FINAL PROJECT PAPER

ISQS-6339-001 / Dr. LUCUS David J.

GROUP 7
ALI EZZAOUOUI
BENJAN ADHIKARI
RUSHIT PARMAR
ZOUMANA KEITA

POPULATION GROWTH AND ITS IMPACT

Table of Contents

1.Introduction

The world population in the 1800s was estimated to 1 billion, and currently, this estimation reached 8 billion. By 2025, it would be around 9 to 10 billion. Based on this observation, we can understand how fast the world population is growing.

Given these facts, knowing the impact of the population growth can provide us with more information, thus understand how it could affect our daily lives.

At the speed we're going, we have almost used half of the available resources on our planet. The impact of a growing population will result in a further demand for resources and space. Therefore, we have chosen to investigate and analyze the impact of the population growth on many factors. These are one of the most crucial factors that impact our daily lives like food production, energy consumption, inflation, GDP per capita.

Furthermore, these are important for all Nations across the globe closely for development and substance. We don't claim these factors are impacted only by population growth, but they are obvious.

So, the main question of our dissertation is:

"Does population growth impact energy consumption, food production, inflation and GDP per capita?"

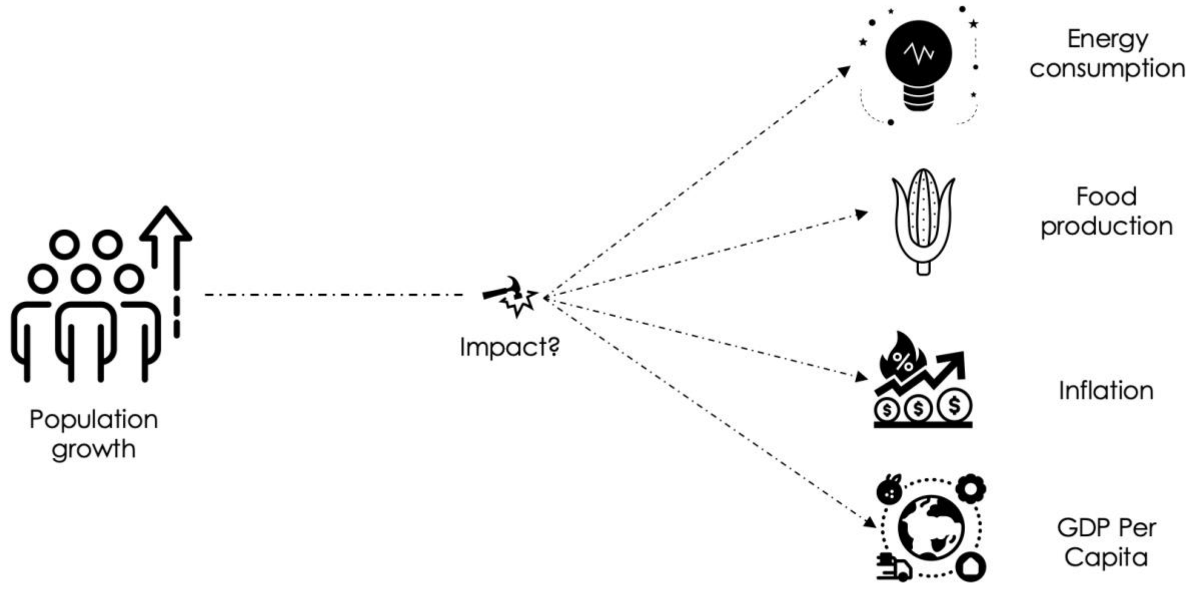


Figure 1: *Problem statement diagram*

In this conceptual deliverable, we will first explore more on the data collected for each domain of interest. Then, we will walk you through all the Extract, Transform and Load (ETL) process, before highlighting all the difficulties we faced and how we tackled them.

We will conclude by trying to answer the previous question form our analysis.

2.Data Domain Discussion

Domains -

a. Global Demography & Geography: Population

Government counts their people every 5 to 10 years, this process is called census. This process of census helps the countries allocate the resources based on the needs of each territory. I t also records information about the citizens like age, sex, and occupation, etc.

The statistical study of population is called demography which studies the size, structure and migration of population around the globe over time. Demographers collect data through the censuses and government registries of birth and death. In addition, demographers gather data through doing surveys in small groups of population to draw conclusions about a population of a specific region.

b. Agriculture: Food Production Around the World

Agriculture plays the main key development in the rise of human civilization. Development of farming led to creation of food surplus which enabled the humans to settle in the cities. The major agricultural products can be divided into foods, fuels, raw materials and fibers. Food can be grouped into grains, vegetables, fruits, oils, meat, eggs and milk. Nowadays, farmers have the tools and resources to produce and to make farming more sustainable.

Last 80 years agricultural products have been highly increased because of the energy-intensive mechanization, pesticides and fertilizers. Most of this energy comes from fossil fuel sources.

c. GDP and Inflation.

Inflation definition goes either in the way of an increase in the money supply or increase in price levels. The general idea about Inflation has to do with a rise in prices.

Gross domestic product (GDP) in a country represents the total aggregate output of its economy. GDP figures that were reported to investors are usually adjusted for inflation. If GDP was reported 6% higher than the previous year and inflation measured 2%, GDP WOULD BE REPORTED AS 4%.

GDP does not reflect the real growth in an economy due to inflation. In other words, GDP growth causes inflation.

d. Energy: Energy consumption

Energy consumption means any form of energy used in manufacturing something or carrying out an action. There are some obvious examples such as total energy consumption used in making cars, energy consumption used in a household like gas, water, electricity and all forms that help us live a comfortable life. In addition, there is energy consumption used in transportation as form of diesel or gasoline.

