A Package for Transparent and Reproducible Statistics: Package FAOSTAT

Michael. C. J. Kao
Food and Agriculture Organization
of the United Nations

Abstract

The aim of this document is to introduce the FAOSTAT package developed by the Food and Agricultural Organization of the United Nations which serves as an open gate way to an extensive library of agricultural statistics on FAOSTAT and the core back bone for the production of the Statistcal Yearbook.

Dealing with official statistics is a very tedious and complex task which is usually overlooked. This paper will address some of the common problems we have encountered and at the same time demonstrate how some of these obstacles can be alleviated by the package and provide a framework for reproducible statistics.

The use of open source software R and LaTeX brings tremendous amount of benefits, speeding up the production process and open up the data and methodology to the general public.

In this small paper we will illustrate the production process and demonstrate how the use of the package can increase transparency and sustainability. We will also point out details along the process which are typically overseen by analysts and researchers.

Keywords: R, Official Statistics.

1. Introduction

The idea of utilizing R and LaTeX for the production of the FAO statistical yearbook was initiated by Adam Prakash and Mattieu Stigler in the ESS division of the Food and Agricultural Organization of the United Nations. The initiative was taken in order to replace the labour intensive work with a streamline system which integrates data extraction, manipulation, statistical graphics and tables into one single comprehensive system.

This paper will demonstrate how the FAOSTAT package is used to download, and process data under the framework of the FAO Statistical Yearbook. The goal is to provide a framework for users to access, and reproduce the statistics released in the publication and also easy the pain for dealing with official statistics from various sources. By providing the data and methodology to the public, we hope to receive feedbacks from domain experts and professionals in order to improve the system and at the same time to promote research and analysis in the field which will support for better policies and decision.

First, we will demonstration the usage of the `getFAOtoSYB` and `getWDItoSYB` functions to download data from the FAO FAOSTAT and the World Bank WDI API. This is then followed by the demonstration of the `translateCountryCode` and `mergeSYB` function to merge data from various sources and address some of the common complications. Finally we provide examples of how the aggregates can be computed using the function`aggRegion` for different composition.

2. Motivation

Compiling hundreds of statistics from different sources under traditional approach such as Excel can be very labour intensive and error proned. Furthermore, the knowledge and the experience is almost impossible to sustain in the long run resulting in inconsistent results and treatments over time. As a result, the ESS took the initiative to use R and LaTeX as the new architecture for a sustainable and cost-effective way to produce the statistical yearbook. This approach increases the sustainability and coherence of the publication as all the data manipulation, and exceptions are recorded in the source code.

In addition to these working motives, the use of R enables the data generated by the publication to be reproducible and readily accessible to researchers and analysts around the world. This open-data philosophy has proven to create tremendous amount of benefits for both the user and the data provider. We hope that this initiative will increase the visibility of agricultural related statistics and spark more research and analysis which the organization and its beneficiaries will gain.

Reproducibility is the norm in academics, this property allows one to verify, improve and reproduce the research for future use. We believe that a publication such as the statistical yearbook which publishes statistics and aggregates should be examined under the same transparency standard. The publication of the methodology is equally important as the statistics itself.

The package can be installed from the CRAN repository just like all other R packages, and this documentation is also the vignette of the package.

```
if(!is.element("FAOSTAT", .packages(all.available = TRUE)))
   install.packages("FAOSTAT")
library(FAOSTAT)
## vignette("FAOSTAT", package = "FAOSTAT")
```

3. Download data from FAOSTAT

FAOSTAT is the largest agricultural database in the world, it contains data from land productivity to agricultural production and trade dating back from 1960 to the most recent available data. Detailed information on meta data, methods and standards can be found on the official website of FAOSTAT http://faostat3.fao.org/home/index.html and the Statistics Division (ESS) http://www.fao.org/economic/ess/en/.

In order to access the data using the FAOSTAT API, the domain, item and element code of the indicator of interest is required. They are defined as:

Domain Code :
: The domain associated with the data. For example, production, trade, prices etc.

Item Code :
: These are the codes relating to the commodity or product group such as wheat, almonds, and aggregated item like "total cereals".

Element Code :
: Lastly, this is the quantity/unit or data collection type associated with the commodity. Typical elements are quantity, value or production/extraction rate.

An interactive function `FAOsearch` has been provided for the user to identify the respective codes. An object `.LastSearch` will be assigned after the search and can be used by both the `getFAO` and `getFAOtoSYB` functions as the sole argument to download the data.

```
## Use the interective function to search the code.
FAOsearch()

## Use the result of the search to download the data.
test = getFAO(query = .LastSearch)
```

The getFAOtoSYB is a wrapper for the getFAO to batch download data, it supports error recovery and stores the status of the download. The function also splits the data downloaded into entity level and regional aggregates, saving time for the user. Query results from FAOsearch can also be used.

```
## A demonstration query
FAOquery.df = data.frame(varName = c("arableLand", "cerealExp", "cerealProd"),
                         domainCode = c("RL", "TP", "QC"),
                         itemCode = c(6621, 1944, 1717),
                         elementCode = c(5110, 5922, 5510),
                         stringsAsFactors = FALSE)

## Download the data from FAOSTAT
FAO.lst = with(FAOquery.df,
    getFAOtoSYB(name = varName, domainCode = domainCode,
                itemCode = itemCode, elementCode = elementCode,
                useCHMT = TRUE, outputFormat = "wide"))
```

The object returned is a list of length three, these are entity level data, aggregates and the download status. The function supports both long and wide format.

