Manipulating data from Statistics Canada Tables

LJ Valencia[*]

Introduction

This paper is is an example on how to manipulate CANSIM data from Statistics Canada. Three R packages are used in this demonstration: `tidyverse`, `cansim`, and `writexl`. The `cansim` package is very useful in retrieving data tables and time-series from Canada's socioeconomic repository, CANSIM. The `tidyverse` package is a collection of R packages designed for data science[1]. Lastly, `writexl` is a package that is used to export the dataframes in '.xlsx' format. In this demonstration, I will be using the Consumer Price Index, monthly, not seasonally adjusted[2].

Importing Packages

The relevant packages are loaded using the `library()` function. It is important to set up a file cache using the `options()` function. Although this step is optional, it is highly recommended as it would speed up data retrieval. This is because the `cansim` package will cache the data in a temporary directory for the duration of the R session.

```
# import packages
library(cansim)
library(tidyverse)
library(writexl)
```

```
options(cansim.cache_path = "StatCan4R") # Set-up File Cache
```

List of Vectors

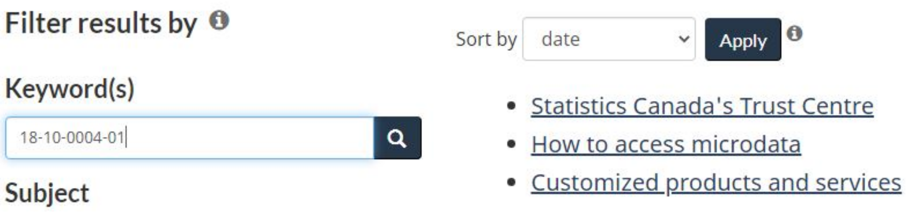
Next, is to search for the relevant data. Statistics Canada data are accessed by using vector identifiers. This is because retrieving by table number as the package will load the entire package, putting an excessive demand on RAM and slowing down the retrieval process. Searching the desired vectors is done by using the **Statistics Canada Data Search Tool**.

To find the correct vector identifier(s), enter the table number in the "Keyword(s)" field. Then click the matching StatCan table in the search results. Then click on Add/Remove data to select which specific dimensions and customize the data. It is important to be specific and check the right boxes when customizing the table as this yields the desired vector identifier(s). I customized the data table by CPI All-items, and in the four provinces across Western Canada. Copy the relevant identifiers and paste them to a vector object.

[*] Bachelor of Arts (Honors) in Economics, University of Alberta

[1] For more information, see https://www.tidyverse.org/

[2] Statistics Canada. Table 18-10-0004-01 Consumer Price Index, monthly, not seasonally adjusted



Filter results by

Keyword(s)

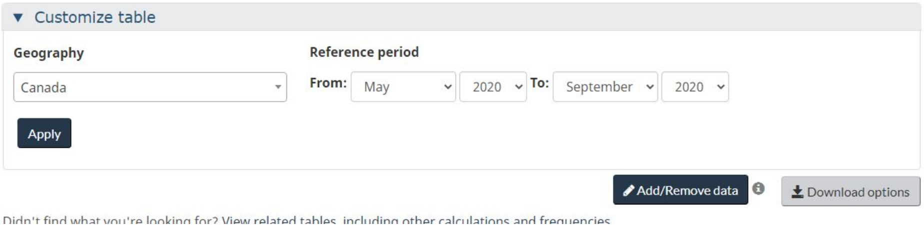
18-10-0004-01

Subject

Sort by date Apply

- Statistics Canada's Trust Centre
- How to access microdata
- Customized products and services

Figure 1: Search by table number



▼ Customize table

Geography: Canada

Reference period From: May 2020 To: September 2020

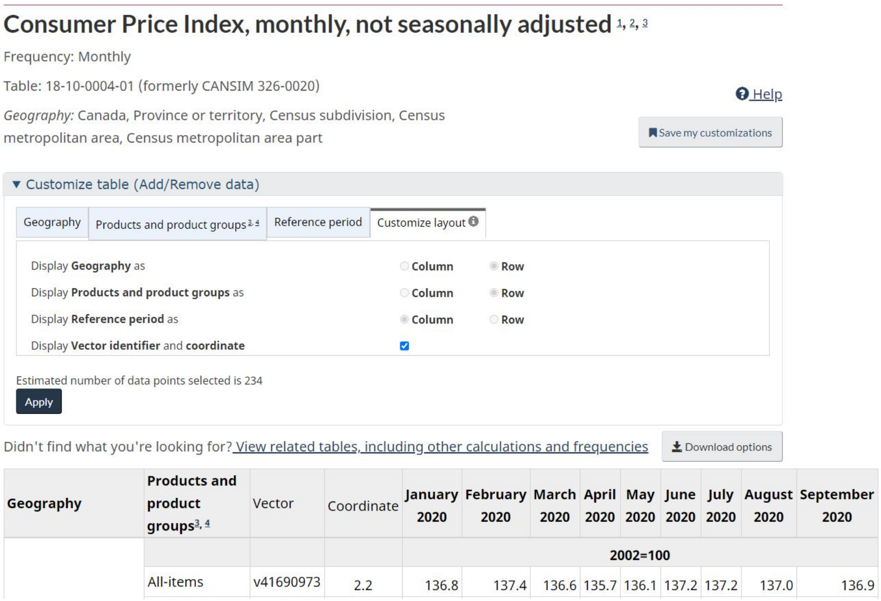
Apply

Add/Remove data

Download options

Didn't find what you're looking for? View related tables, including other calculations and frequencies

Figure 2: Click Add/Remove data



Consumer Price Index, monthly, not seasonally adjusted [1], [2], [3]

