Daina Andries (andriesd) SI 601 Final Project Report

Annual Immigration Rates for European Union Countries (2004-2013)

Motivation

Migration into the European Union has been reported to be steadily increasing in recent years. For this project, I was interested in comparing the immigration rates of a selection of western and central European Union countries to identify which of these countries experienced the most dramatic fluctuations in immigration rate over the ten-year period of 2004-2013. I also wanted to find out which countries have been experiencing the greatest annual influx of incoming immigrants relative to their total populations on average over this time period. I chose these years because I thought it would be interesting to see whether there would be significant changes in immigration rate among EU countries occurring in 2008, the year of the beginning of the global financial crisis. 2008 falls in the middle of this time frame, permitting a glimpse into the years immediately before and after the crisis.

Data Sources

Immigration rate is defined by the number of incoming immigrants per thousand people already in the population of a destination country in a given year (definition taken from Population Reference Bureau at http://www.prb.org/). Calculating immigration rates for a selection of European countries over a span of ten years required one dataset containing total population data for the individual countries of the EU over a decade and another dataset containing the number of incoming immigrants for each country per year over the same decade.

My source for immigration inflow data was OECD.Stat, a database maintained by the Organization for Economic Cooperation and Development, which contains an International Migrations Database with data for OECD member countries. My source for total population data was Eurostat, the European Commission website for statistics.

The OECD interface for the International Migration Database had a built-in query that returned data tables for OECD member countries (not limited to Europe) according to different variables. I selected the variable 'Inflow of foreign population by nationality' and the interface returned a table with inflow data for OECD member countries from 2000-2013. The variable 'Inflow of foreign population by nationality' was explained in a general note on population and scope of the available data (accessible by clicking the blue information mark next to the variable). The note explains that the numbers for incoming immigrants are obtained from population registers, number of permanent residence and work permits issued, and special surveys. I chose this source over other available immigration databases because it offered a table where the data for cross-country comparisons had already been normalized, rather than offering access to multiple tables with data for individual countries. I scraped the HTML code from the OECD page (https://stats.oecd.org/Index.aspx?DataSetCode=MIG#) that displayed the table showing 'inflow of foreign population by nationality'.

For the total population data from Eurostat, I downloaded a table containing annual total population data for all European Union countries from 2004-2015. The variables in the table were country, year, and total population. The population values in the table represented ‘the number of persons having their usual residence in a country on 1 January of the respective year’. This means that the number could include non-citizens, but the numbers still point to persons who were currently already registered as residents on the first day of the new year. This table was available for download in .xls format at this url. I chose to download the file as an .xls file with a short description of the data then converted it to a .csv.

Data Processing

My intended final products were to be two new tables in .csv format. The first table would show individual immigration rates for European countries by year, from 2004-2013. The second table would show the average annual immigration rate for each European country. My selection of European countries was largely determined by what data was available. Since I was interested in the change of rate over time, I decided to drop European countries that didn’t have complete inflow data. Fortunately, most of the European countries listed in my OECD dataset had complete data for those years. I was mostly interested seeing the results for western and central European countries, but if a complete set of inflow data was available for an eastern European country (which was the case for Poland and Estonia), I decided to include those countries as well.

To create this end product, I needed to merge the two datasets according to country name. Fortunately, the country names employed in both tables were consistently standardized (the names used were English language country names), so I didn’t have to incorporate a set of country codes into my code to facilitate matching the datasets by country name. Once the datasets were merged by country name within a common data structure, I would need to perform the rate calculation for each country by year and store the countries and their respective sets of rates in a list of lists. From my list of lists, I could take the sets of rates to compute the average rate for the whole decade by country.

The data variables within the two tables I had found were all necessary for my analysis: countries, years, total population counts for the first table (EC), and countries, years, and total inflow counts for the second table (OECD). The biggest challenge was cleaning the data, eliminating explanatory text, metadata, and footnotes within data cells (such as ‘e’ for estimates) that would interfere with converting the numbers into floats.

With the population data, when I printed the lines from the .csv file after opening it, I found the notational symbols that appeared next to data in the cells had their own position in the lists for the rows after I ran Python’s ‘split’ function on each line, so I could easily eliminate these symbols, using ‘continue’ while building a new list of lists containing population values for each country, which I stored in the variable ‘populations’. I later refined this list into a list of tuples named ‘euro_populations_numbers’, where each tuple contained a country name and a list of ten float values representing populations over the decade. This two-tuple format made it easy for me to convert this variable into a dictionary.

