Socio-Economic and Technological Trends Analysis

ENSF692 Group Project Report

Bo Zheng Ma, Rick Zhang, Warisa Khaophong

Project outline

Our goal with this project is to create a program that imports datasets containing information of global socio-economic and technological data. We want to present the user with a way of seeing such trends within select countries and allow them to perform insightful analysis.

Datasets

This project used 10 datasets, combining and mixing them into 5 different topics to explore. These datasets are either related to technology, economic development, public health, energy consumption, and digital connectivity.

- **Cell Phones (Total)**: Number of cell phones that exist in each country each year
- **Coal Consumption (Total)**: How much coal (oil tonnes equivalent) is consumed in each country annually.
- **Electricity Generation (Total)**: How much electricity (KWH) is generated in each country, each year.
- **Residential Electricity Use (Total)**: How much electricity (KWH) is used in residences in each country, each year
- **Population**: Amount of people per country, per year
- **Inflation (Annual %)**: Inflation as measured by the annual growth rate of the GDP implicit deflator shows the rate of price change in the economy of each country.
- **Total GDP (US$, inflation-adjusted)**: GDP at purchaser's prices
- **Average Daily Income, $/Person/Day, Inflation- & Price Adjusted**: Mean daily household per capita income
- **Life Expectancy (At Birth)**: Male and female life expectancy at birth.
- **Number Of People Using Internet**: Total number of people who used the internet in the last 3 months of each year of each country.

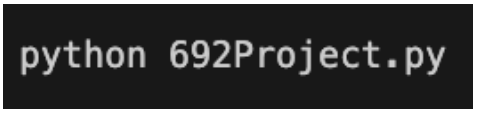
Topics of Analysis

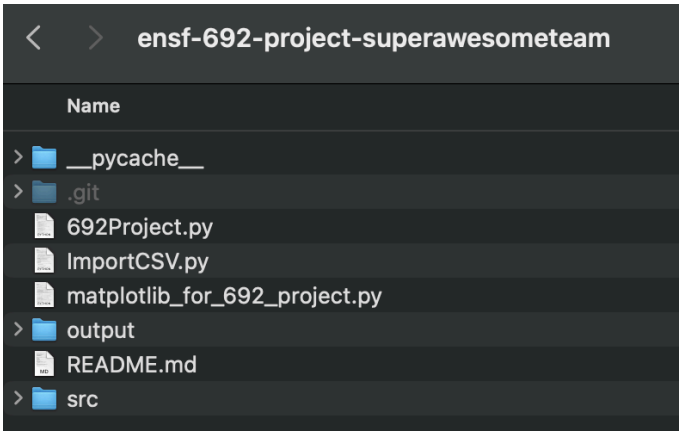
1. **Life quality**:
    a. Datasets used: Life Expectancy, GDP Per Capita (Added column), Number of cell phones, Internet Users (Average per year).
    b. Provides insights into how technological advancements and economic conditions contribute to overall life quality.
2. **Economy**:
    a. Datasets used: Total GDP (in USD, inflation adjusted), Population, Inflation (annual percent)
    b. Provide insights into how population changes influence GDP and inflation rates.
3. **Technology and economy**:

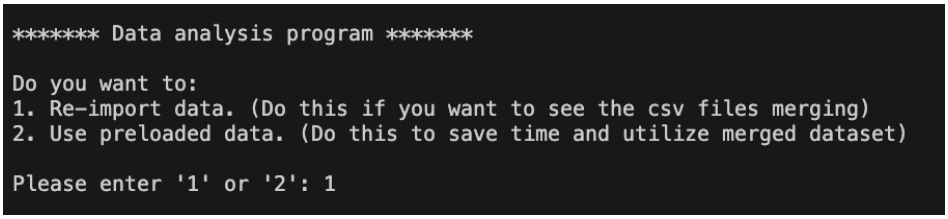
    a. Datasets used: Number of cell phones, Internet Penetration Rate, Total GDP (in USD, inflation adjusted), Daily Income
    b. Provides insight into how the adoption of technology (cellphones and internet) correlates with economic prosperity (GDP and daily income).
4. **Energy**:
    a. Datasets used: Electricity generated (kilowatt hours), Total Coal Consumption (tonnes oil equivalent), Net Internet Users (Average per year)
    b. Could reveal whether increases in internet usage drive higher electricity demand, potentially straining energy resources or leading to greater coal consumption.
5. **Digital infrastructure**:
    a. Dataset used: Number of Cellphones, Net Internet Users, Electricity generated (kilowatt hours)
    b. Provides insights into how well a country is equipped to support and sustain digital activities, which are crucial for modern economies and societies.