3. Analysis of data

This section focuses on providing a brief description of each data we have collected.
Further, some of the key issues that we found in almost all datasets were missing values, that will be explained in section 5.

a. Area Dataset

This dataset lists countries across the globe along with their ranking based on their size, that is the surface area. This data is very useful to understand how big a country is and can be used to assess its impact.

Column Name	**Data Type**	**Column Description**
rank	Integer	Rank of country based on the surface

		area
country	String	Country name
area	Float	Surface area of country, units square Kms

b. Country Dataset

This dataset provides a list of countries along with their official name and continent. This is a good reference for data enrichment and standardization.

Column Name	**Data Type**	**Column Description**
country	String	Country name
offName	String	Official country name
continent	String	Continent name

c. Energy Dataset

This dataset is a collection of key metrics for various energy sources such as biofuel, coal, oil, coal, hydro, fossil fuels, electricity, solar and wind. These key metrics provide an overview of various dimensions of a fuel source. There are over 129 attributes therefore we provide this link for its metadata instead of actual list.

d. Food Dataset

This dataset provides the annual yield for different countries. Furthermore, it also provides information regarding the area of land used for agriculture and overall harvest land. This information is very useful when we try to gauge the average land requirement based on current artifacts for agriculture.

This is more like a fact file as the majority of columns are metrics/measurements. However, for our use case column Production (t) is critical as we want to know how much is the agricultural yield in each country.

e. GDP Per Capita Dataset

This dataset provides GDP information from 1960 for different countries across. This dataset has a mix of string type columns and numeric column.

Column Name	Data Type	Column Description
Country Name	String	Country name
Country Code	String	Standard country code
Indicator Name	String	Metric type - GDP per capita (current US$)
Indicator Code	String	Metrics type code - NY.GDP.PCAP.CD

The dataset has a different structure wherein actual years appear as columns and its corresponding value is the actual GDP. So, in order to utilize the data, select years columns have to be converted into one column, which we handle further in the data transformation phase.

f. Inflation Dataset

This dataset covers almost 196 countries from the period 1970 - 2022 measuring key inflammation metric Headline consumer price index (CPI) inflation.

Column Name	Data Type	Column Description
Country Name	String	Country name
IMF Country Code	String	Standard country code
Indicator Type	String	Metric type - Inflation
Series Name	String	Metrics name - Headline Consumer Price Inflation

The dataset has a different structure wherein actual years appear as columns and its corresponding value is the actual CPI. So, in order to utilize the data, select years columns have to be converted into one column, which we handle further in the data transformation phase.

g. Population

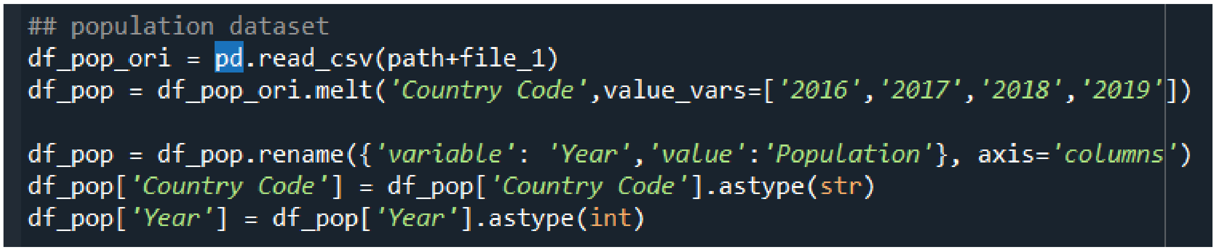
This dataset has the world population by countries from 1960 to 2021.

Column Name	**Data Type**	**Column Description**
Country Name	String	Country name
Country Code	String	Standard country code

The dataset has a different structure wherein actual years appear as columns and its corresponding value is the actual census count. So, in order to utilize the data, select years columns have to be converted into one column, which we handle further in the data transformation phase.

4. Data Reading, Preparation and Cleaning

This section explains in depth the data reading, preparation and cleaning proves.

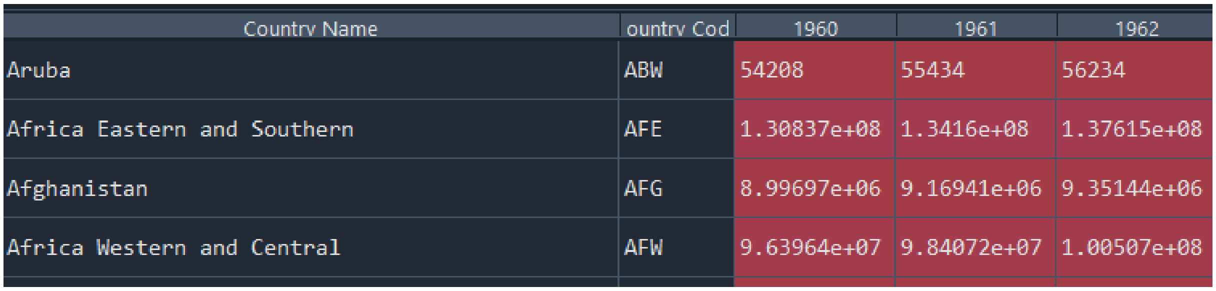


```python
## population dataset
df_pop_ori = pd.read_csv(path+file_1)
df_pop = df_pop_ori.melt('Country Code',value_vars=['2016','2017','2018','2019'])

df_pop = df_pop.rename({'variable': 'Year','value':'Population'}, axis='columns')
df_pop['Country Code'] = df_pop['Country Code'].astype(str)
df_pop['Year'] = df_pop['Year'].astype(int)
```

Figure 4.1: *Reading, and processing of the population data*

We will use *figure 4.1* to explain this section. This is how we prepare the population dataset which is one of the seven datasets used for merging. These six lines of code serve as the standard flow of code. There were a few obstacles and extra steps for some data frames which will be discussed in Section 6 where we talk about obstacles, we ran in the project process.