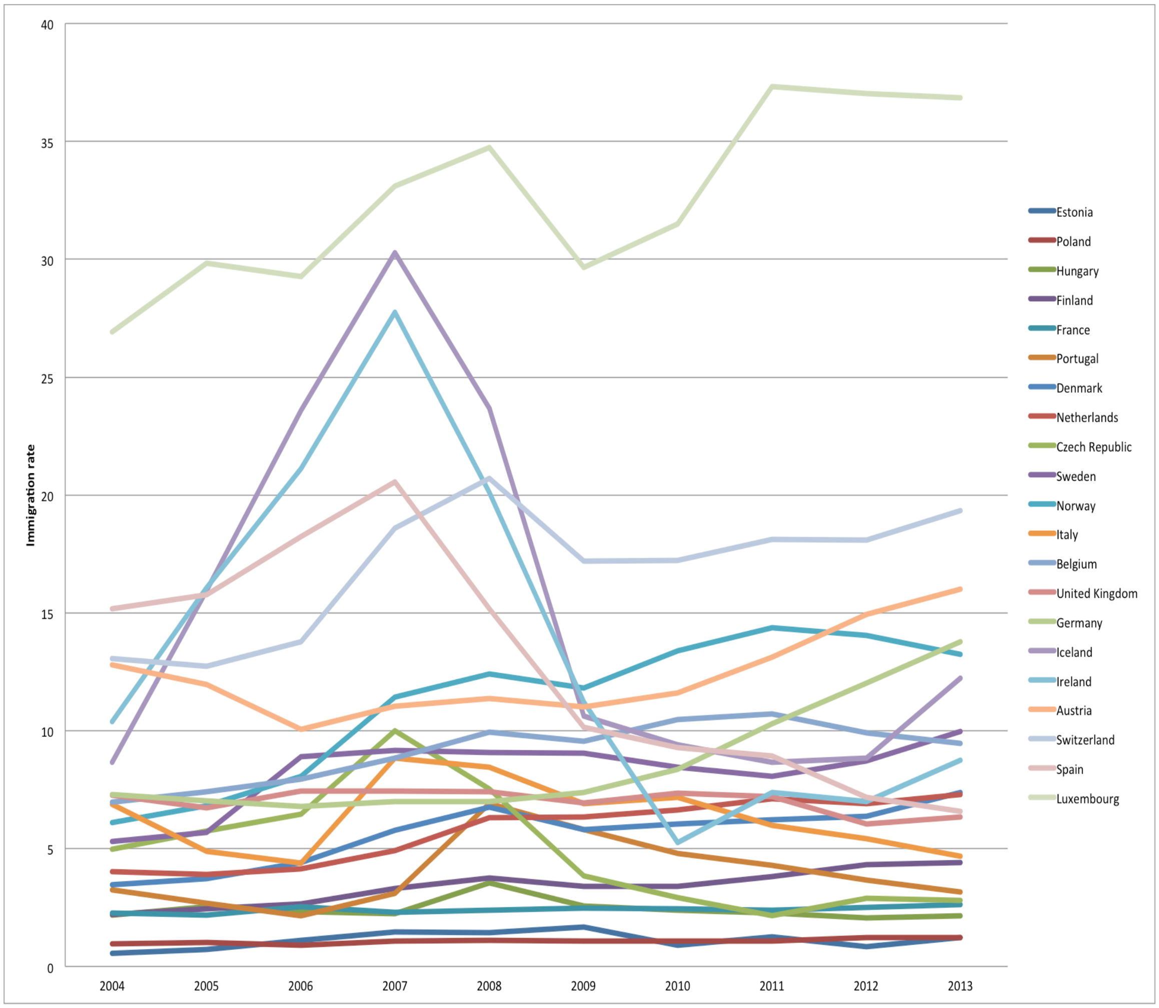
I scraped the inflow data in HTML format from the online OECD interface and saved it as an HTML file locally using Python’s urrlib2 and bs4 modules. Producing the final list of lists that I wanted took me a few iterations. First, I extracted the tabular data using BeautifulSoup’s ‘find_all’ function to find all ‘td’ elements, and eliminated all elements containing explanatory notation (which also contained duplicate values for the inflow number appearing next to the notational symbol). I used a regular expression and the regex ‘match’ function to pull and drop duplicate inflow numbers tagged with symbols. I converted the text strings from Unicode to UTF-8 strings. I stored the country names and inflow values in a list called ‘migration_inflows’. The problem with this initial list was that it was flat, whereas I wanted a two-dimensional list, where each inner list would contain the name and data for each country. I knew the interval by which I needed to group the data would be consistent, since I hadn’t yet dropped any placeholders for rows of incomplete data. The length of each row was 16 (country name plus inflow counts for 15 years). I found a function on StackOverflow that would chunk data according to a fixed interval. I named this function ‘group’ in my code and recreated my ‘migration_inflows’ list as a two-dimensional list, ‘immigration_raw’. With the new list, I dropped unneeded data from years that were out of my project’s scope, replaced UTF-8 code points for whitespaces with whitespace, and dropped all empty cell placeholders, converted all numbers to floats and created an ‘immigration_data’ list. This list was finally converted into a ‘transfer’ list, a list of tuples containing each country’s name and a list with that country’s inflow values.