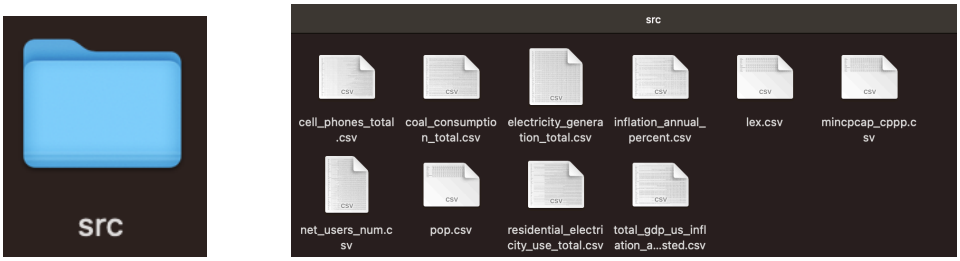
Program execution

User Instruction

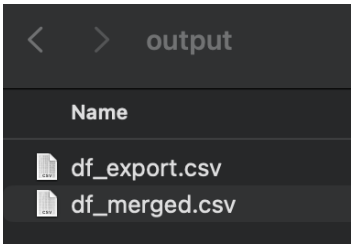
1. To run the program, enter `python 692Project.py` in the terminal after selecting the correct directory. `**692Project.py**` contains the main code. `**matplotlib_for_692_project.py**` contains functions related to plotting. `**ImportCSV.py**` contains functions related to csv file importing and merging.
   a. 

   b. 

2. Upon starting the program, select option `**1**` to import the 10 csv files from the `src` folder. The program merges the csv files to create a multi-indexed data frame with 11 columns and approximately 1000 rows; it is exported as `**df_merged.csv**`. Then the program will add 4 more columns by running calculations on the original data frame. The final data frame contains 15 columns and approximately 1000 rows; it is exported as `**df_export.csv**`. Option `**2**` is for debugging purposes. It's created due to the fact that importing and formatting all the csv files takes about 20 seconds and the developers want a way to bypass that.
   a. 

   b. 

c.



d.

```
******* Data analysis program *******

Do you want to:
1. Re-import data. (Do this if you want to see the csv files merging)
2. Use preloaded data. (Do this to save time and utilize merged dataset)

Please enter '1' or '2': 1

Loading data begin....
1. Loading inflation data...
9941 values imported successfully...
2. Loading mincpcap data...
58500 values imported successfully...
3. Loading internet user data...
5541 values imported successfully...
4. Loading coal consumption data...
4023 values imported successfully...
5. Loading total GDP data...
10005 values imported successfully...
6. Loading life expectancy data...
57310 values imported successfully...
7. Loading population data...
59000 values imported successfully...
8. Loading electricity generation data...
2674 values imported successfully...
9. Loading electricity consumption data...
4672 values imported successfully...
10. Loading cell phone user data...
10051 values imported successfully...
Finished data import.
Merging data...
10
.
.
.
.
.
.
.
.
.
Final data size: (1283, 10)
Finished data merge.
Data preview:
              inflation_percent  coal  internet  daily_income  GDP_USD_Total  life_exp_year  population  electricity_generation  residential_electricity_use  cell_phone_total
country year
UAE     1991              0.811     0         0          86.8   128000000000           68.7     2009999             17200000000                   7290000000             43000
        1992              1.810     0         0          88.0   133000000000           68.8     2120000             18700000000                   7850000000             48900
        1993              1.280     0         0          87.8   134000000000           68.8     2220000             21700000000                   9130000000             70600

Adding columns to the dataframe...
Successfully added columns: GDP_per_capita, Internet_Penetration_Rate
```

