Automated Standard Error (SE) Calculation with PUMS Data with Excel Pivot Table

Toshihiko Murata

American Community Survey (ACS) data comes in many different formats. Public Use Microdata Sample (PUMS) data is one format available that is good for creating custom tabulations that are not the part of predefined ACS tables. In my work, for example, I need the count of persons 16 years or older, currently not attending school, and without a secondary education credential.

There is one complication, however. When creating custom tables using PUMS data, standard errors have to be calculated on your own. The formula for calculating SE for PUMS data using replication weight is $\mathrm{SE} = \sqrt{\frac{4}{80}\sum_{r=1}^{80}(\mathrm{X_r} - \mathrm{X})^2}$. It is not difficult, but it is repetitive and tedious.

Luckily, computers excel at doing this kind of repetitive and tedious task. Microsoft Excel, the commonly accessible computer software, is a good option for this task. It lets users create custom tables easily, and it can automate the SE calculation by using Excel macro (Visual Basic for Application (VBA)), which is already written and shared here.

{Demonstration} sample data file **http://tinyurl.com/omfhqsa**

ACS PUMS data

Data availability

- Census (http://www.census.gov/acs/www/data_documentation/pums_data/)
- IPUMS (Minnesota population center)
- Data Ferret

I prefer using IPUMS or Data Ferret because I can choose only the variables I need for the analysis, keeping the file size relatively small. There are various documents and video tutorials available online on how to use IPUMS and Data Ferret.

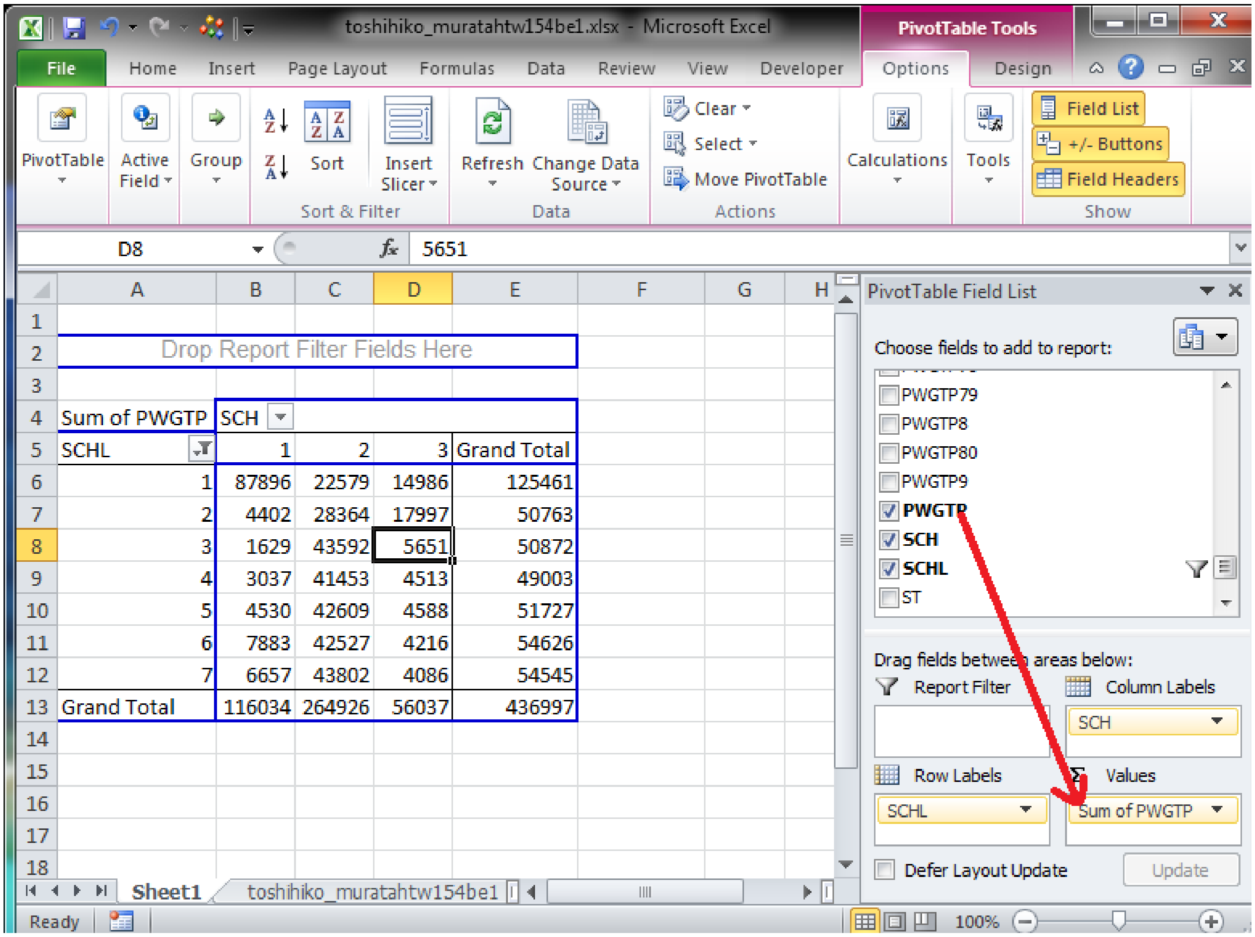
Download variables of interest, plus main weight and replication weights