Frequency: Monthly

Table: 18-10-0004-01 (formerly CANSIM 326-0020)

Geography: Canada, Province or territory, Census subdivision, Census metropolitan area, Census metropolitan area part

Help

Save my customizations

▼ Customize table (Add/Remove data)

Geography | Products and product groups[3, 4] | Reference period | Customize layout

Display **Geography** as ○ Column ● Row

Display **Products and product groups** as ○ Column ● Row

Display **Reference period** as ● Column ○ Row

Display **Vector identifier** and **coordinate** ☑

Estimated number of data points selected is 234

Apply

Didn't find what you're looking for? View related tables, including other calculations and frequencies

Download options

Geography	Products and product groups[3, 4]	Vector	Coordinate	January 2020	February 2020	March 2020	April 2020	May 2020	June 2020	July 2020	August 2020	September 2020
				2002=100								
	All-items	v41690973	2.2	136.8	137.4	136.6	135.7	136.1	137.2	137.2	137.0	136.9

Figure 3: Check 'Display Vector indentifier and coordinate'

In the section of code below, the necessary vector identifiers are assigned into a vector object in R. Second, the desired start and end periods are defined. Next I use the `get_cansim_vector()` to retrieve the necessary tables. Lastly, the variable names in the dataframe are changed and the irrelevant columns are dropped.

```
# assign the vector identifiers into a vector
vectors.cpi <- c("v41692055",
                 "v41692191",
                 "v41692327",
                 "v41692462")
date.start <- "2000-01-01" # start period
date.end <- "2020-09-01" # end period
# retrieve tables using the vector
df <- get_cansim_vector(vectors.cpi, date.start, date.end) %>% # start and end dates
      rename(CPIallitems = VALUE) %>% # rename variables; rename VALUE into CPIallitems
      rename(date = REF_DATE) %>%
      rename(vector = VECTOR) %>%
      select(-c(1, 4:8)) # drop the irrelevant variables.
```

Spreading the Data

Although I have shown how to retrieve StatCan data using the `cansim` package, The `get_cansim_vector()` does not return a tidy data. Data points by vector indentifiers are piled on top of one another. To address this, the code below shows how to spread the data using the `spread()` function. This clean data is then exported as a '.xlsx' file.

Next, I use the `get_cansim_vector_info()` function to get the information about vectors. I use a for loop to repalce the vector names in the dataframe with their appropriate english names.

```
# get information about vectors
title.vectors <- get_cansim_vector_info(x) %>%
        select(-c(1,3:4,6:10)) # get the titles
for (i in 1:length(x)){ # rename vectors with their recognized titles;
  # replace vector numbers with actual names;
  # match vector with the appropriate titles.
  df[df==title.vectors$VECTOR[i]] <- title.vectors$title_en[i]
  }
df <- spread(df, key=vector, value=CPIallitems) # spread the dataframe
write_xlsx(df , path="StatCan4R\CPI-all-items.xlsx") # export data
```

Full Implementation using a Function

The code below is a full implementation, but with a function. Functions are very useful to perform specific tasks. First, I used an if-else statement to check if the vector is not a NULL type object. After using the `get_cansim_vector()` for retrieval and selection of useful information, I used a for loop to rename the vectors into more informative names. Then the data is spread using the `spread()` function. It is important to note that when creating a vector of names to replace the original identifiers, they have to match the order of their vector identifiers.

```r
date.start <- "2000-01-01" # start period
date.end <- "2020-09-01" # end period
# function for importing the necessary vectors
query <- function(x, date1, date2){
  if (is.null(x) != TRUE){# if list of vectors is not NULL (empty)
    if ((is.null(date1) != TRUE) && (is.null(date2) != TRUE)){# if dates are not NULL (empty)
      df <- get_cansim_vector(x, "2000-01-01", "2019-12-01") %>% # start and end dates
        rename(value = VALUE) %>% # rename variables
        rename(date = REF_DATE) %>%
        rename(vector = VECTOR) %>%
        select(-c(1, 4:8)) # drop the irrelevant variables.
      title.vectors <- get_cansim_vector_info(x) %>%
        select(-c(1,3:4,6:10)) # get the titles
      for (i in 1:length(x)){
        # replace vector numbers with actual names;
        # match vector with the appropriate english title
        df[df==title.vectors$VECTOR[i]] <- title.vectors$title_en[i]
      }
      df <- spread(df, key=vector, value=value) # spread the dataframe
      return(df) # return dataframe
    }
    else{ # warning statement that says inputs are not valid; 'stop' halts execution of code.
      stop("Please enter appropriate start and end dates.")
    }
  }
  else{
    stop("Your inputs are not valid. Please use a list of CANSIM vectors or a vector.")
  }
}
df <- query(vectors.cpi, date.start, date.start)
write_xlsx(df, path="StatCan4R\CPI-all-items.xlsx") # export data
```

Conclusion

I have shown how to retrieve, manipulate and clean the data from StatsCan using three libraries: `cansim`, `tidyverse`, and `writexl`. The Data Enthusiast's Blog and the `cansim` package reference guide have been been very helpful to me in creating this demonstration. Links to relevant webpages are attached in the References section.

References

Baranovsky, Petr. 2019. “Working with Statistics Canada Data in R, Part 2: Retrieving Cansim Data.” https://dataenthusiast.ca/.

Ooms, Jeroen. 2020. *Writexl: Export Data Frames to Excel 'Xlsx' Format*. https://CRAN.R-project.org/package=writexl.

von Bergmann, Jens, and Dmitry Shkolnik. 2020. *Cansim: Functions and Convenience Tools for Accessing Statistics Canada Data Tables*. https://mountainmath.github.io/cansim/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. “Welcome to the tidyverse.” *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.