3. The user then gets to see an aggregate statistic for the entire dataset and is offered 5 topics to choose from. Enter the corresponding number to continue, or enter `0` to exit the program. For example, enter `2`.

a.

```
Topics for analysis:
1. Life Quality
2. Economy
3. Energy
4. Technology
5. Digital Infrastructure
0. Exit

Please select the information you want to retrieve by entering their corresponding number: 2
```

4. Once the topic is selected, the user is presented with a list of countries they can perform the topic analysis on. For example, enter `Turkey`.

a.

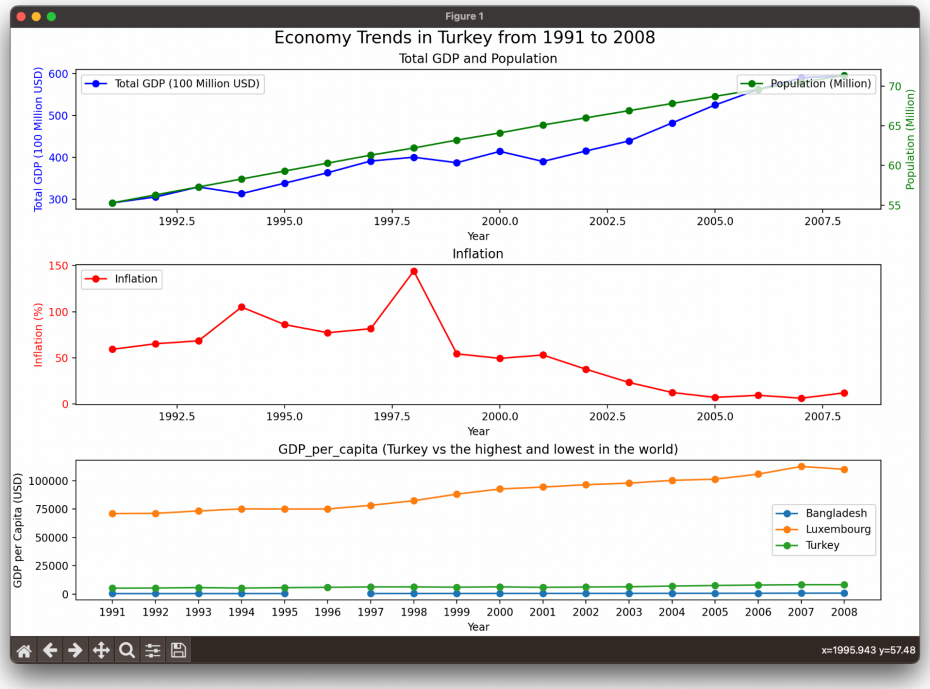
```
Please select the information you want to retrieve by entering their corresponding number: 2


Available countries:
Algeria  Argentina  Australia  Austria  Azerbaijan  Bangladesh  Belarus  Belgium  Brazil  Bulgaria  Canada  Chil
e  China  Colombia  Croatia  Cyprus  Czech Republic  Denmark  Ecuador  Egypt  Estonia  Finland  France  Germany
 Greece  Hong Kong, China  Hungary  Iceland  India  Indonesia  Iran  Ireland  Israel  Italy  Japan  Kazakhstan
Kuwait  Latvia  Lithuania  Luxembourg  Malaysia  Mexico  Morocco  Netherlands  New Zealand  Norway  Oman  Pakist
an  Peru  Philippines  Poland  Portugal  Qatar  Romania  Russia  Saudi Arabia  Singapore  Slovak Republic  Slove
nia  South Africa  South Korea  Spain  Sri Lanka  Sweden  Switzerland  Thailand  Trinidad and Tobago  Turkey  Tu
rkmenistan  UAE  UK  USA  Ukraine  Uzbekistan  Vietnam


Please enter the country you want to analyze (Enter in displayed format): Turkey
```

5. Once all the inputs are taken, the program will provide a specific analysis based on the topic and print the corresponding pivot table.

