Ben Stabley – GEO242
Final Project
2023-06-16

Overview

I created a toolbox named `CensusTools` which contains 4 tools. Two of these tools perform useful geoprocessing and data acquisition work, while two perform housekeeping on ancillary data files the toolbox creates and uses.

This toolbox makes it significantly easier to obtain and apply data from the US Census Bureau to feature classes containing FIPS codes. The tool *Import census CSV* is useful when one has a CSV of a "table" obtained from https://data.census.gov. It does the annoying "data wrangling" that GIS students (and practitioners?) are familiar with. The *Add data to census geography* tool allows one to browse and select specific variables or groups (tables), or a combination, for a specified year and dataset (eg. ACS5 detailed tables). The tool acquires data from https://api.census.gov, and requires and internet connection. Both tools create a new feature class containing the input feature class (census geographies) joined with the data.

To increase operation speed and reduce redundant calls to the census API, the tool maintains a *cache* of "metadata" for datasets. This metadata is used by the tools to enhance the tool user interface in the geoprocessing pane by populating widgets used to select datasets, variables, and groups. It is also used to "wrangle" the data into correct data/field types and to remove non-data columns, should the user select the option. The *View cache* and *Purge cache* tools operate on this cache by letting the user view and delete items in the cache.

Included test data

I have included in the folder `test_feature_classes` a number of feature classes containing census geographies for various years and at all 5 target geography levels: state, county, tract, block group, and block. Most of the feature classes are in the geodatabase `motherlode.gdb`, although I have also included a Shapefile and a Geopackage formatted feature class, named `county_2010_OR_all.shp` and `block_2022_OR_WA.gpkg`. In all feature classes, the naming convention used is `LEVEL_YEAR_DESCRIPTION`. Some of the feature classes are straight copies from original geodatabases downloaded from the Census Bureau, and some have had simple geoprocessing (select, merge) performed in order to create subsets and supersets as one would during a full analysis workflow.

I have also included in the folder `test_csv_datasets` a handful of CSV files from data.census.gov for use in testing the *Import census CSV* tool. These are unaltered other than removing them from the ZIP archive in which they are obtained. There are 4 with various ACS datasets for all US counties and 1 with a decennial census dataset for all blocks in Multnomah County.

Tool details

The toolbox is found in the `tool` folder. The entirety of the code is in `CensusTools.pyt`. I tried to liberally comment the code, for your sake and for the sake of my future self. ArcGIS tool metadata (which provides helpful tooltips in the geoprocessing pane) is found in this folder as well. Instead of using the script tool utility to configure a tool's parameters using the ArcGIS GUI, I decided to create the entire toolbox in code.

Import census CSV

1. CSV file: Select a file containing census data. The tool should allow you to select only files ending with `.csv`. The CSV **must** contain the 2nd row with each variable's label (that is, "junk" like `Estimate!!Number!!HOUSEHOLD INCOME BY RACE AND HISPANIC OR LATINO ORIGIN OF HOUSEHOLDER!!Households`)
2. Year: The year or "vintage" of the data. The tool will attempt to find a year in the CSV's filename and auto-fill this parameter for you.

3. Dataset: The dataset (eg ACS5 detailed tables) from which the CSV's data was taken. The tool will attempt to determine a dataset from the CSV's filename and auto-fill this parameter for you.
4. Geographies: The input feature class with census geographies that match or are a subset of the geographies of the CSV data. The feature class must have a field named GEOID containing the FIPS code.
5. Output feature class: The name of the feature class to create with data from the CSV.
6. Option: Remove annotations: Will remove columns from the data that end in "A", such as the annotation for the estimate and annotation for the margin of error. *Default is Yes*.
7. Option: Remove non-annotations: Will remove columns from the data that are variable *attributes* (in census parlance) but are not annotations (ending in "A"). The prime example is Margin of Error columns for each estimate, ending in "M".
8. Option: Tool verbosity: Adjusts the amount of messages the tool outputs to the message window while running.