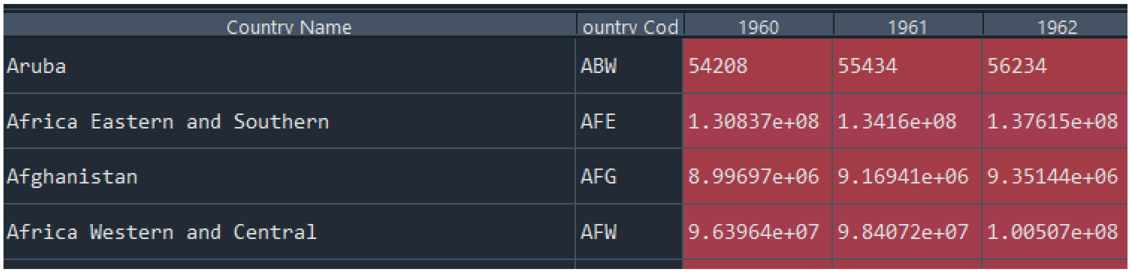
- First, we used the *pandas.read_csv* command with the path of the file as a parameter to read the files into dataframes. This is how the dataframe originally looks:



Country Name	ountry Cod	1960	1961	1962
Aruba	ABW	54208	55434	56234
Africa Eastern and Southern	AFE	1.30837e+08	1.3416e+08	1.37615e+08
Afghanistan	AFG	8.99697e+06	9.16941e+06	9.35144e+06
Africa Western and Central	AFW	9.63964e+07	9.84072e+07	1.00507e+08

Figure 4.2: *First few rows of the population data*

- In *figure 4.2,* you can see that the years are in the column names. However, we want the years to act as keys along with country code or country names, so we use *dataframe.melt* command to record the years into a single column.
- Now, after using the melt function, it gives us 'variable' and 'value' columns which need to be changed to 'Year' and 'Population' - as this dataframe is measuring population. 'Country Code', 'Country names' and 'Year' are used for merging, so we standardize the names of these three columns in each dataframe if necessary. For example, if the year column in a dataframe is registered as 'year' with a small 'y', we rename it to 'Year'. This makes it easier while merging and gives standard column names.
- Finally, we standardize the data types for each column which are used as keys for merging; in this case the columns are 'Country Code' and 'Year'.
- Now, the individual dataframe is ready to merge with others and this is what it looks like:

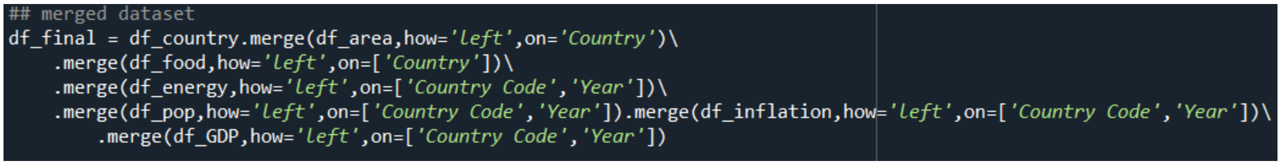


Country Name	ountry Cod	1960	1961	1962
Aruba	ABW	54208	55434	56234
Africa Eastern and Southern	AFE	1.30837e+08	1.3416e+08	1.37615e+08
Afghanistan	AFG	8.99697e+06	9.16941e+06	9.35144e+06
Africa Western and Central	AFW	9.63964e+07	9.84072e+07	1.00507e+08

Figure 4.3

5. Data Merging and Data Enrichment

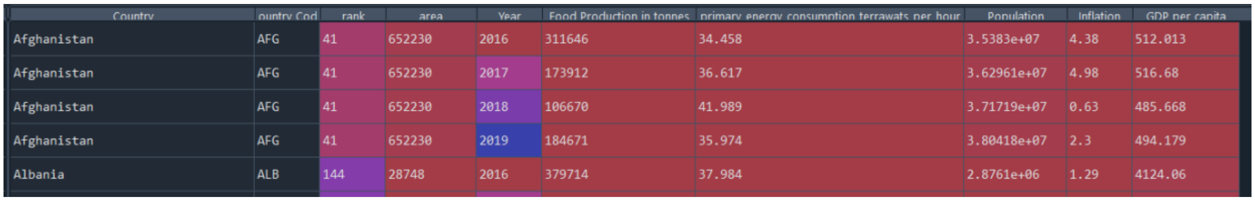
After the seven individual dataframes were prepared, we merged these dataframes.



```
## merged dataset
df_final = df_country.merge(df_area,how='left',on='Country')\
    .merge(df_food,how='left',on=['Country'])\
    .merge(df_energy,how='left',on=['Country Code','Year'])\
    .merge(df_pop,how='left',on=['Country Code','Year']).merge(df_inflation,how='left',on=['Country Code','Year'])\
        .merge(df_GDP,how='left',on=['Country Code','Year'])
```

Figure 5.1

Figure 5.1 shows the code used for merging. We use the ***.merge*** function to merge the seven files. ‘Country’ dataframe which had two attributes ‘Country’ and ‘Country Code’ was used as the primary dataframe and it was left-joined sequentially with others. The keys for merging were: ‘Country’, ‘Country Code’ and ‘Year’ columns. The merged dataframe initially has 773 rows and 10 columns. *Figure 5.2* shows how the final dataframe looks after merging:

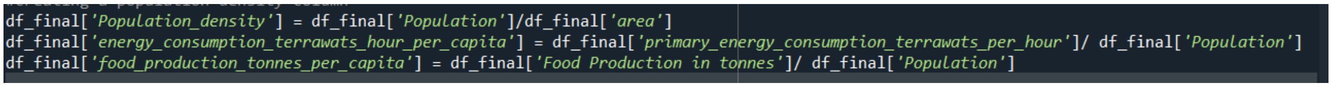


Country	ountry Cod	rank	area	Year	Food Production in tonnes	primary energy consumption terrawats per hour	Population	Inflation	GDP per capita
Afghanistan	AFG	41	652230	2016	311646	34.458	3.5383e+07	4.38	512.013
Afghanistan	AFG	41	652230	2017	173912	36.617	3.62961e+07	4.98	516.68
Afghanistan	AFG	41	652230	2018	106670	41.989	3.71719e+07	0.63	485.668
Afghanistan	AFG	41	652230	2019	184671	35.974	3.80418e+07	2.3	494.179
Albania	ALB	144	28748	2016	379714	37.984	2.8761e+06	1.29	4124.06

Figure 5.2: *Dataframe after the merge*

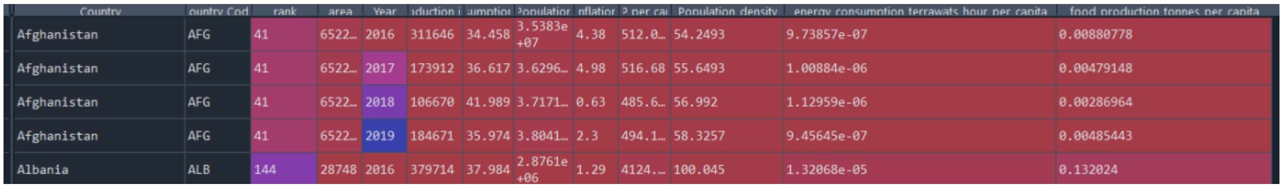
After merging, we need to do some data enrichment which involves adding data and columns which would make the analysis process better and dealing with missing values.

In *Figure 5.3 and 5.4*, we provide two figures with code showing how and which new columns we add and how the data frames look after this new addition of columns.



```
df_final['Population_density'] = df_final['Population']/df_final['area']
df_final['energy_consumption_terrawats_hour_per_capita'] = df_final['primary_energy_consumption_terrawats_per_hour']/ df_final['Population']
df_final['food_production_tonnes_per_capita'] = df_final['Food Production in tonnes']/ df_final['Population']
```

Figure 5.3: *Creation of additional columns*



Country	ountry Cod	rank	area	Year	oduction i	umption	Population	nflation	P per ca	Population density	energy consumption terrawats hour per capita	food production tonnes per capita
Afghanistan	AFG	41	6522…	2016	311646	34.458	3.5383e +07	4.38	512.0…	54.2493	9.73857e-07	0.00880778
Afghanistan	AFG	41	6522…	2017	173912	36.617	3.6296…	4.98	516.68	55.6493	1.00884e-06	0.00479148
Afghanistan	AFG	41	6522…	2018	106670	41.989	3.7171…	0.63	485.6…	56.992	1.12959e-06	0.00286964
Afghanistan	AFG	41	6522…	2019	184671	35.974	3.8041…	2.3	494.1…	58.3257	9.45645e-07	0.00485443
Albania	ALB	144	28748	2016	379714	37.984	2.8761e +06	1.29	4124.…	100.045	1.32068e-05	0.132024

Figure 5.4: *Dataframe with additional columns*

Now, the final dataframe with newly added columns contains 773 rows and 13 columns.

A business can use this dataframe to make prediction models using linear regression lines. For example, it can make predictions for energy consumption using population and GDP as independent variables.

Handling missing values

We used two main approaches to handle missing values as illustrated on the image below.

- **Drop**: Year, Area, and Rank columns represented in total 8% of the overall datasets. We decided to drop underlying observations, because missing values in those columns generated missing values in more than 50% of the rest of the columns.

- **Median replacement**: this approach was the best fit after observing the distribution of the data which was skewed. It has been applied to the remaining 92% of the data. Replacing those missing values using the average would include bias in the data.