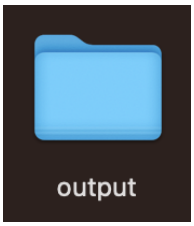
```
Please enter the country you want to analyze (Enter in displayed format): Turkey

Analyzing Economy data for TURKEY:
Category specific aggregation and analysis:
       GDP_USD_Total    population  inflation_percent
count   1.800000e+01  1.800000e+01          18.000000
mean    4.182778e+11  6.352778e+07          52.842778
std     9.632226e+10  5.043086e+06          38.009146
min     2.910000e+11  5.530000e+07           6.200000
25%     3.442500e+11  5.955000e+07          15.125000
50%     3.955000e+11  6.365000e+07          53.650000
75%     4.712500e+11  6.757500e+07          75.000000
max     5.950000e+11  7.130000e+07         144.000000

Pivot table for visualizing trends:
       GDP_USD_Total                                          ...  population
year             1991          1992          1993          1994  ...        2005        2006        2007
2008
country                                                       ...
Turkey   2.910000e+11  3.050000e+11  3.290000e+11  3.130000e+11  ...  68700000.0  69600000.0  70500000.0  713000
00.0

[1 rows x 54 columns]

Pivot table saved as 'output/economy_turkey_pivot.csv'.

Final dataframe saved as: output/df_export

Do you want to view and store analysis plots for current country? (1 for yes, 2 for no):
```

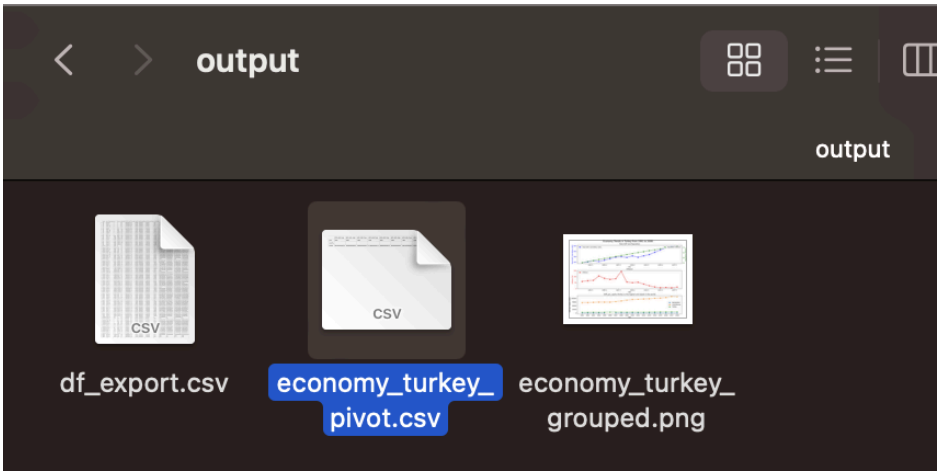
a.

6. The user will then be asked whether to print and save analysis plots. Enter `1` to show and save the plot; enter `2` to skip it. If the user prints the plot, they must close the plot before they can continue with the program.



a.

7. The /output/ directory will contain the final merged dataset named df_export.csv, pivot tables, and plots created by the program.
    a. 