- Main Population weight (PWGTP)
- Main Household weight (WGTP)
- Replication weights consist of 80 separate variables for each type of weight (PWGTP1 through PWGTP80 for population, and WGTP1 thorough WGTP80 for household)

File size can easily become very large because you need to get 81 variables for weights alone. An answer to “How big is too large for Excel Pivot table?” depends largely on the available memory of the computer used in the analysis. My work computer with 16GB of installed memory starts to complain with a database containing 100 variables and 1.7 million rows. The dataset, 2013 ACS 1Year Estimate for the entire U.S. population has a little over 3 million rows of records. This is too big to use in a raw data format. So if I need to use an entire file, I need some kind of work around.

Building the Pivot Table

The first step is building a pivot table with the downloaded data. Select the “insert” tab from the ribbon, then click on the pivot table button to start the pivot table dialog. If you have the data file open, that data will be selected as the default. Pivot tables also work with external formats such as MS Access MS SQL Server tables.



D8 5651

Drop Report Filter Fields Here

Sum of PWGTP	SCH			
SCHL	1	2	3	Grand Total
1	87896	22579	14986	125461
2	4402	28364	17997	50763
3	1629	43592	5651	50872
4	3037	41453	4513	49003
5	4530	42609	4588	51727
6	7883	42527	4216	54626
7	6657	43802	4086	54545
Grand Total	116034	264926	56037	436997

