Yuhui Wang
INF1340 Programming for data science
Instructor Shion Guha
Midterm project write-up

UN migrant data cleaning mindset:
I clean the dataset based on each table and split 7 tables into 10 tables in total.

Problems and solutions in the dataset based on 5 principles

Principle 1: column names need to be informative, valuable names

- Numbers of year are used as column names in all 7 tables, such as 1990, 2000, and 1990-1995. These number should be put in cells as variables/values, not as column names.
- Gender data are also showed as column names, which are supposed to be in cells.

Principle 2: each column needs to consist of one and the only one variable

- The information of gender and year are mixed together within one column/cell in most tables. For example, the total population of male in 1990 is presented in a same cell. We should split these two types of data and put them in two columns.

Principle 3: variables need to be in cells, not rows and columns

- The numbers of year (1990, 1995, 2000..) are spread across columns.
- Gender data (both sexes, male and female) also are spread across columns.
- In ANNEX table, four types of region/country (developed region, least developed country, Sub-Saharan Africa, less developed region) are spread across columns.
- Above three types of data should be in cells, not columns.

Principle 4: each table needs to have a singular data type

- In table6, three types of data are mixed in one table, which should be separated. (including estimated refugee stock, refugees as a percentage of the international migrant stock, annual rate of change of the refugee stock)
- Type of data should also be pulled out and stored in one independent table, instead of repeating them in each table.

Principle 5: the same observational unit should not be spread across multiple tables

- Type of data are spread across multiple tables.
- In ANNEX table (classification of countries and areas), the classification of world (developed region, least developed country, Sub-Saharan Africa, less developed region) is not fully presented, and the last type of region is missed. Meanwhile, the code of four types of region/country is only presented in other tables, not in this table.

The process of cleaning data

Table 1—International migrant stock

1. **Step:** Read the second sheet (table 1) in the excel uploaded in the data file starting from 15th row. Print all column names to see what columns we have.
   Reason: From above column names, we can see that some columns are unnamed, violate principle 3, or are not accurate. So, we need to change the unnamed into

corresponding year numbers (same as what in excel) and transfer year numbers from column names to cells because year numbers are variables not variable names. Meanwhile, gender data, same as year data, are spread across columns. Therefore, we should separate gender from year data and establish independent columns for both gender and year.

2. **Step:** Select columns with both sexes data and columns that can be used as id variables ("sort order", "major area, region, country or area of destination", "country code") and create a new table named v1. Then rename the new table v1 columns with year numbers to fill inaccurate and unnamed column names. Print v1. Find NaN values and numbers with one decimal place, so drop NaN data and redefine the type of numbers.
   Reason: In order to create two new columns for gender and year, I first separate table1 into three parts: both sexes, male, and female, which corresponds three new tables v1, v2, v3. By doing so, I can rename the column names and put the numbers of years and gender into appropriate places. Thus, I first create v1, rename its column names, and print it. I find there are NaN values in table v1 and the numbers of sort order and country code are one decimal place, so I drop all NaN data and change number type to integer.
3. **Step:** Melt table v1 by identifying "Sort order", "major area, region, country or area of destination", and "Country code" as id variables, "Year" as variable name, "International migrant" as value name. Print v1 to check out.
   Reason: The melt function can pull the number of years from column names, put it into cells and establish a new column that only contains the number of international migrants. By doing so, we have independent columns for year and the number of international migrants.
4. **Step:** Create a new data frame c1 (a column) for gender, and all values under the gender column are "both sexes". Add this gender column into table v1. The combination of c1 and v1 is the final v1.
   Reason: Since v1 only contains data of both sexes, the values of gender columns are all same, that is, both sexes. I use for loop to fill each cell under gender column and concat function to combine v1 and c1 together. Thus, we have an independent column for gender data and a complete and clean international migrant table for "both sexes".
5. **Step:** Following will have the same manipulation to male and female data by creating new tables v2 and v3.
   Reason: In this way, I can clean table 1 by separating it into three parts.
6. **Step:** After getting clean table v1, v2, v3, I combine them together with concat function to get a new table 1, tb1.
   Reason: Since each table (v1, v2, v3) has same column names, I use concat function to pile up on a one table (tb1). Tb1 is also the first clean table that I get.

Table 2—Total population
Table 3—International migrant percentage
Table 4—Female migrant percentage
Table 5—Annual rate of change of the migrant stock

Step: they all have same steps as table 1, except different primary data (e.g. total population/international migrant percentage…).

Table 6—Estimated refugee stock
Table 7—Refugee percentage
Table 8—Annual rate of change of refugee stock

1. **Step:** I split the original table 6 into three tables: table 6, table 7, table 8.
   Reason: Because the original table 6 has three types of data: estimated refugee stock, refugee percentage, annual rate of change of refugee stock, which violates principle 4.
2. **Step:** these three tables have same cleaning process as table 1 until step 3, because they don't have gender difference. There is no need to create a new gender column.

Table 9—Classification of countries and areas (ANNEX)

1. **Step:** Read the 7th sheet in the UN excel starting from 15th row and print all column names of table ANNEX.
   Reason: By comparing with the excel version, it shows that one type of region (Less developed region) is not presented in the table and the types of regions spread across columns. Thus, I need to add "less developed region" data, pull region type out from column names and put them into cells.
2. **Step:** Select columns that includes three types of regions (Developed region, Least developed country, Sub-Saharan Africa) from ANNEX. Create an empty list c1. I use a for loop and if conditions to check the type of regions for each country and add four types of regions into the c1. Do the same for c2, which is the corresponding code of region/country types.
   Reason: This step cleans the data of region type and region code by establishing new columns for it.
3. **Step:** Select all columns in ANNEX except the type of regions columns and name this table as tb9. Name column c1 as "World" and c2 as "Code.2". Combine tb9 with c1 and c2 by adding c1, c2 as a new column of tb9.
   Reason: The type of regions and its code are the only dirty data. After adjusting the region type and code data, we get a clean table 9 (classification of countries and areas).

Table 10—type of data

1. **Step:** read the second sheet (table1) in the excel starting from 15th row and name it as table 10. Print all column names in the sheet. Select columns that can be used as id variables ("sort order", "major area, region, country or area of destination", "country code") and type of data column.
   Reason: Unlike above tables, type of data column is clean. The problem of it is that it is placed in wrong tables and should be placed in an independent table.
2. **Step:** Drop all NaN data and change "Sort order" and "Country order" data type into integer.
   Reason: After this, we get a clean table 10 related to type of data.