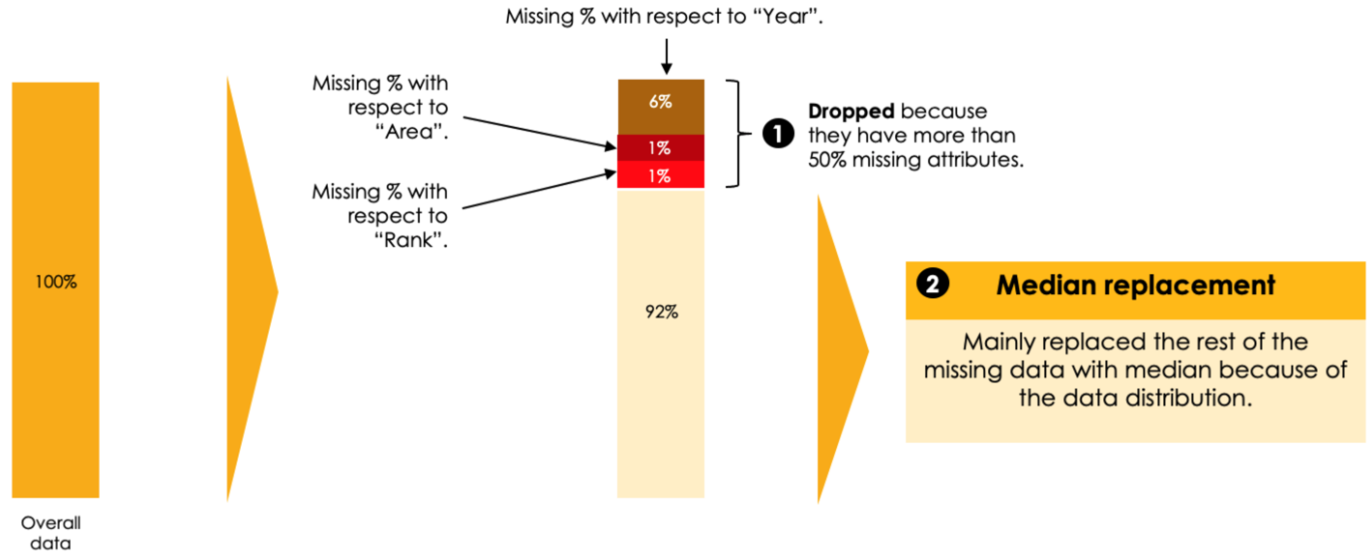


Figure 5.5: *Percentage of missing values in Area, Year and Rank columns*

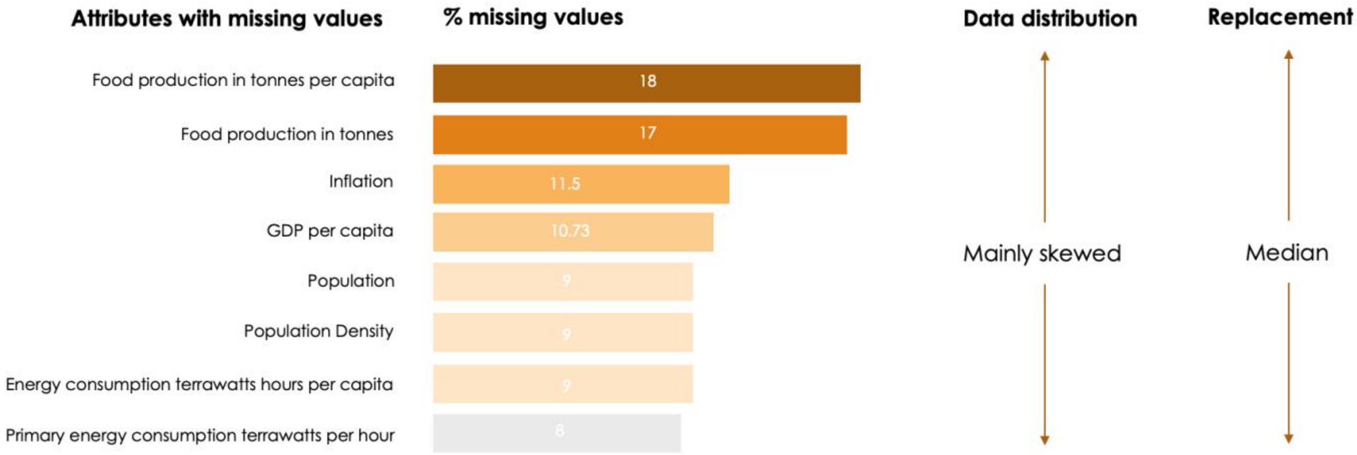


Figure 5.6: *Percentage of missing values in the rest of the columns and their replacement approach*

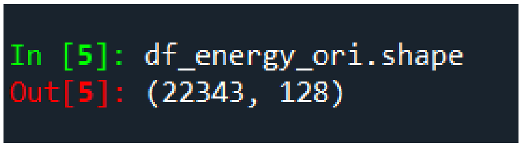
6. Obstacles and Solutions

To paraphrase **Dr. Lucus**, every project has problems and problems are essential for learning. Our project was no stranger to this and in this section, we will talk about the obstacles and our approach to solve them.

This is how we approach this section: we will mention what problems we ran into, what the reason for the problem was and what was our approach to solve it.

a. Irrelevant columns

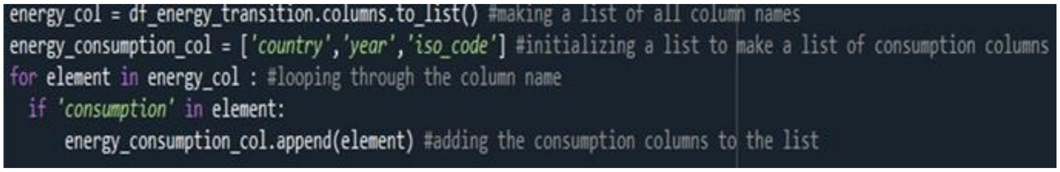
The energy dataframe originally contained 128 columns and only 3% of them were relevant. We had to get rid of all the others and select only relevant ones. We could not eyeball each one of the 128 columns due to time constraints. (*Figure 6.1*)



```
In [5]: df_energy_ori.shape
Out[5]: (22343, 128)
```

Figure 6.1: *Number of rows and columns of energy data*

To solve this, as we were only interested in the consumption part of energy, we looped through all the columns and selected only those that contained the word 'consumption' in them. (*Figure 6.2*)

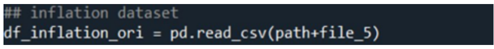


```
energy_col = df_energy_transition.columns.to_list() #making a list of all column names
energy_consumption_col = ['country','year','iso_code'] #initializing a list to make a list of consumption columns
for element in energy_col : #looping through the column name
  if 'consumption' in element:
      energy_consumption_col.append(element) #adding the consumption columns to the list
```

Figure 6.2: *Selection of columns containing "consumption"*

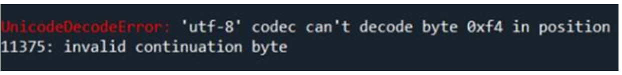
b. File Encoding issue

We used ***pandas.read_csv*** command with file path as parameter to read the file into dataframes. This worked for all but the inflation file which was giving us an encoding error. *(Figure 6.3 and 6.4)*



```
## inflation dataset
df_inflation_ori = pd.read_csv(path+file_5)
```