Add data to census geography

1. Geographies: The input feature class with census geographies for which you want to find data. The feature class must have a field named GEOID containing the FIPS code.
2. Year: The year or "vintage" of the data. The tool will attempt to find a year in the geography filename and auto-fill this parameter for you.
3. Dataset: The dataset (eg ACS5 detailed tables) from which you want to obtain data. The tool will attempt to determine a dataset from the geography filename and auto-fill this parameter for you.
4. Variables: *optional* Select individual variables from the dataset to get for each unit of the census geography. This is useful if you want to "cherry pick" variables. Unfortunately, choosing a variable here **will not** include its *attributes* such as annotations and margin of error.
5. Groups: *optional* Select groups (aka tables) of variables from the dataset to get for each unit of the census geography. This **will** include *attributes*, though they can be removed using tool options.
6. Output feature class: The name of the feature class to create with data.
7. Option: Remove annotations: Will remove columns from the data that end in "A", such as the annotation for the estimate and annotation for the margin of error. *Default is Yes*.
8. Option: Remove non-annotations: Will remove columns from the data that are variable *attributes* (in census parlance) but are not annotations (ending in "A"). The prime example is Margin of Error columns for each estimate, ending in "M".
9. Option: Tool verbosity: Adjusts the amount of messages the tool outputs to the message window while running.
10. Option: Cache data: If selected, the tool will put the raw census api result into the tool's cache. If the tool is run again **with the same Year, Dataset, Variables, and Groups parameters**, the tool should be able to use the cached data instead of downloading it again from the census api. This could be useful if the tool is run again and again with the same parameters, perhaps while testing a workflow in "mode builder" for example.

View cache

This tool has no parameters. A table of items in the cache is shown in the tool messages.

Purge cache

1. Age: Items as old or older than this number of days will be deleted (default of 0).
2. Tag: *Metadata* about census datasets, variables, and groups are tagged with `meta`. Data cached when using the cache option for *Add data to census geography* is tagged with `data`. `*` will remove all items from the cache.

Age and Tag are combined with boolean AND. The number of items deleted is shown in the tool message.

Known issues

One large issue is that only a subset of geography levels are supported. Due to the manner in which FIPS codes are evaluated, it is possible that a FIPS level could be misinterpreted and the *Add data to census geography* tool could attach incorrect data to the input feature class.

The tools have particular requirements for what fields are and are not in the input feature class. Specifically, `GEOID` is required, though in in some feature classes the field containing the FIPS number is named something else (for example, it is `GEOID10` in `county_2010_OR_all.shp` in the supplied test geographies). The tool currently has no way to specify the field in input feature class, and will fail if it does not find `GEOID`. Furthermore, if the input feature class contains any fields with a name that is or will be in the census data to join, the operation will fail. Likely conflicting names include: `state`, `county`, `tract`, `block_group`, `block`, and `fips`.

When using *Add data to census geography* with datasets that contain a lot of variables and groups, such as ACS5 detailed tables, the tool requires a "long" time (10-15 sec) to reach the stage where it begins to execute the actual script code. Because this does not happen with smaller datasets, I suspect that it has something to do with "internal validation" done on these large lists of options by the tool within Esri/ArcGIS/arcpy's internal processes. For example, the ACS5 detailed tables for 2020 contains 27889 individual variables, and the ACS5 subject tables contain 18810 variables. In contrast, the 2020 decennial census (PL) dataset contains only 335 variables.

I noticed some other odd behaviors in the tools' geoprocessing pane GUI. For example, after changing the year parameter, the dataset parameter's list of choices does not update to reflect the valid choices for the new year until the user clicks somewhere in the geoprocessing pane's background. I suspect this has something to do with the GUI not "knowing" the parameter was updated in some circumstances. There are some similar odd glitches when selecting variables and groups in *Add data to census geography*.

Future

As mentioned in the final project proposal, if this tool seems useful, I hope to develop it further and make it available to the wider GIS community. To that end, these are some thoughts I have for future improvements. I also welcome suggestions from GIS practitioners for tools that would be helpful, as well as people testing the tools "in the wild" and providing feedback.

- Support all census geography levels, such as AIANNH area, zip code tabulation areas, school districts, congressional districts, voting districts, county subdivisions, urban areas, census designated places, and so on.
- Improve tool robustness, input validation, error handling, and messaging. For example, the tools could check that the geography level of the input feature class matches that of the CSV data or one of the available levels for the chosen dataset.
- A tool (or option in the existing tools) to rename fields to be more descriptive. I wanted to do such a thing for this project, but after getting to know the census data better, felt that initial idea was not going to provide the experience I had hoped. Now, I am thinking about some sort of "mass rename" tool. It would be nice to use the official variable labels (as seen in the *Add data to census geography* tool's variable list) for field names, but they are universally poor, undescriptive, excessively long, and probably have illegal characters.
- A tool to download census geographies from the Census Bureau FTP site. This tool could use the cache to keep a pristine copy on hand for future use (unless deleted using *Purge cache*) and also avoid having to download the file repeatedly from the census website.
- Organize the tool "build" process so the code is modularized while in development, but is packaged and distributed as a single file. Also, make the code and project available on GitHub.