Tommy Wang
1340 Midterm Project
November 17, 2022

Introduction

Today, we live in the world where migrants and refugees have an significant impact on our life in terms of social, economic and political aspects. Thus, it is important to measure the scale and direction of the movement of these people in order to understand the related rising challenges and how the governments around the globe could deal with it. In this project, I will carry out a data cleaning process with a dataset related to migrants and refugees published by the United Nations. This excel file contains one content sheet with general introduction about the excel file, six sheets of tables with relevant data, one sheet of annex and one sheet of notes. I will mainly focus on the six tables with the help of the annex and the notes.

Estimates are presented for 1990, 1995, 2000, 2005, 2010, and 2015 and are available for 232 countries and areas of the world. Table 1 contains data about international migrant stock at mid-year. It is disaggregated by major area, region, country or area of destination, sex and year. Table 2 contains data about total population and disaggregated by major area, region, country or area of destination, sex and year. Table 3 contains data about international migrant stock as a percentage of the total population and is disaggregated by major area, region, country or area of destination, sex and year. Table 4 contains data about female migrants as a percentage of the international migrant stock and is disaggregated by major area, region, country or area of destination and year. Table 5 contains data about annual rate of change of the migrant stock and is disaggregated by major area, region, country or area of destination, sex and year. Table 6 contains data about estimated refugee stock at mid-year, refugees as a percentage of the international migrant stock, and annual rate of change of the refugee stock. It is disaggregated by major area, region, country or area of destination, and year.

Methods

Since there are multiple tables in the file, I start the cleaning by importing each table separately in the Jupyter notebook and work on them one by one. Following the importing, I observe that Table 1 to Table 6 all contain a few rows that are not necessary in the data cleaning process so I removed them by adjusting their indexes and headers. Next, in Table 1, 2, 3, 5 and 6, since there are many columns that needs to be cleaned, I separate them respectively into 3 sections. For Table 1, 2,3 and 5, I separate them based on sex and for Table 6, I separate them based on the type of variables. Table 4 contains less columns so I did not break it down into different parts. Next, I renamed the column names.

All these tables have values of year spread across column headers so I use melt function to create a new column for year. After melting, I notice there are two variables in the column head, which is the population and sex. Hence, I separate them into two columns using melt function and stylize the variable Sex. Afterward, I combine all the sections and now all the columns with multiple variable values are corrected except the major area, region, country column. Hence, I create multiple list for major area, region and country and use them to add the following separate new columns: major area, region, country. Next, I notice there are 5 variables in the column of region and country, which includes *Developed regions, Developing regions, Least developed countries, Less developed regions excluding least developed countries* and *Sub-Saharan Africa*. I also found there are information about *Developed region, Least developed*

country, and *Sub-Saharan Africa* in the Table Annex. Hence, I decided to add these three variables to the table but remove *Developing regions* and *Less developed regions excluding least developed countries* since there were no extra information provided in the Table Annex.

Now, all the tables have been cleaned properly. I can merge different tables with the same observational unit. I merge Table 1, 2, 3, 4, 6.1, 6.2 into one table and Table 5 and 6.3 into another. This is because although all the tables have the same observational unit, the variable of "year" is captured differently: the former one is timeframe and the latter is duration. In the last step, I notice there are two observational units in these two tables, which is the country and various migrant/refugee data. I then divide the two tables respectively into two sections: one contains the country and the other contains migrant/refugee data, which then gives me four tables in total at the end. In the next section, I will further explain the cleaning process using the 5 principles.

5 Principles

Principle 1: column names need to be informative, variable names and not values

A common problem of messy data is that headers are values instead of variable names. The tables in the UN dataset all have this problem. Here, I will use Table 1 as an example to illustrate. Table 1 explores the relationship between the population of the international migrant stock and the destination country. Variables related to Principle 1 is the year (1990, 1995, 2000, 2005, 2010, 2015) as illustrated in UN.Table 1. To tidy it, I need to turn columns into rows by using the “Melt” function in Pandas as illustrated in UN.Table2. The new variable column is named “year”, and the value column renamed to “international migrant stock at mid-year”. This method is applied to all the tables in the dataset, for example, UN.Table 3 and 4 for Table 2 in the dataset.