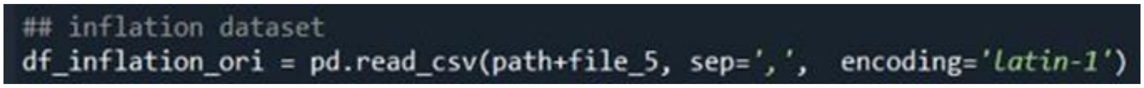
Figure 6.3



```
UnicodeDecodeError: 'utf-8' codec can't decode byte 0xf4 in position
11375: invalid continuation byte
```

Figure 6.4

To solve this, we had to pass the separator and encoding type of the file as additional parameters. *(Figure 6.5)*

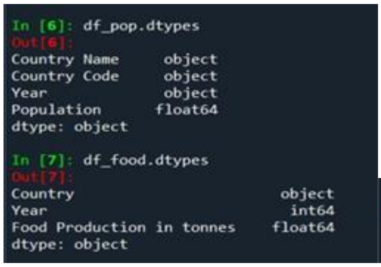


```
## inflation dataset
df_inflation_ori = pd.read_csv(path+file_5, sep=',',  encoding='latin-1')
```

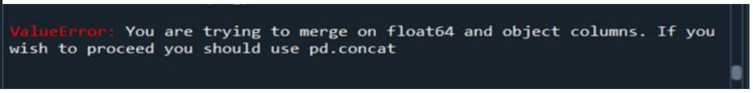
Figure 6.5: *Source code to handle encoding issue*

c. Data Type Mismatch

This problem occurred while we were trying to merge the files. We used year to merge the dataframes. In some dataframes the 'Year' column was registered as object class while in some it was registered as integer class. Because we were trying to merge by keys which were different data types, the program did not understand it and gave us an error. (Figure 6.6 and 6.7)



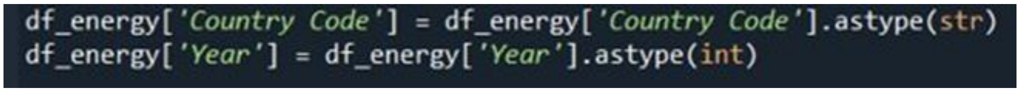
```
In [6]: df_pop.dtypes
Out[6]:
Country Name     object
Country Code     object
Year             object
Population      float64
dtype: object

In [7]: df_food.dtypes
Out[7]:
Country                        object
Year                            int64
Food Production in tonnes     float64
dtype: object
```



```
ValueError: You are trying to merge on float64 and object columns. If you
wish to proceed you should use pd.concat
```

Figure 6.6 ***Figure 6.7***

To solve this, we went back to the individual dataframes and standardized the data types for each of the columns that would later be used as keys for merging. *(Figure 6.8)*



```
df_energy['Country Code'] = df_energy['Country Code'].astype(str)
df_energy['Year'] = df_energy['Year'].astype(int)
```

