coerced as integers of the lower interval bound, as `5`. Finally, HMD `Population` files, obtained via

3

```
USpop <- readHMDweb("USA","Population",username = us, password = pw)
head(USpop)
  Year Age OpenInterval   Female1    Male1  Total1   Female2     Male2   Total2
1 1933   0        FALSE  984472.3 1015362 1999834  937185.8  968955.4 1906141
2 1933   1        FALSE 1040496.0 1064088 2104584  970696.5  993352.8 1964049
3 1933   2        FALSE 1093043.8 1117527 2210571 1062002.5 1083452.4 2145455
4 1933   3        FALSE 1107994.3 1135047 2243041 1095555.1 1121220.2 2216775
5 1933   4        FALSE 1130624.4 1179514 2310138 1105999.3 1132665.9 2238665
6 1933   5        FALSE 1168930.6 1228225 2397156 1141944.4 1197735.4 2339680
```

, where columns ending in 1 indicate January 1st estimates and columns ending
in 2 indicate December 31st estimates, and the `Year` and `Age` columns are co-
erced to an integer class. The JMD, CHMD, and HFC are all called in similar
ways, but without authentication:

```
# 31 columns!
USasfrBO <- readHFC("USA","ASFRstand_BO")
# 5x5 male lifetables for Aomori prefecture:
Aomori <- readJMDweb("02","mltper_5x5")
# 5x5 lifetables for Alberta:
ALB <- readCHMDweb("alb","mltper_5x5")
```

The JMD and CHMD follow the same formatting standards and naming con-
ventions as the HMD, although the data products available are a subset of those
produced by the HMD. The HFC follows different standards and conventions
than the HFD.

4   Conclusions

Reading data directly from HMD, HFD and similar databases directly from the
web into R is made easy with the `HMDHFDplus` package. Common R pitfalls are
removed by coercing columns to useful classes by default. At this time, utilities
are provided for reading data from the HMD, JMD, CHMD, HFD, and HFC
websites.

5   Acknowledgements

References

Canadian Human Mortality Database. Department of Demography, Université de Montréal, 2015. Available at http://www.bdlc.umontreal.ca/chmd/.

Human Fertility Collection. Max Planck Institute for Demographic Research (Germany) and Vienna Institute of Demography (Austria), 2015. Available at http://www.fertilitydata.org/cgi-bin/index.php.

Human Fertility Database. Max Planck Institute for Demographic Research (Germany) and Vienna Institute of Demography (Austria), 2015. Available at www.humanfertility.org.

Human Life-Table Database. Max Planck Institute for Demographic Research (Germany) and University of California, Berkeley (USA) and Institut national d'études démographiques (France), 2015. Available at http://www.ipss.go.jp/p-toukei/JMD/index-en.html.

Human Mortality Database. University of California, Berkeley (USA) and Max Planck Institute for Demographic Research (Germany), 2015. Available at www.mortality.org or www.humanmortality.de (data downloaded on July 10, 2014).

Japanese Mortality Database. National Institute of Population and Social Security Research (Japan), 2015. Available at http://www.ipss.go.jp/p-toukei/JMD/index-en.html.

Jon Minton. Merging, exploring, and batch processing data from the human fertility database and human mortality database. Technical Report TR-2015-001, Max Planck Institute for Demographic Research (MPIDR), April 2015. URL http://www.demogr.mpg.de/papers/technicalreports/tr-2015-001.pdf.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. URL http://www.R-project.org/. ISBN 3-900051-07-0.