8. The program then jumps back to step 3. The program continues running until `0` is entered to exit the program.
    a. 

*** Note: All inputs are sanitized with try/except statements.

Requirements

Requirement 1: Useful Information and User Interface
Requirement: Your application must return useful information. Design an interface that allows users to search based on some sort of criteria or keywords.
Implementation:
The function ***get_user_selection(df)*** presents a user interface where users can select a category of analysis and a country. The options are displayed clearly, and users can input their choices.

Requirement 2: Two Pieces of Information

Requirement: The user must provide at least two pieces of information/selection (e.g., "school name" and "grade").

Implementation:

The function ***get_user_selection(df)*** requires users to provide two pieces of information: the category of analysis and the country. This ensures the system can generate specific and meaningful reports based on user input.

```
Topics for analysis:
1. Life Quality
2. Economy
3. Energy
4. Technology
5. Digital Infrastructure
0. Exit

Please select the information you want to retrieve by entering their corresponding number: 1     ← User's input 1


Available countries:
Algeria  Argentina  Australia  Austria  Azerbaijan  Bangladesh  Belarus  Belgium  Brazil  Bulgaria  Canada  Chile  China  Colombia
 Croatia  Cyprus  Czech republic  Denmark  Ecuador  Egypt  Estonia  Finland  France  Germany  Greece  Hong kong, china  Hungary  Ic
eland  India  Indonesia  Iran  Ireland  Israel  Italy  Japan  Kazakhstan  Kuwait  Latvia  Lithuania  Luxembourg  Malaysia  Mexico
Morocco  Netherlands  New zealand  Norway  Oman  Pakistan  Peru  Philippines  Poland  Portugal  Qatar  Romania  Russia  Saudi arabi
a  Singapore  Slovak republic  Slovenia  South africa  South korea  Spain  Sri lanka  Sweden  Switzerland  Thailand  Trinidad and t
obago  Turkey  Turkmenistan  Uae  Uk  Ukraine  Usa  Uzbekistan  Vietnam


Please enter the country you want to analyze: china     ← User's input 2
```

Requirement 3: Clear Input Instructions and Error Handling

Requirement: Give the user clear input instructions. If an invalid entry is given, use try/except statements to handle the error and continue to prompt for user input.

Implementation:

The function ***get_user_input(prompt, options)*** prompts the user for input and validates it against a list of valid options. If the user provides invalid input, an error message is displayed, and the user is prompted again.

```python
def main():
    try:
        df = load_data_choice()
        df = add_columns(df)
        print("\nAggregate statistics for the entire dataset:")
        print(df.describe())
        while True:
            category, country = get_user_selection(df)
            if category is None or country is None:
                break
            analyze_data(df, category, country)
    except Exception as e:
        print(f"An error occurred: {e}. \n\nExiting the program.")
        return
```

Requirement 4: Not hard-code any data values
Requirement: You must not hard-code any data values (the data within your spreadsheets could be changed!).
Implementation:
All data values are dynamically loaded from CSV files using the ***loaddata()*** function. This function reads and merges the data from the CSV files at runtime, ensuring that the application always works with the latest data without hard-coding values.

```
def load_data_choice():
    """
    Function to prompt the user to choose between re-importing the data or using the preloaded data.
    """
    print("\n\n******* Data analysis program *******")
    while True:
        try:
            choice = input("\nDo you want to: \n1. Re-import data. (Do this if you want to see the csv files merging)
            if choice == '1':
                return loaddata()
            elif choice == '2':
                try:
                    return pd.read_csv('df_final.csv')
                except FileNotFoundError:
                    print("File df_final.csv not found. Loading new data instead.")
                    return loaddata()
            else:
                raise ValueError("Invalid choice. Please enter '1' or '2'.")
        except ValueError as e:
            print(f"An error occurred: {e} Please try again.")
```