UN.Table1 Before melting

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	Type_of_data(a)	1990	1995	2000	2005	2010	2015
0	NaN	NaN	NaN	NaN	NaN	1990	1995	2000	2005	2010	2015
1	1.0	WORLD	NaN	900.0	NaN	152563212	160801752	172703309	191269100	221714243	243700236
2	2.0	Developed regions	(b)	901.0	NaN	82378628	92306854	103375363	117181109	132560325	140481955
3	3.0	Developing regions	(c)	902.0	NaN	70184584	68494898	69327946	74087991	89153918	103218281
4	4.0	Least developed countries	(d)	941.0	NaN	11075966	11711703	10077824	9809634	10018128	11951316

UNTable2 After melting

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	Type_of_data(a)	year	International_migrant_stock_at_mid-year(both_sexes)
0	1.0	WORLD	NaN	900.0	NaN	1990	152563212
1	2.0	Developed regions	(b)	901.0	NaN	1990	82378628
2	3.0	Developing regions	(c)	902.0	NaN	1990	70184584
3	4.0	Least developed countries	(d)	941.0	NaN	1990	11075966
4	5.0	Less developed regions excluding least develop...	NaN	934.0	NaN	1990	59105261

UN.Table 3 Before melting

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	1990	1995	2000	2005	2010	2015
0	NaN	NaN	NaN	NaN	1990.000	1995.000	2000.000	2005.000	2010.000	2015.000
1	1.0	WORLD	NaN	900.0	5309667.699	5735123.084	6126622.121	6519635.850	6929725.043	7349472.099
2	2.0	Developed regions	(b)	901.0	1144463.062	1169761.211	1188811.731	1208919.509	1233375.711	1251351.086
3	3.0	Developing regions	(c)	902.0	4165204.637	4565361.873	4937810.390	5310716.341	5696349.332	6098121.013
4	4.0	Least developed countries	(d)	941.0	510057.629	585189.354	664386.087	752804.951	847254.847	954157.804

UN.Table 4 After melting

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	year	Total_population_of_both_sexes_at_mid-year(thousands)
0	1.0	WORLD	NaN	900.0	1990	5309667.699
1	2.0	Developed regions	(b)	901.0	1990	1144463.062
2	3.0	Developing regions	(c)	902.0	1990	4165204.637
3	4.0	Least developed countries	(d)	941.0	1990	510057.629
4	5.0	Less developed regions excluding least develop...	NaN	934.0	1990	3655147.008

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

After melting the year values, the column contains a combination of two variables, which is the population of international migrant stock and male and female as illustrated in UN.Table 5. I use the melt function again and UN.Table 6 shows the results after splitting the single column into two variables: population and sex. Next, UN.Table 7 shows the results of stylizing the variable "Sex".

UN.Table 5

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	Type_of_data(a)	year	International_migrant_stock_at_mid-year(Male)	International_migrant_stock_at_mid-year(Female)
0	1.0	WORLD	NaN	900.0	NaN	1990	77747510	74815702
1	2.0	Developed regions	(b)	901.0	NaN	1990	40263397	42115231
2	3.0	Developing regions	(c)	902.0	NaN	1990	37484113	32700471
3	4.0	Least developed countries	(d)	941.0	NaN	1990	5843107	5236216
4	5.0	Less developed regions excluding least develop...	NaN	934.0	NaN	1990	31641006	27464255

UN.Table 6

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	Type_of_data(a)	year	Sex	International_migrant_stock_at_mid-year
0	1.0	WORLD	NaN	900.0	NaN	1990	International_migrant_stock_at_mid-year(Male)	77747510
1	2.0	Developed regions	(b)	901.0	NaN	1990	International_migrant_stock_at_mid-year(Male)	40263397
2	3.0	Developing regions	(c)	902.0	NaN	1990	International_migrant_stock_at_mid-year(Male)	37484113
3	4.0	Least developed countries	(d)	941.0	NaN	1990	International_migrant_stock_at_mid-year(Male)	5843107
4	5.0	Less developed regions excluding least develop...	NaN	934.0	NaN	1990	International_migrant_stock_at_mid-year(Male)	31641006