Once I had cleaned both datasets, I converted the ‘euro_populations_numbers’ list into a dictionary. To build a new dictionary, I wrote a loop that iterated through the list of inflow values stored in the ‘transfer’ variable, checking to see if the first element in each tuple matched any of the dictionary keys. If there was a match, the inflow counts and the population values would be zipped together into a single list, where each tuple corresponded to a given year, and the first element was the inflow count and the second element was the population count. These lists were stored in the new dictionary ‘dct’ as values with the country names as their keys. From there it was easy to calculate the immigration rates for each country by year and store the results in a new lists of lists named ‘immigration_rates’, the contents of which were written to an output .csv file. Finally, with the lists stored in the variable ‘immigration_rates,’ I looped over each list and computed the average annual immigration rate for each country from 2004-2013, storing these values in the two-dimensional list ‘average_rates’. These results were written to a second .csv file.

Visualizations and Analysis

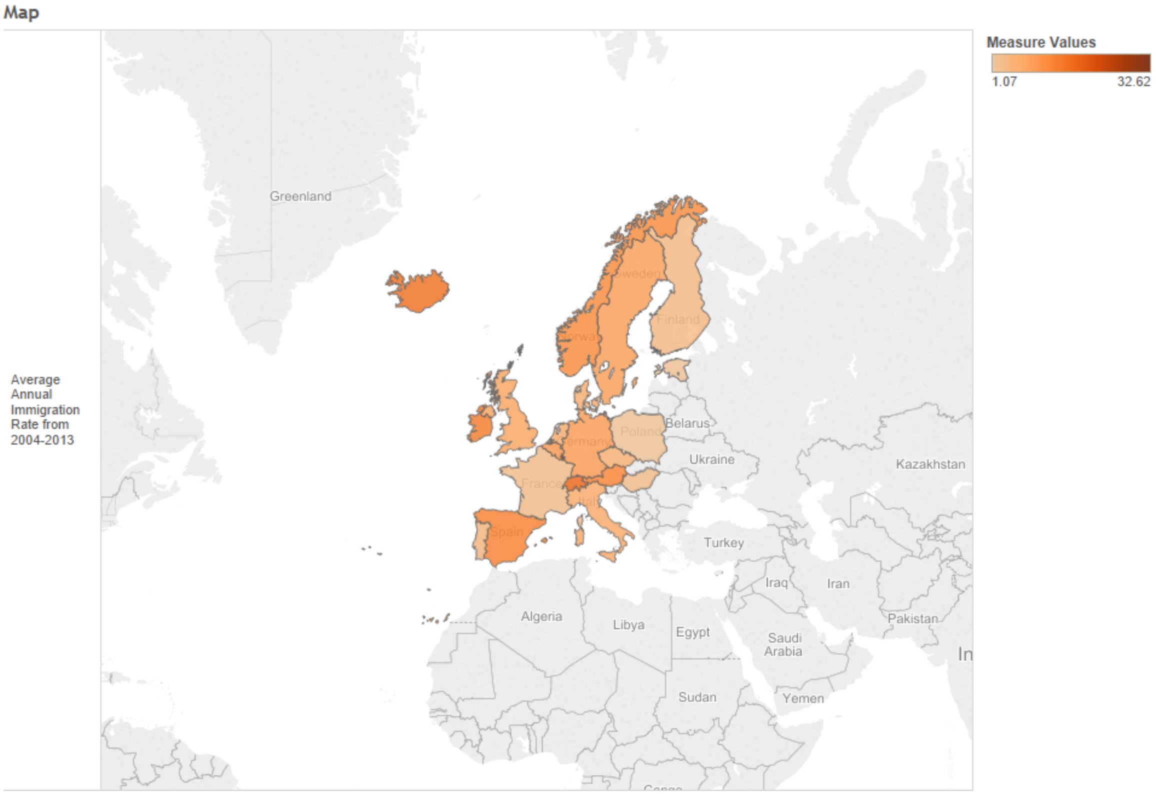
Once the code had run, the final products were two spreadsheets with immigration rate data for twenty-one European countries. The part of my source code that enabled my analysis can be found in the last two sections, where I compute immigration rates for each country by year and then compute average immigration rates for each country for all ten years combined. My analysis of this data product also depended on visualizing this data, since it was hard to immediately spot the trends in tabular format.

