**Population analysis**

A sample comma separated value (csv) file, containing 2010 U.S. Census data is provided. The csv contains attributes on resident age and gender population by Census tract. In its current state, the csv file can’t be automatically joined with spatial data in ArcGIS because the table does not conform to the required format. A GIS professional has been repeatedly performing the process described in this report, by hand and has requested a tool to perform these steps automatically to streamline the workflow. Sample data is provided (2010 US Census Data csv file and a North Carolina tract Shapefile). Figure 1 shows a screenshot of a portion of the raw csv file which needs to be modified.

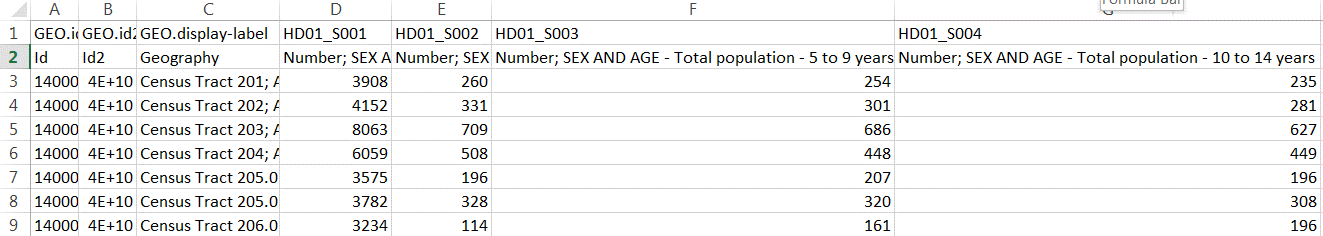


Figure 1: A small portion of the 2010 Census Data csv file.

This application should be able to modify the given csv file (or files like it) and create corresponding Shapefiles and reports. The csv files from this source don’t fulfill the requirements for ArcGIS table formats. The field names are not in the first row and they contain spaces and special characters and exceed the length requirement. The application needs to remove the first row, modify some of the field names (see ‘ArcGIS Table Field Names’ note below) and create the following outputs:

* An edited csv file that is importable to ArcGIS, saved in the output directory and containing all of the original columns.
* A csv file that is importable to ArcGIS, saved in the output directory and containing only selected those columns pertaining to a user selected gender (female, male, or total) and only those row pertaining to the county selected by the user.
* Shapefile with selected category and county data.
* An HTML report for the input file. The HTML report have an appropriate title and should at least contain a county name and summary output for the selected counties, including average, total, minimum, and maximum statistics for the selected gender (depending on the user selected category of interest) and an image of the resulting map (the arcpy.mapping module allows you to export an image).

The resulting csv files, shapefile, and HTML files need to be saved in a new folder, designated by the user. Within ArcMap, a button needs to be created. This button will launch a GUI interface (created with a script tool), prompting the user for several pieces of input information. Allow the user to select:

* The csv file name or a directory containing multiple csv files for conversion.
* The name of a Shapefile containing the NC track numbers
* The output location of the new folder where the shapefiles will be saved.
* A census data category gender selection (FEMALE, MALE, or TOTAL)
* A list of one or more North Carolina counties.
* The full path to an mxd file (to be used as the template).
* Output file base name for output formats (e.g., if the user gives ‘myTest’ here the output will be: myTest.csv, myTestFull.csv, myTest.shp, myTestReport.html

The best way to approach this is to open the csv file, read the contents, modify the contents, and create an edited full csv that can be imported into ArcGIS. (Use the file open/read/close functions you learn in class, not the csv module.) To repair the field names, create shortened names such as, ‘MPop80\_84’ to stand for ‘Number; SEX AND AGE - Male population - 80 to 84 years’. Look for patters in the field names, so that you can accomplish this in 15 or less lines of code. Use a dictionary to store your 15 or less replacements. Create a new field named ‘County’, extract the county name from the ‘Geography’ field, and store the county name in the ‘County’ field. The new full csv should contain all these changes.

Next, a second csv with only the fields pertaining to the user gender choice and selected counties should be created. This will involve only write to file the desired columns (in a row by row fashion), so you’ll need to determine a way to keep track of which columns to write and only write those as you loop through each row. In fact, not all rows will be written to this output file, since you’ll only want to write those rows with the selected county names.

Use the input Shapefile to join this table with geographic features to create the output Shapefile containing only the select counties. This can be accomplished using the AddJoin (Management) tool and specifying the ‘join\_type’.

Next, you’ll need to calculate the statistics for the report. This can be accomplished in a couple different ways. One way is to use a cursor to pass through the csv that you’ve just created. Finally, create the HTML report.

Aim to complete the csv processing and shapefile generation once we go over file input/output in Python. You don’t have to have a button and GUI yet at that point, but your code should be handling the user input via the system arguments by then.

As you start the project, you will not yet know how to automatically add data to the map, how to write an HTML file, or how to make a button and a graphical user interface to get input from the user. But you’ll soon learn how to read and write text files in class. Get your code to create the full output CSV for this test file first. Next, get your code to also create the reduced CSV. The csv is a text file (with commas separating the column values). So the next step is to redo the csv file fixing that you did by hand with your script and put this together with the geoprocessing to create the shapefiles. Next, make your script flexible by using arguments instead of hard-coding the input variable, such as the field name row number and so forth. Finally, you’ll need to create the button, GUI, and output HTML.

**ArcGIS Table Field Names**

-- Make the field names are in the first row.

-- Follow these practices for field naming:

* Field names must start with a letter.
* Field names must contain only letters, numbers, and underscores. (replace any blanks or special characters with underscores).
* Field names must not exceed 64 characters.