UN.Table7

	Sort_order	Major area, region, country or area of destination	Notes	Country_code	Type_of_data(a)	year	Sex	International_migrant_stock_at_mid-year
386	122.0	State of Palestine	(12)	275.0	B	1995	Male	128328
2534	150.0	Norway	(16)	578.0	B	2005	Female	184700
2716	67.0	Saint Helena	(4)	654.0	B	2010	Female	257
3074	160.0	Holy See	(17)	336.0	I	2015	Female	426
1656	67.0	Saint Helena	(4)	654.0	B	1990	Female	161
906	112.0	Cyprus	(10)	196.0	B	2005	Male	50421
1225	166.0	Serbia	(18)	688.0	B	2010	Male	364654
873	79.0	China	(5)	156.0	C	2005	Male	379920
1492	168.0	Spain	(19)	724.0	B	2015	Male	2855179
2518	134.0	Republic of Moldova	(13)	498.0	B	2005	Female	97357

Another example is the variable called *"Major area, region, country or area of destination"* as you can see in UN.Table7. Here we can see there are three variables in this single column. In order to separate them, I create multiple list for major area, region, and country. Next, I create functions for region and major area, which labels each row based on the values of a column and indicates which region and major area a particular country belongs to. The result is illustrated in UN.Table 8

UN.Table 8

	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	year	Sex	International_migrant_stock_at_mid-year
0	1	WORLD	WORLD	WORLD	NaN	900	NaN	1990	Both	152563212
1	2	WORLD	Developed regions	Developed regions	(b)	901	NaN	1990	Both	82378628
2	3	WORLD	Developing regions	Developing regions	(c)	902	NaN	1990	Both	70184584
3	4	WORLD	Developing region	Least developed countries	(d)	941	NaN	1990	Both	11075966
4	5	WORLD	Developing region	Less developed regions excluding least develop...	NaN	934	NaN	1990	Both	59105261
5	6	WORLD	Sub-Saharan Africa	Sub-Saharan Africa	(e)	947	NaN	1990	Both	14690319
6	7	Africa	Africa	Africa	NaN	903	NaN	1990	Both	15690623

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

UN.Table 8 shows that the column of region and country have 5 variables in them, which is *Developed regions, Developing regions, Least developed countries, Less developed regions excluding least developed countries* and *Sub-Saharan Africa.* I decide to move these variables to their respective columns. However, it is important to note that since the Table Annex only provides information regarding *Developed, Least developed regions* and *Sub-Saharan Africa,* the information about *Developing regions* and *Less developed regions excluding least developed countries* will be dropped from the final data.

UN.Table 9 : I skip the first 6 rows that contain variables such as developed regions.

	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	year	Sex	International_migrant_stock_at_mid-year
6	7	Africa	Africa	Africa	NaN	903	NaN	1990	Both	15690623
7	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	1990	Both	5964031
8	9	Africa	Eastern Africa	Burundi	NaN	108	B R	1990	Both	333110
9	10	Africa	Eastern Africa	Comoros	NaN	174	B	1990	Both	14079
10	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	1990	Both	122221

UN.Table 10 : I selected specific columns in the Table Annex

	Country or area	Major area	Region	Developed region	Least developed country	Sub-Saharan Africa
0	Afghanistan	Asia	Southern Asia	No	Yes	No
1	Albania	Europe	Southern Europe	Yes	No	No
2	Algeria	Africa	Northern Africa	No	No	No
3	American Samoa	Oceania	Polynesia	No	No	No
4	Andorra	Europe	Southern Europe	Yes	No	No

UN.Table 11: Next, I combine Table 1 with selected columns from the Table Annex.

	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	year	Sex	Developed region	Least developed country	Sub-Saharan Africa	International_migrant_stock_at_
0	7	Africa	Africa	Africa	NaN	903	NaN	1990	Both	NaN	NaN	NaN	1569
1	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	1990	Both	NaN	NaN	NaN	596
2	9	Africa	Eastern Africa	Burundi	NaN	108	B R	1990	Both	No	Yes	Yes	33
3	10	Africa	Eastern Africa	Comoros	NaN	174	B	1990	Both	No	Yes	Yes	1
4	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	1990	Both	No	Yes	Yes	12

Principle 5: a single observational units must be in 1 table

After all the tables are cleaned, I combine the tables based on the same observational units. Here, all the tables share the same observational unit, which is the country. I could have combined all the tables into one. However, since the variable of year in Table 1,2,3,4,6.1 and 6.2 is captured as timeframe while it is captured as duration in Table 5 and 6.3 , I combine the former tables as one table (*merge1*) and the latter as another one (*merge2*).