To visualize the first dataset, showing changes in immigration rate over time, I used Excel’s charting tools to generate a line graph, where the X-axis represents time, the Y-axis represents immigration rate, and each country was assigned a uniquely color-coded line. I discovered it was extremely hard to create a readable graph with 21 lines, where each one required a unique color! This problem prompted me to go back and revise my source code: in line 136 of my code, I sort each list inside the list ‘immigration_rates’ by x[1], i.e., by the second element of the list, the first immigration rate (2004) for each country. I did this to make the graph easier to interpret: as the colors for the countries change going down the graph legend, the corresponding color of the starting point of each country's line changes going up the Y-axis (for example, Estonia's blue is at the very top of the legend, but Estonia's blue line starts at the bottom of the tangle of lines at point '2004', then Poland's red line, then Hungary, Finland, France, and so on).



Having addressed this issue of distinguishing the lines as best as I could, I examined the graph to identify which countries experienced the most dramatic changes in immigration rate. This country is Ireland. Immigration rates for Iceland and Ireland stand out from the rest, rising steeply and peaking in 2007 before falling fast over 2007-2010. Iceland's rate rose more than Ireland's, but Ireland's immigration rate fell harder after 2007 (the time of the financial crisis) than Iceland's rate either rose or fell. Spain's immigration rate also dropped dramatically in 2007. Germany was one of the few countries whose immigration rate did not drop in 2007, but rather continued to increase. Austria's immigration rate was declining prior to 2006 and climbed afterwards. Luxembourg had the highest rate overall over the whole decade. Switzerland's rate also remained among the highest, although both Switzerland's and Luxembourg's rates dropped noticeably in 2008-2009.

I used Tableau Public to visualize the average immigration rate data from the second table I created on a filled map:



On the map, the darkest oranges represent the highest average annual immigration rates. An interactive version of the map is available here. Luxembourg is the darkest orange, though it is difficult to see here because the country is so small. Other countries that are more darkly shaded are Iceland, Ireland, Switzerland and Spain. Poland is shaded the lightest, with an average annual rate of 1.07 immigrants per thousand inhabitants. After examining the map, I returned to the averages data table and used Excel's median formula to find that the United Kingdom's average was the median average immigration rate for this selection of EU countries. I was surprised to see that France's average is so low relative to most of the other countries of western Europe, but I think this result may have to do with the nature of the scope of the source immigration inflow data, which was limited to registration for residence and work permits.

There definitely appears to be a consistent pattern in immigration rate of a peak and drop around the time of the global financial crisis of 2007-2008, with the exception of a few countries such as Germany and Austria. A future analysis could involve comparing these patterns in immigration rates with shifts in some set of economic indicators during this decade to make a case for a correlation between these immigration rates and the financial crisis. Another interesting project would be to run a network analysis to look for patterns in migrations based on datasets with variables for countries of origin as well as destination for immigrants entering or moving around the European Union.