Jacob Miller

Professor Sonstein

DSCI.623.01

18 October 2019

Midterm – Part 1 – Obtain, Cleanse, and Prepare Data

              For this data analysis, I have elected to use the *Global Terrorism Database* – referred to as “GTD” – (https://www.kaggle.com/START-UMD/gtd) and *Infant Mortality, Fertility, Income* – referred to as “MFI” – (https://www.kaggle.com/burhanykiyakoglu/infant-mortality-fertility-income) datasets used so far during class. Additionally, I am making use of the *World University Rankings* (https://www.kaggle.com/mylesoneill/world-university-rankings) dataset, specifically the “Educational Expenditure Supplementary Data” portion – referred to as “EES”. I am using this dataset over the one used in class because it includes data as recent as 2011, where as the “Public Education Expenditure as Share of GDP” only contains information through 1993. As explained below, I am strictly looking at years post-2000 in this analysis.

The goal of the data cleansing was to setup files with similarly formatted countries in the first column and subsequent columns capturing all available information post-2000. In terms of how the United States is viewed internationally, as well as how the United States views other nations, there were significant changes after to the September 11, 2001 attacks. One of the driving decisions in expanding the organization is stability within a country, which drastically changed in many countries after this event. For this reason, I have elected to look back roughly 20 years, to just prior to these attacks.

              I have chosen to process all the data in Bash, using many of the eBay tsv-utilities (https://github.com/eBay/tsv-utils). I was completely unfamiliar with Bash prior to this analysis, so this was an opportunity to learn while optimizing the data cleansing process by using several speedy processing tools. I wrote a short program to both cleanse the data and ask the user which pieces of information they would like to look at. It should be noted that this program was written in the MacOS environment, so some commands are specific to that setting.

Before running the program, the user must ensure the three raw “.csv” files are stored in the *Raw\_Data* directory. Upon starting the program, it first creates directories to store temporary files. Where possible, intermediate information is stored in arrays within the Bash script, but some temporary files are generated during execution. It next uses the *csv2tsv* command to convert the “.csv” datafiles to “.tsv” files, which can be used with the rest of the eBay utilities. Each of the three files are then processed separately.

The EES dataset required minimal preprocessing, and is handled first. The file has extra whitespace at the beginning of some lines, so this is stripped away with a *sed* command. Next, *tsv-filter* is used to select all public institutions. I elected to look at public institutions, as opposed to including private, as these better represent how public funds (e.g. GDP) are allocated. These filtered lines are piped into a *cut* command which selects the country and post-2000 columns. This data is now ready for viewing, so an *Education.tsv* file is redirected to the *Cleaned\_Data* directory.

The MFI dataset is processed next. This file appears to have been generated ex-United States, so commas are used to separate whole numbers from decimals. Therefore, a *sed* command is used to swap all commas with periods. It can also be seen that periods are not used elsewhere, so this doesn’t introduce confusion processing downstream.

Column headers in the MFI dataset are comprised of years with a single letter prefix indicating if the column represents (m)ortality, (f)ertility, or (i)ncome. The previous *sed* is piped into another *sed*  which removes the letter-prefix from all columns post-2000 (we disregard any other years). Note the “1” which limits *sed* to only looking at the header. Finally, a sequence of three *cut* commands is used to pull out the country column and the columns associated with mortality, fertility and income. Each of these is redirected to a separate file in the *Cleaned\_Data* folder.

The GTD file is the last to be processed. The goal in cleaning this file was to see how many successful events, as defined by the GTD, happened in each country in the timeframe we are interested in. With that in mind, the first step is to use *tsv-filter* to select all successful events. These rows are piped into *tsv-select*, which allows for selection and reordering of the columns of interest: country, year, and success.

Next, using *tsv-summarize*, rows are grouped by country and year, giving a count of unique combinations of country/year. This is piped into *tsv-filter*, giving us counts of terrorism events by country for every year post-2000. However, years run in a row format here, and we would like them in columnar format.

The next two commands are performed on both the country and year columns. First, *tsv-summarize* is used to generate a list of all the unique countries. However, this command automatically generates a header when used with the *--unique-values*, so it is piped into a *tail* command which selects all rows except the header. Finally, this is captured in an array and assigned to the variable *countries*. An identical command is used on the years column and assigned to the variable *years*.

With these two arrays, nested *for*-loops are used in combination with *tsv-filter*to create a single array of a country and number of events that occurred for each consecutive year. At the end of each iteration of the outside loop, the individual array is appended to a temporary file. Entries in a row are separated by spaces, so the final step is to use *sed* to swap whitespace characters with tabs. Using regular expressions and capture groups, we contain this command to only the specific instances we are interested in. This cleaned file is the last to be redirected to the *Cleaned\_Data* folder.

At this point, the data cleansing portion of the program is completed and separate, cleaned “.tsv” files have been generated for Education, Mortality, Fertility, Income, and Terrorism. Each file has countries listed in the first column, with subsequent columns showing relevant information in the years since 2000. The program now moves onto ask the user whether they would like to merge any of these files into a single file which allows them to look at country and year combinations of subjects (Education, Mortality, etc.) they are interested in.