UN.Table 12 : merging Table 1,2,3,4,6.1 and 6.2 (*merge1*)

	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	year	Sex	Developed region	Least developed country	Sub-Saharan Africa	International_migrant_stock_at_
0	7	Africa	Africa	Africa	NaN	903	NaN	1990	Both	NaN	NaN	NaN	1569
1	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	1990	Both	NaN	NaN	NaN	596
2	9	Africa	Eastern Africa	Burundi	NaN	108	B R	1990	Both	No	Yes	Yes	33
3	10	Africa	Eastern Africa	Comoros	NaN	174	B	1990	Both	No	Yes	Yes	1
4	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	1990	Both	No	Yes	Yes	12

Sex	Developed region	Least developed country	Sub-Saharan Africa	International_migrant_stock_at_mid-year	Total_population_at_mid-year(thousands)	International migrant stock as a percentage of the total population(thousands)	Percentage of the international migrant stock	Estimated refugee stock at mid-year	Refugees as a percentage of the international migrant stock_
Both	NaN	NaN	NaN	15690623	631614.304	2.48421	NaN	5687352	36.246821
Both	NaN	NaN	NaN	5964031	198231.687	3.008616	NaN	3168001	53.118453
Both	No	Yes	Yes	333110	5613.141	5.934467	NaN	267929	80.43259
Both	No	Yes	Yes	14079	415.144	3.391353	NaN	0	0
Both	No	Yes	Yes	122221	588.356	20.773307	NaN	54508	44.597901

UN.Table 13: merging Table 5 and 6.3 (*merge2*)

	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	year	Sex	Developed region	Least developed country	Sub-Saharan Africa	Annual rate of change of the migrant stock	Annual rate of change of the refugee stock_
0	7	Africa	Africa	Africa	NaN	903	NaN	1990-1995	Both	NaN	NaN	NaN	0.826734	0.076037
1	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	1990-1995	Both	NaN	NaN	NaN	-3.435412	-5.30801
2	9	Africa	Eastern Africa	Burundi	NaN	108	B R	1990-1995	Both	No	Yes	Yes	-5.355717	-3.390926
3	10	Africa	Eastern Africa	Comoros	NaN	174	B	1990-1995	Both	No	Yes	Yes	-0.199873	..
4	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	1990-1995	Both	No	Yes	Yes	-4.058465	-9.763426

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

The principle 4 states that messy data could contain values collected at multiple levels or on different types of observational units. After combining the tables and resulting in two tables, I notice they actually contain observations on two types of observational units: the country and the migrant/refugee data. This manifests itself through the duplication of facts about the country: *Sort_order, major_area, region, country, Notes, Country_code, Developed region, Least developed country,* and *Sub-Saharan Africa are repeated*. Therefore, each table needs to be broken down into two datasets: a country dataset which stores *Sort_order, major_area, region, country, Notes, Country_code, year, Sex, Developed region, Least developed country and Sub-Saharan Africa*, and a migrant/refugee dataset which gives the relevant migrant/refugee data for each country in each year. UN.Table 14 and 15 is the breakdown of *merge1* and UN.Table 16 and 17 is the breakdown of *merge2*.

UN.Table 14 : *merge1* country list

	id	Sort_order	major_area	region	country	Notes	Country_code	Type_of_data(a)	Developed region	Least developed country	Sub-Saharan Africa
0	1	7	Africa	Africa	Africa	NaN	903	NaN	NaN	NaN	NaN
1	2	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	NaN	NaN	NaN
2	3	9	Africa	Eastern Africa	Burundi	NaN	108	B R	No	Yes	Yes
3	4	10	Africa	Eastern Africa	Comoros	NaN	174	B	No	Yes	Yes
4	5	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	No	Yes	Yes