Requirement 5: Clear Output Information
Requirement: Any output information must be clearly defined using printed headers (DataFrame tables) or sentences (scalar values).
Implementation:
The analysis results are printed with headers and descriptive statistics. The function ***analyze_data(df, category, country)*** provides structured output for each category and country combination.

```
Analyzing Life Quality data for CHINA:
Category specific aggregation and analysis:
       life_exp_year  GDP_per_capita  cell_phone_total  Internet_Penetration_Rate
count      15.000000       15.000000      1.500000e+01                  15.000000
mean       71.793333     2624.220609      2.117167e+08                   5.510507
std         1.549408     1035.752824      2.176573e+08                   6.860270
min        69.400000     1363.636364      1.570000e+06                   0.001190
25%        70.700000     1835.096774      1.855000e+07                   0.102092
50%        71.700000     2362.204724      1.450000e+08                   2.700787
75%        72.750000     3226.923077      3.640000e+08                   8.061538
max        74.300000     4691.729323      6.410000e+08                  23.007519
Pivot table for visualizing trends:
        GDP_per_capita                                                  ... life_exp_year
year              1994    1995         1996         1997    1998         1999  ...          2003  2004  2005  2006  2007  2008
country                                                                 ...
China      1363.636364  1500.0  1634.146341  1774.193548  1896.0  2023.809524  ...          72.3  72.5  73.0  73.7  74.2  74.3

[1 rows x 60 columns]
Pivot table saved as 'output/life_quality_china_pivot.csv'.

Final dataframe saved as: output/df_export
```

Methodology

Use of describe method for the entire dataset:

```
print("\nAggregate statistics for the entire dataset:")
print(df.describe())
```

```
Aggregate statistics for the entire dataset:
              year  inflation_percent          coal      internet  ...  residential_electricity_use  cell_phone_total  GDP_per_capita  Internet_Penetration_Rate
count  1283.000000        1283.000000  1.283000e+03  1.283000e+03  ...                 1.283000e+03      1.283000e+03     1283.000000                1283.000000
mean   1999.812159          36.715938  3.506647e+07  6.761938e+06  ...                 4.746725e+10      1.417723e+07    19592.706944                  17.909367
std       5.112291         207.312315  1.289733e+08  2.223011e+07  ...                 1.434430e+11      4.214308e+07    19818.996959                  23.688670
min    1991.000000         -26.300000  0.000000e+00  0.000000e+00  ...                 3.600000e+08      0.000000e+00      501.834862                   0.000000
25%    1996.000000           2.080000  6.470000e+05  7.010000e+04  ...                 6.145000e+09      2.170000e+05     3862.886773                   0.402387
50%    2000.000000           4.760000  3.700000e+06  7.080000e+05  ...                 1.420000e+10      1.910000e+06    10673.854447                   5.649123
75%    2004.000000          11.300000  1.490000e+07  4.079999e+06  ...                 3.430000e+10      9.270000e+06    32677.639724                  28.690162
max    2008.000000        3330.000000  1.610000e+09  3.060000e+08  ...                 1.390000e+12      6.410000e+08   112500.000000                  90.566038

[8 rows x 13 columns]
```

Here we used the .describe() method to show information of the entire dataset after adding the 2 new columns.

Masking operation:

```
# masking to focus on relevant data points for each analysis
df_country = df_country[df_country['GDP_USD_Total'] > (df_country['GDP_USD_Total'].max()*0.25)]  # masking operation
```

We decided to use a masking operation to ensure that the GDP data only contained data greater than 25% of the maximum value in that column. This ensured that values too small weren't included in future calculations.