In some cases multiple China are provided, for example the trade domain provides data on China (41), Taiwan (214) and China plus Taiwan (357). The CHMT function avoids double counting if multiple China are detected by removing the aggregated if detected. The default in getFAOtoSYB is to use CHMT when possible. Otherwise, the FAOcheck function can be used to sanitize the data.

```
FAOchecked.df = FAOcheck(var = FAOquery.df$varName, year = "Year",
    data = FAO.lst$entity)
```

4. Download data from World Bank

The World Bank also provides an API where data from the World Bank and various international organization are made public. More information about the data and the API can be found at http://data.worldbank.org/

The author is aware of the **WDI** package, but we have wrote this function before the recent update of the package with additional functionalities. We have plans to integrate with the **WDI** package to avoid confusion for the users.

```
## Download World Bank data and meta-data
WB.lst = getWDItoSYB(indicator = c("SP.POP.TOTL", "NY.GDP.MKTP.CD"),
                     name = c("totalPopulation", "GDPUSD"),
                     getMetaData = TRUE, printMetaData = TRUE)
```

The output is similar to the object generated by `getFAOtoSYB` except that if the arguement `getMetaData` is specified as TRUE then the meta data is also downloaded and saved.

One point to note here, it is usually unlikely to reconstruct the world aggregates provided by the World Bank based on the data provided. The reason is that the aggregate contains Taiwan when available, yet the statistics for Taiwan are not published.

5. Merge data from different sources

Merge is a typical data manipulation step in daily work yet a non-trivial exercise especially when working with different data sources. The built in *mergeSYB* function enables one to merge data from different sources as long as the country coding system is identified. Currently the following country coding translation are supported and included in the internal data set FAOcountryProfile of the package:

- United Nations M49 country standard [UN_CODE]
  http://unstats.un.org/unsd/methods/m49/m49.htm
- FAO country code scheme [FAOST_CODE]
  http://termportal.fao.org/faonocs/appl/
- FAO Global Administrative Unit Layers (GAUL).[ADM0_CODE]
- ISO 3166-1 alpha-2 [ISO2_CODE]
  http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
- ISO 3166-1 alpha-2 (World Bank) [ISO2_WB_CODE]
  http://data.worldbank.org/node/18
- ISO 3166-1 alpha-3 [ISO3_CODE]
  http://en.wikipedia.org/wiki/ISO_3166-1_alpha-3
- ISO 3166-1 alpha-3 (World Bank) [ISO3_WB_CODE]
  http://data.worldbank.org/node/18

Data from any sources employ country classification listed above can be supplied to `mergeSYB` in order to obtain a single merged data. However, the column name of the country coding scheme is required to be the same as the name in square bracket, the responsibility of identifying the coding system lies with the user.

Nevertheless, often only the name of the country is provided and thus merge is not possible or inaccurate based on names. We have provided a function to obtain country codes based on the names matched. In order to avoid matching with the wrong code, the function only attempts to fill in countries which have exact match.

```
## Just a demonstration
Demo = WB.lst$entity[, c("Country", "Year", "totalPopulation")]
demoResult = fillCountryCode(country = "Country", data = Demo,
    outCode = "ISO2_WB_CODE")

## Countries have not been filled in.
unique(demoResult[is.na(demoResult$ISO2_WB_CODE), "Country"])

## [1] "China" "Sudan"
```

We have not implemented a regular expression match for the identification reason listed below. From the above example we can see that both China and Sudan are not filled in, the identification of Sudan prior to 2011 and China should be carefully examined.

Below we list some commonly observed problem when merging data from different sources.

5.1. Identification problem

Due to the fact that different organization are bounded by different political agenda the users need to be aware of the precise definition and legal recognition of countries.

Take example, the China provided by the World Bank does not include Taiwan, Hong Kong and Macao. On the other hand, FAO provides not only a single China (FAO = 41), but also China plus Taiwan (FAO = 357) depending on the context. In addition, it is common to observed statistics for China (ISO2 = CN or ISO3 = CHN) which includes Taiwan, Hong Kong and Macao. The default trnslates China from other country coding scheme to Mainland China (FAO = 41) and is not matched in `fillCountryCode`.