UN.Table 15 : *merge1* migrant/refugee data

	id	year	Sex	International_migrant_stock_at_mid-year	Total_population_at_mid-year(thousands)	International migrant stock as a percentage of the total population(thousands)	Percentage of the international migrant stock	Estimated refugee stock at mid-year	Refugees as a percentage of the international migrant stock_
0	1	1990	Both	15690623	631614.304	2.48421	NaN	5687352	36.246821
1	2	1990	Both	5964031	198231.687	3.008616	NaN	3168001	53.118453
2	3	1990	Both	333110	5613.141	5.934467	NaN	267929	80.43259
3	4	1990	Both	14079	415.144	3.391353	NaN	0	0
4	5	1990	Both	122221	588.356	20.773307	NaN	54508	44.597901

UN.Table 16 : *merge2* country list

	id	Sort_order	major_area	region	country	Notes	Type_of_data(a)	Country_code	Developed region	Least developed country	Sub-Saharan Africa
0	1	7	Africa	Africa	Africa	NaN	903	NaN	NaN	NaN	NaN
1	2	8	Africa	Eastern Africa	Eastern Africa	NaN	910	NaN	NaN	NaN	NaN
2	3	9	Africa	Eastern Africa	Burundi	NaN	108	B R	No	Yes	Yes
3	4	10	Africa	Eastern Africa	Comoros	NaN	174	B	No	Yes	Yes
4	5	11	Africa	Eastern Africa	Djibouti	NaN	262	B R	No	Yes	Yes

UN.Table 17 : *merge2* migrant/refugee data

	id	year	Sex	Annual rate of change of the migrant stock	Annual rate of change of the refugee stock_
0	1	1990-1995	Both	0.826734	0.076037
1	2	1990-1995	Both	-3.435412	-5.30801
2	3	1990-1995	Both	-5.355717	-3.390926
3	4	1990-1995	Both	-0.199873	..
4	5	1990-1995	Both	-4.058465	-9.763426

In addition, the benefit of breaking down the two tables is that it also saves memory and is better for data storage. For example, as shown in UN.Table 18 to 20, the memory of *merge2* decreases from 338.2 KB to 178.1 KB after the breakdown.

UN.Table 18 : *merge2*

```
<class 'pandas.core.frame.DataFrame'>
MultiIndex: 3969 entries, (7, 'Africa', 'Africa', 'Africa', nan, 903, nan, nan, nan, nan) to (265, 'Oceania', 'Polyne
sia', 'Wallis and Futuna Islands', nan, 876, 'B', 'No', 'No', 'No')
Data columns (total 4 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   year                                        3969 non-null   object
 1   Sex                                         3969 non-null   object
 2   Annual rate of change of the migrant stock  3969 non-null   object
 3   Annual rate of change of the refugee stock_ 1319 non-null   object
dtypes: object(4)
memory usage: 338.2+ KB
```

UN.Table 19 : *merge2* country list

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 265 entries, 0 to 264
Data columns (total 11 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   id                      265 non-null    int64
 1   Sort_order              265 non-null    int64
 2   major_area              265 non-null    object
 3   region                  264 non-null    object
 4   country                 265 non-null    object
 5   Notes                   26 non-null     object
 6   Type_of_data(a)         265 non-null    int64
 7   Country_code            232 non-null    object
 8   Developed region        216 non-null    object
 9   Least developed country 216 non-null    object
 10  Sub-Saharan Africa      216 non-null    object
dtypes: int64(3), object(8)
memory usage: 22.9+ KB
```

UN.Table 20 : *merge2* migrant/refugee data

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3969 entries, 0 to 3968
Data columns (total 5 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   id                                          3969 non-null   int64
 1   year                                        3969 non-null   object
 2   Sex                                         3969 non-null   object
 3   Annual rate of change of the migrant stock  3969 non-null   object
 4   Annual rate of change of the refugee stock_ 1319 non-null   object
dtypes: int64(1), object(4)
memory usage: 155.2+ KB
```

Discussion

At first, I felt this project was challenging given I have no previous coding background. However, after completing it, I realized that this project was really helpful in the way that it gave me the chance to familiarize what I had learned in class including various codes as well as the 5 principles and applied the knowledge to a real-world dataset. In addition, I had learned how to use various sources on the Internet to solve the problems in the cleaning process. The challenging part was that there was no specific way of completing the cleaning and no consistent data format. On the one hand, it gave me freedom and flexibility to conduct the project with my own judgement. I could decide which data to include and exclude in the dataset and how to do it. On the other hand, the flexibility also made me slightly disoriented at the beginning and constantly made me question my own decisions along the process.