☑ PWGTP
☑ SCH
☑ SCHL

Column Labels: SCH
Row Labels: SCHL
Values: Sum of PWGTP

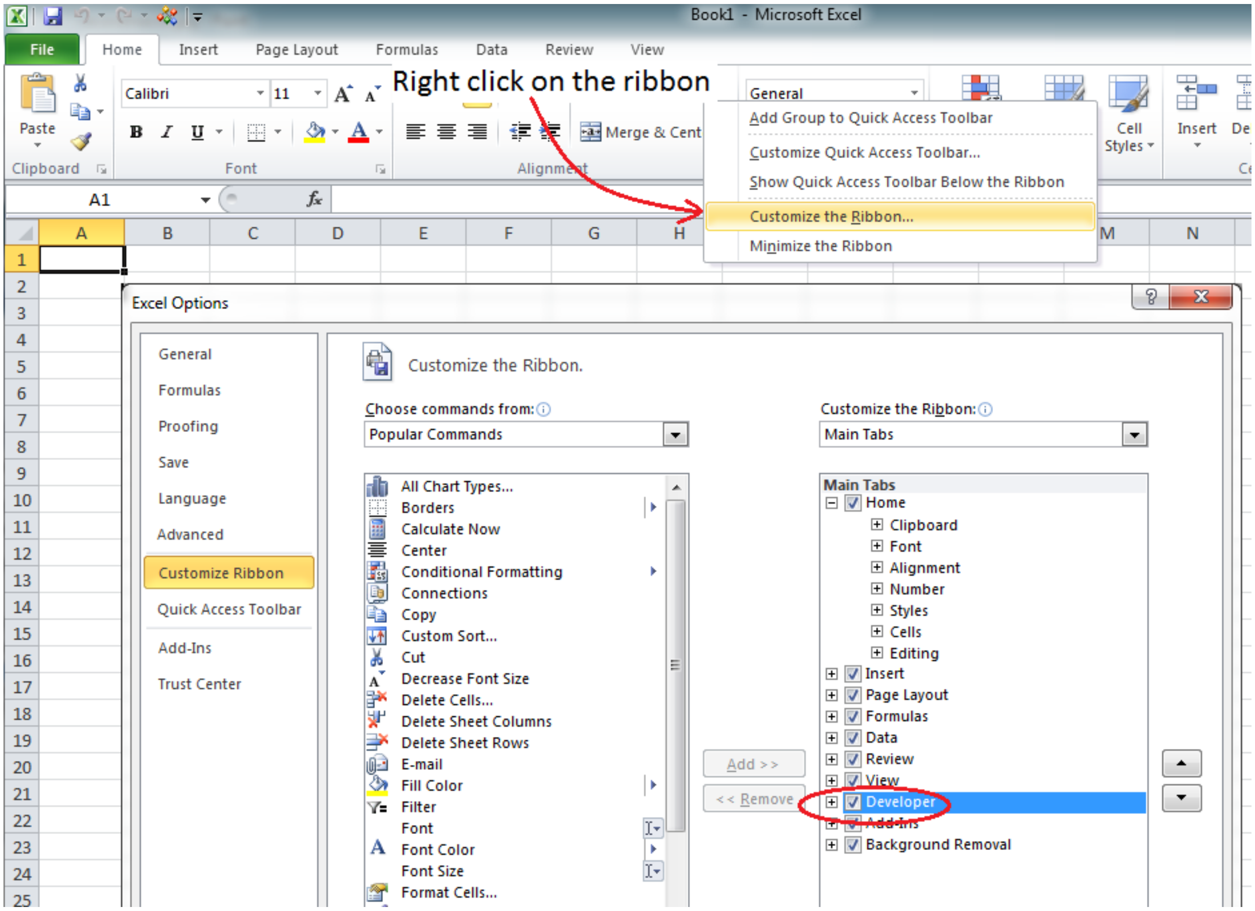
Once data is selected, click OK. The rest is simple --drag and drop variables into columns, rows and report filter areas of the pivot table--process. PUMS data is weighted (each case represents multiple cases), “values “or “pivot table data field” must be set to sum of main weight variable (PWGTP for population, WGTP for households).

Visual Basic Code for Calculating Standard Errors

Visual Basic for Application (VBA) for Excel

Excel Visual Basic for Application is an application development environment that is particularly suited for data and statistical work. It is full of useful data and statistics-related "objects" in this object oriented programming environment. For example, "Pivot table" is one of the Excel objects and is used here. This object is already fully developed and defined, so by using code, we are making this object do useful things and capturing the results.

Interaction with VBA code takes place in a separate "Visual Basic for Application" window. The easiest way to get this window is from the ribbon. The buttons to start this window is hidden by the default. To show this button, use" Customize the ribbon" dialog, and check the "developer" option. This will add the "developer" tab on ribbon.