Figure 6.8: *Source code to handle Year type issue*

7. Data Visualization

Correlation matrix

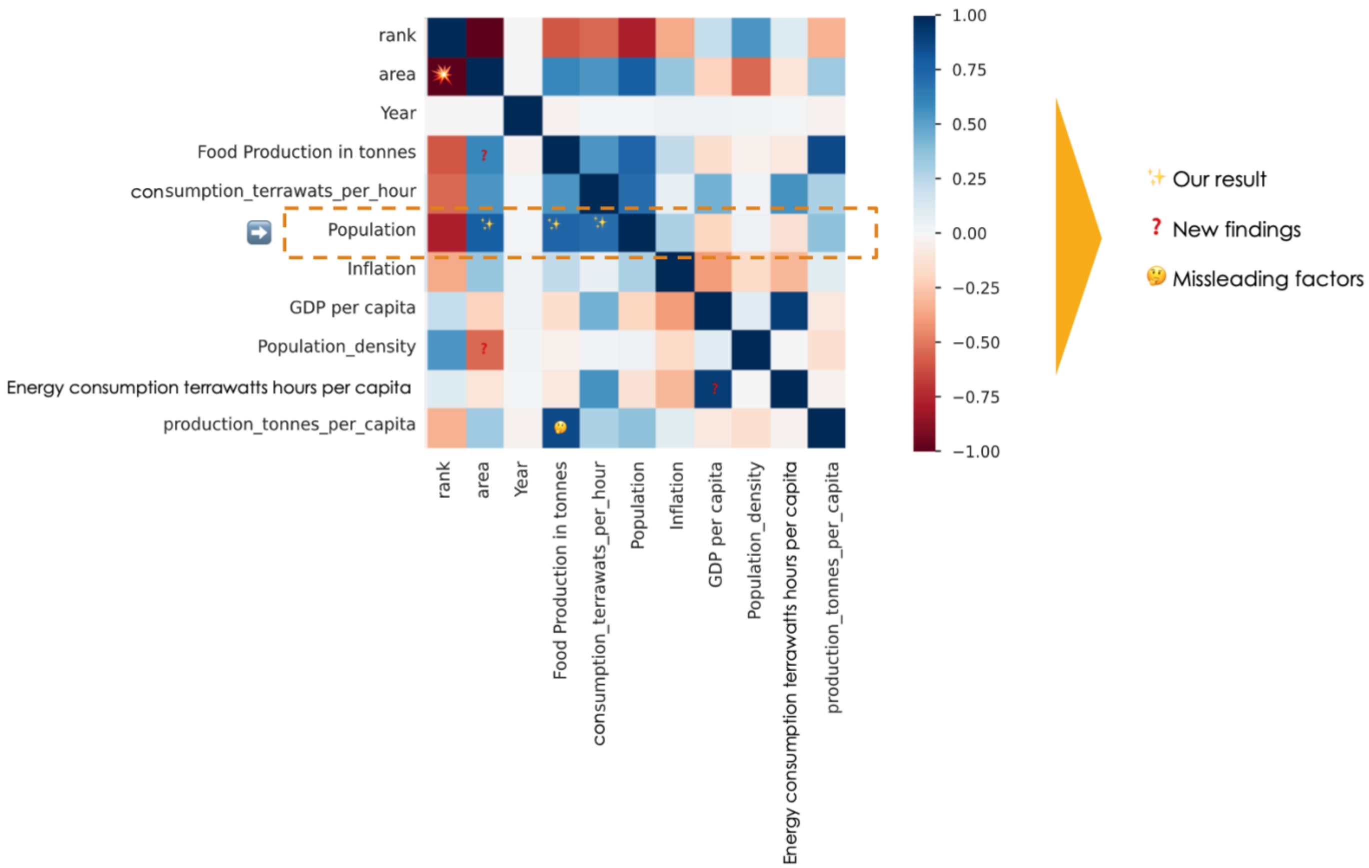


***Figure 7.1:** correlation diagram of the final dataframe*

Figure 7.1 is the correlation diagram of the final dataframe. Below is some information and analysis of it:

- The scale goes from -1 to 1 in numbers and from red to blue in colors. -1 or red means strong negative correlation and 1 or blue means strong positive correlation.

- There are two things that should be discarded from the beginning. The diagonal is blue but it is because it is the correlation of a variable with itself. Also, the ‘area’ and ‘rank’ box is red; this is because rank is based on area. For instance, Russia has the highest area and thus it is ranked one. So, this box tells us nothing new either.

- In our introduction, we focused on the effects of population. So, let’s see the relation of population with other variables. ‘Population’ and ‘area’ are positively correlated because the bigger a country is, the more its population. ‘Population’ and ‘Food production’ are positively correlated because the more the population, more is the manpower and more is production of food. Same with ‘energy consumption’ as with a bigger population, more people need to use energy.

- We can see correlation among other variables not involving population. So, let's look at these new findings. ‘Food production in tonnes’ and ‘area’ are positively correlated as more area means there is more space to sow crops and do agriculture. ‘GDP’ and ‘Energy Consumption’ also have positive correlations. GDP denotes the wealth of citizens and if the citizens are rich then they can afford to do more things, buy more materials and use more resources and thus richer people will use more energy. ‘Area’ and ‘Population density’ are negatively correlated: this was not intuitive to us. After more thinking, we came up with this explanation: if you look at the biggest countries they have a lot of inhabitable lands, like Russia with Siberian parts, Australia with deserts but smaller countries do not have such inhabitable parts and thus negative correlation between ‘Area’ and ‘Population Density’.

- There were some misleading factors in the correlation diagram. Spotting these factors from essential findings are the things that separates good analysis from bad. If you look at ‘Food production in tonnes’ and ‘Food production in tonnes per capita’, they are highly correlated. However, this is not a new finding. As ‘Food production in tonnes per capita’ is a column that we use by using ‘Food production in tonnes’ column, this correlation is expected and very trivial which should be considered. Without much knowledge of the data and concept of correlation, these things could be easily misinterpreted.

8. Flow chart of the project

The following diagram highlights all the steps of the project, from problem understanding to final decision making, with a specification of the technology used for each step.