5.2. Representation problem

Moreover, the situation is further complicated by disputed territories or economic union such as Kosovo and Belgium-Luxembourg which does not have representation under particular country coding system.


```
## Countries which are not listed under the ISO2 international standard.
FAO.df = translateCountryCode(data = FAOchecked.df, from = "FAOST_CODE",
    to = "ISO2_CODE")

##
## NOTE: Please make sure that the country are matched according to their definition

## Warning:  The following entries does not have 'ISO2_CODE' available

##     FAOST_CODE ISO2_CODE
## 29          15      <NA>
## 56         259      <NA>
## 58         351      <NA>
## 62         357      <NA>
## 95          62      <NA>
## 193        147      <NA>
##                                          OFFICIAL_FAO_NAME
## 29                                      Belgium-Luxembourg
## 56                                         Channel Islands
## 58  China (China mainland, Hong Kong SAR, Macao SAR, Taiwan)
## 62                            China (China mainland, Taiwan)
## 95                                               Ethiopia PDR
## 193                                   the Republic of Namibia


## Countries which are not listed under the UN M49 system.
WB.df = translateCountryCode(data = WB.lst$entity, from = "ISO2_WB_CODE",
    to = "UN_CODE")

##
## NOTE: Please make sure that the country are matched according to their definition
```

```
## Warning:  The following entries does not have 'UN_CODE' available

##     ISO2_WB_CODE UN_CODE OFFICIAL_FAO_NAME
## 155           KV      NA            Kosovo
```

5.3. Transition problem

Finally, the discontinuity and transition of countries further increases the complexity of the data. The South Sudan was recognized by the United Nations on the 9th of July 2011, however, the statistics reported by The republic of the Sudan in the same year can also includes data for South Sudan thus failing the mutually exclusive test. Moreover, sources which uses ISO standard country code have no way to distinguish between the new and old Sudan (SD and SDN are used for both entity) which causes problem in merge with country system that distingiushes the entity.

Finally, if historical aggregates are computed then a region composition which does not backtrack in time will result in an aggregate which is incorrect. For more details about historical and transitional countries please refer to http://unstats.un.org/unsd/methods/m49/m49chang.htm

Given the lack of an internationally recognized standard which incorporates all these properties, we suggests the use of the FAO country standard and region profile shipped with the package which addresses most of these problems.

```
merged.df = mergeSYB(FAOchecked.df, WB.lst$entity, outCode = "FAOST_CODE")

##
## NOTE: Please make sure that the country are matched according to their definition
##
##
## NOTE: Please make sure that the country are matched according to their definition
```

6. Computing regional or economical aggregates

Aggregation is another data manipulation step that is commonly over seen. The result can vary due to the differences between the regional composition and the set of countries used. Furthermore, it is complicated by the amount of missing values which can render the aggregates incomparable. Given the missing values and diverging country sets, aggregation can only serve as approximates in order to inform the general situation of the region. The following rules are implemented to ensure the aggregates computed are meaningful and comparable.

- A minimum threshold in which the data must be present, the default is 65% for every individual year.
- The number of reporting entities must be similar over the years. It does not make sense to compare aggregates of 1995 and 2000 if the number of reporting countries differ vastly, the default tolerance is 15.

In addition, historical countries are aggregated to ensure comparability over time. For example, The Former Soviet Union is not part of the current definition of the M49 standards, nevertheless, it would be ignorant to omit it from the aggregation.

```
## Compute aggregates under the FAO continental region.
relation.df = FAOregionProfile[, c("FAOST_CODE", "FAO_MACRO_REG")]

FAOregion.df = aggRegion(data = merged.df, relationDF = relation.df,
                         aggVar = c("arableLand", "cerealExp", "cerealProd",
                                    "totalPopulation", "GDPUSD"),
                         aggMethod = rep("sum", 5),
                         unspecifiedCode = "Unspecified")

## Compute aggregates under the UNSD M49 continental region.
relation.df = FAOregionProfile[, c("FAOST_CODE", "UNSD_MACRO_REG")]

UNregion.df = aggRegion(data = merged.df, relationDF = relation.df,
                        aggVar = c("arableLand", "cerealExp", "cerealProd",
                                   "totalPopulation", "GDPUSD"),
                        aggMethod = rep("sum", 5),
                        unspecifiedCode = "Unspecified")
```

7. Conclusion

Acknowledgement

The author owes a great debt to Fillipo Gheri, Adam Prakash, Guido Barbaglia, Amy Heyman, Amanda Gordon, Jacques Joyeux, and Markus Gesmann for their contribution to the pacakge, whom the package would not exist without their expertise.

The author would also like to express his profound gratitude to the directors Pietro Gennari and Josef Schimidhuber and the entire ESS division for their support, experience and every little thing they have done was greatly appreciated.

Affiliation:

Michael C.J. Kao
Economics and Social Statistics Division
Economic and Social Development Department
United Nations Food and Agriculture Organization
Viale delle Terme di Caracalla 00153 Rome, Italy
E-mail: michael.kao@fao.org