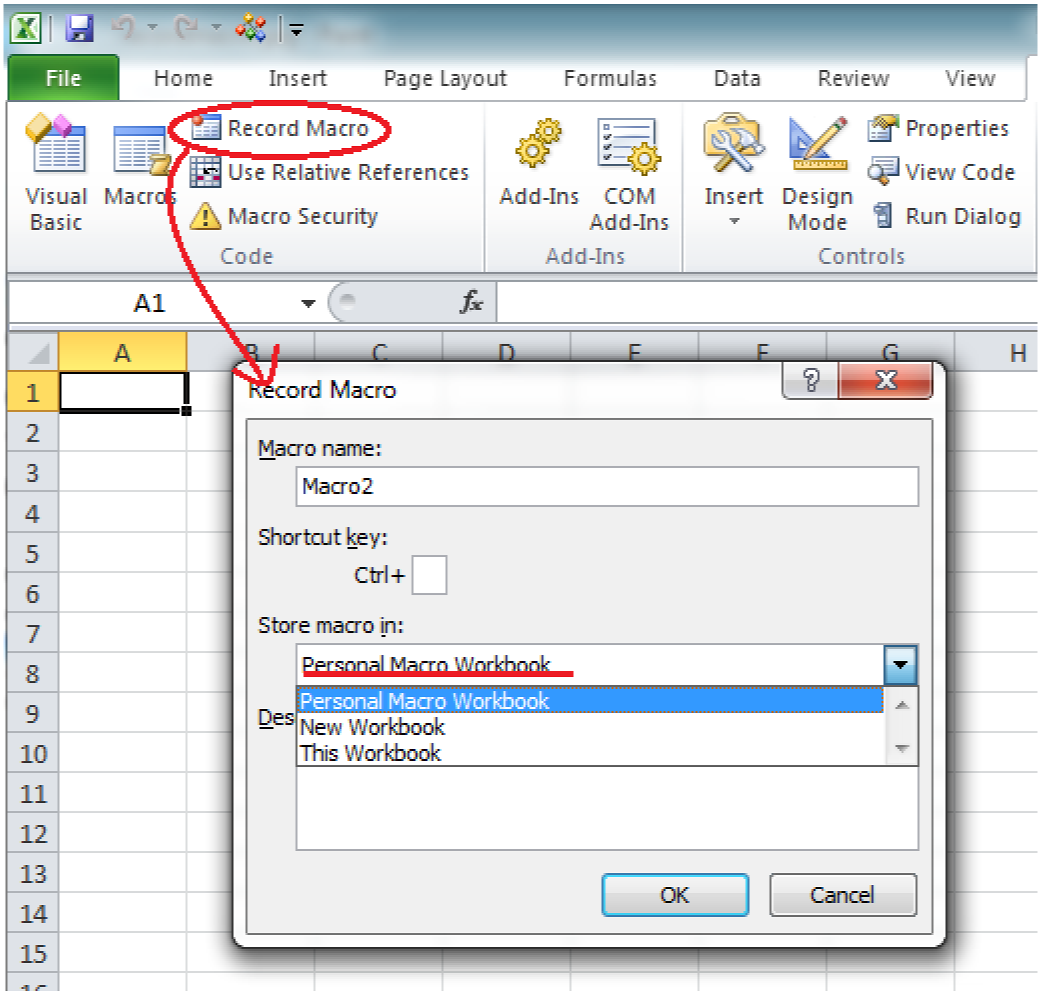
Right click on the ribbon

Locations to Store Visual Basic Codes

The VBA Codes can be saved in an individual workbook, or in the default VBA code storage called "Personal Macro Workbook (personal.xlsb)". One advantage of storing codes here is that once this workbook is created, codes in this workbook are always available anytime you start Excel.

The easiest way to create this workbook is by “record a macro”. When a macro is recorded and saved, Excel will create and save the personal macro workbook with the proper name and the location (C:\Users\<username>\AppData\Roaming\Microsoft\Excel\XLSTART\personal.xlsb).

To record a macro, go to the developer tab on the ribbon, and click the record macro button (see below).



Macro name: Macro2

Shortcut key: Ctrl+

Store macro in: Personal Macro Workbook

In the record macro dialog box, select personal macro workbook option from the dropdown list, and click OK. Then do something (like selecting a different cell). Click on ■ Stop Recording to stop. Open VBA window by clicking the “Visual Basic” button on the “Developer” tab. The code just created from recoding is saved under VBA Projects (“prosonal.xlsb”), Modules – Module 1, sub Macro1().

ACS_se.txt To use the standard error calculation code or sub procedure (Sub get_se()) , simply copy and paste the code into the modules window, below the End Sub statement. Each VBA sub procedure is marked by Sub and End Sub pair.

To run this code, chose “Run” from the menu or click the “Run” button ▶ on the tool bar while in VBA project editor window, while you have the pivot table open in the main Excel window. The code can be also run on the quick access tool bar, or the ribbon by assigning the code to button that you create.

Detailed Look at the VBA code

Open the attached Word document for detailed explanations.

Microsoft Word
Document

Importance of Checking SE When Using PUMS data

With Excel, it is very easy to drill down to finer and finer subgroups. It is important to keep in mind that SE becomes larger as the grouping becomes finer. Because of this, checking SE frequently along the way is very important.

{Demonstration – add sex, race and puma sequentially and show what happens to SE}

{Demonstration – back out, use grouping to reduce SE}

Exercise

Create a Procedure that Calculates Household SE

- Copy and paste the sub procedure, get_se(), into the same module.
- Rename the sub procedure to get_se_hh (or something logical).
- Replace PWGTP with WGTP (see highlighted area in the annotated code)
- Optional: rename variables in the code where appropriate

Original Word Document and sample data at http://tinyurl.com/omfhqsa