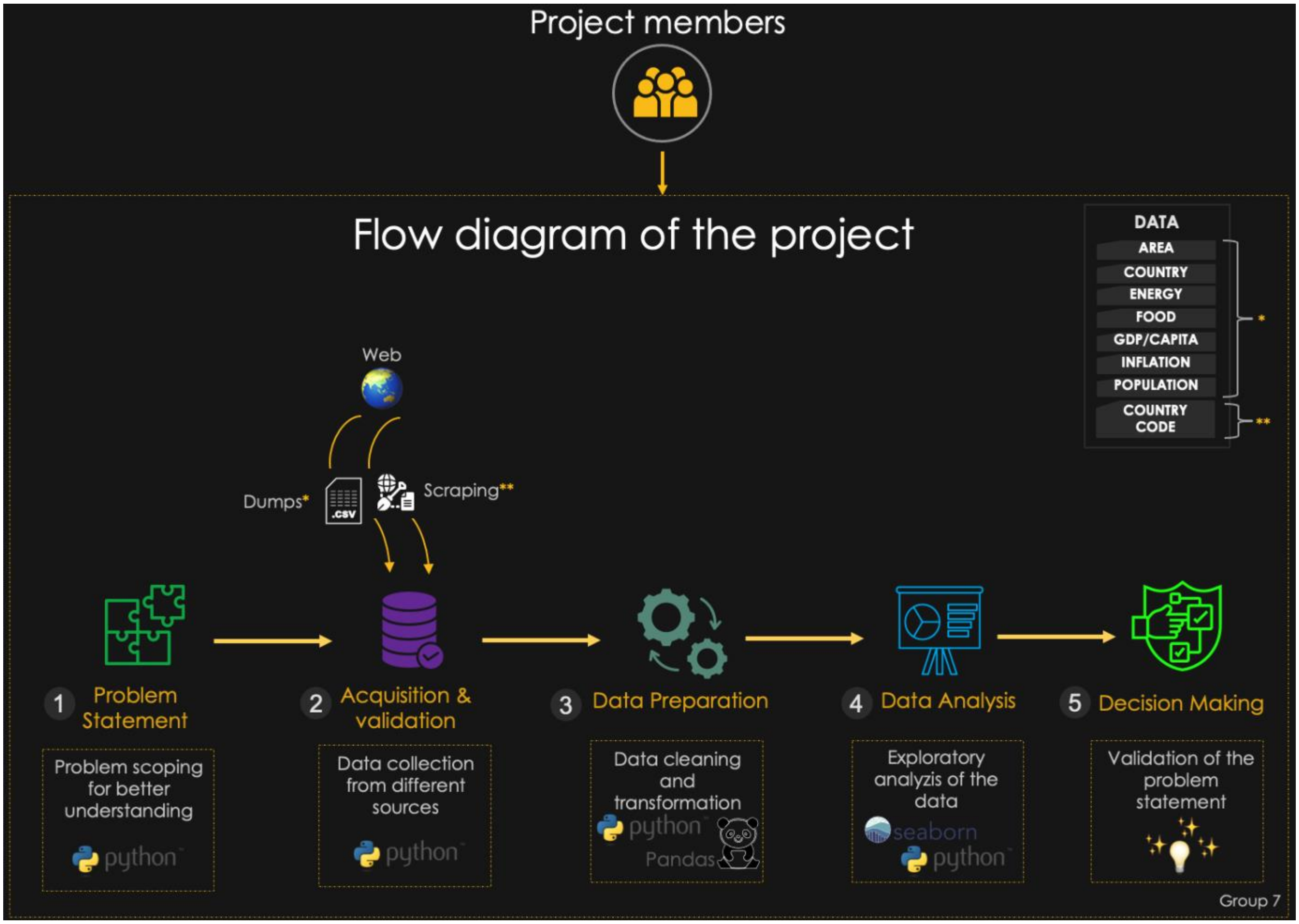

Project members
Flow diagram of the project
DATA
AREA
COUNTRY
ENERGY
FOOD
GDP/CAPITA
INFLATION
POPULATION
COUNTRY CODE
*
**
Web
Dumps*
.csv
Scraping**
1 Problem Statement
Problem scoping for better understanding
python
2 Acquisition & validation
Data collection from different sources
python
3 Data Preparation
Data cleaning and transformation
python
Pandas
4 Data Analysis
Exploratory analyzis of the data
seaborn
python
5 Decision Making
Validation of the problem statement
Group 7

Figure 8.1: *Flow diagram of the project*

Problem statement is crucial to the success of each project because not only it gives a good understanding of the issue being tackled, but also a correct direction to collecting the right datasets.

Real-world data is messy, and this is at least the common denominator of all data practitioners. Data cleaning and transformation play a major role to help in further analysis and decision making.

9. Instructions for running the code

- Import the necessary libraries; in this case we have three.

- Save the ***url*** of the website which will be used for web scraping as a variable ‘URL’.
- Save the path of the directory or folder in your computer as variable ‘path’ which contains the necessary files. You will have to **modify** this step from the original .py file.
- Save the name of each file.
- Each of the files are read into dataframes using the *‘pandas.read_csv’* command with path and file name as parameters. The original dataframes are saved with ‘_ori’ subscript. You can look at them in the Spyder variable explorer.
- Melt the column names into observations using ‘*.melt*’ command and select only the necessary columns.
- Rename the columns.
- Standardize the column data types that will be used as keys for merging.
- To get the country dataframe we have to web scrape. No modification in the ***‘.py’*** file is required for this. You will use the ‘URL’ variable to access and scrape the website.
- Save all the cleaned dataframes to files in your folder by using ‘*.to_csv*’ command. All the source and cleaned csv files are provided in the zip folder. If you do not want to output the cleaned files again, you should **skip** this step.
- Now, merge the dataframes using the **'**.*merge*' command with ‘Country’, ‘Country Code’, and ‘Year’ as keys.
- Create new columns from the existing columns.
- Drop some rows with missing values and populate the rest with median.
- Create a correlation diagram.
- Output the final dataframe to a csv file in your directory. Again, as the final csv file is provided you may want to **skip** this step.

10. Conclusion

After going through the ETL process and data enrichment, we were able to understand and handle all challenges. We get successful application of the ETL approach to real-life problems. We spent most of the time on data Transformation that shows the benefit of doing this project. Therefore, we enhance our capabilities to transfer the acquired skill sets to similar problems in industry. The data analysis shows there are other factors which impact those variables, so performing additional preprocessing and data exploration should be done in next steps.

11. References

https://worldpopulationreview.com/country-rankings/largest-countries-in-the-world (Area)

https://www.visualcapitalist.com/cp/mapped-food-production-around-the-world/ (Food)

https://data.worldbank.org/indicator/NY.GDP.PCAP.CD (GDP per capita)

https://www.worldbank.org/en/research/brief/inflation-database (Inflation)

https://github.com/owid/energy-data (owid-energy-data 3)

https://www.kaggle.com/datasets/kaggleashwin/population-dataset (Population)

https://countrycode.org (country names)

12. Citations

Ganti, A. (2022, September 9). *What real gross domestic product (real GDP) is, how to calculate it, VS nominal*. Investopedia. Retrieved October 12, 2022, from https://www.investopedia.com/terms/r/realgdp.asp

Census. National Geographic Society. (n.d.). Retrieved October 12, 2022, from https://education.nationalgeographic.org/resource/census

Barnes, R. (2022, July 13). *The importance of inflation and GDP*. Investopedia. Retrieved October 12, 2022, from https://www.investopedia.com/articles/06/gdpinflation.asp

Teba, C. (2022, August 31). *What does energy consumption mean?* Dexma. Retrieved October 12, 2022, from https://www.dexma.com/blog-en/energy-consumption-definition/