Added 4 columns:

```python
def add_columns(df):
    '''
    Add columns to the dataframe for GDP per capita and Internet Penetration Rate
    :@param df: dataframe to add columns to
    :@return: dataframe with added columns
    '''

    print("\nAdding columns to the dataframe...")
    try:
        # GDP per capita
        df['GDP_per_capita'] = df['GDP_USD_Total'] / df['population']
        # percentage of the total population that has access to the Internet
        df['Internet_Penetration_Rate'] = (df['internet'].replace(',', '').astype(float) / df['population'].replace(',', '').astype(float)) * 100
        df['Energy_per_capita'] = (df['residential_electricity_use'].replace(',', '').astype(float) / df['population'].replace(',', '').astype(float))
        df['Cell_phone_per_capita'] = (df['cell_phone_total'].replace(',', '').astype(float) / df['population'].replace(',', '').astype(float))
        print("Successfully added columns: GDP_per_capita, Internet_Penetration_Rate, Energy_per_capita, Cell_phone_per_capita\n")
    except Exception as e:
        print(f"Failed to add columns, an error occurred: {e}.")


    return df
```

4 columns are added to the original data frame:

1. Added `GDP_per_capita` by dividing `GDP_USD_Total` by `population`.
2. Added `Internet_Penetration_Rate` by dividing `internet` by `population` and multiplying by 100.
3. Added `Energy_per_capita` by dividing `residential_electricity_use` by `population`.
4. Added `Cell_phone_per_capita` by dividing `cell_phone_total` by `population`.

Aggregation computation for subset of data:

```python
if category == 'Life Quality':
    # Life Expectancy, Total GDP, Number of Cellphones, Internet Users
    grouped_data = df_country.groupby('year').agg({
        'life_exp_year': 'mean',
        'GDP_per_capita': 'mean',
        'cell_phone_total': 'sum',
        'Internet_Penetration_Rate': 'mean'
    })
```

As an example, we performed aggregation by getting the mean of life expectancy per year and getting the sum of all cell phones each year.

Groupby operation:

```python
elif category == 'Energy':
    # Electricity Generation, Coal Consumption, Internet Users
    grouped_data = df_country.groupby('year').agg({
        'electricity_generation': 'sum',
        'coal': 'mean',
        'internet': 'mean'
    })
    pivot = pd.pivot_table(df_country, values=['electricity_gene
```

In the code, we have numerous areas where we used the groupby() function. As an example above, we grouped by year for these 3 datasets to later create a pivot table.

Create/print pivot table:

```
elif category == 'Economy':
    # Total GDP, Population, Inflation
    grouped_data = df_country.groupby('year').agg({
        'GDP_USD_Total': 'sum',
        'population': 'max',
        'inflation_percent': 'mean'
    })
    pivot = pd.pivot_table(df_country, values=['GDP_USD_Total', 'population', 'inflation_percent'], index='country', columns='year', aggfunc='mean')
```

Statistics

Aggregate statistic of the entire dataset:

```
Aggregate statistics for the entire dataset:
              year  inflation_percent          coal      internet  ...  residential_electricity_use  cell_phone_total  GDP_per_capita  Internet_Penetration_Rate
count  1283.000000        1283.000000  1.283000e+03  1.283000e+03  ...                 1.283000e+03      1.283000e+03     1283.000000                1283.000000
mean   1999.812159          36.715938  3.506647e+07  6.761938e+06  ...                 4.746725e+10      1.417723e+07    19592.706944                  17.909367
std       5.112291         207.312315  1.289733e+08  2.223011e+07  ...                 1.434430e+11      4.214308e+07    19818.996959                  23.688670
min    1991.000000         -26.300000  0.000000e+00  0.000000e+00  ...                 3.600000e+08      0.000000e+00      501.834862                   0.000000
25%    1996.000000           2.080000  6.470000e+05  7.010000e+04  ...                 6.145000e+09      2.170000e+05     3862.886773                   0.402387
50%    2000.000000           4.760000  3.700000e+06  7.080000e+05  ...                 1.420000e+10      1.910000e+06    10673.854447                   5.649123
75%    2004.000000          11.300000  1.490000e+07  4.079999e+06  ...                 3.430000e+10      9.270000e+06    32677.639724                  28.690162
max    2008.000000        3330.000000  1.610000e+09  3.060000e+08  ...                 1.390000e+12      6.410000e+08   112500.000000                  90.566038

[8 rows x 13 columns]
```

Sample output containing statistic analysis

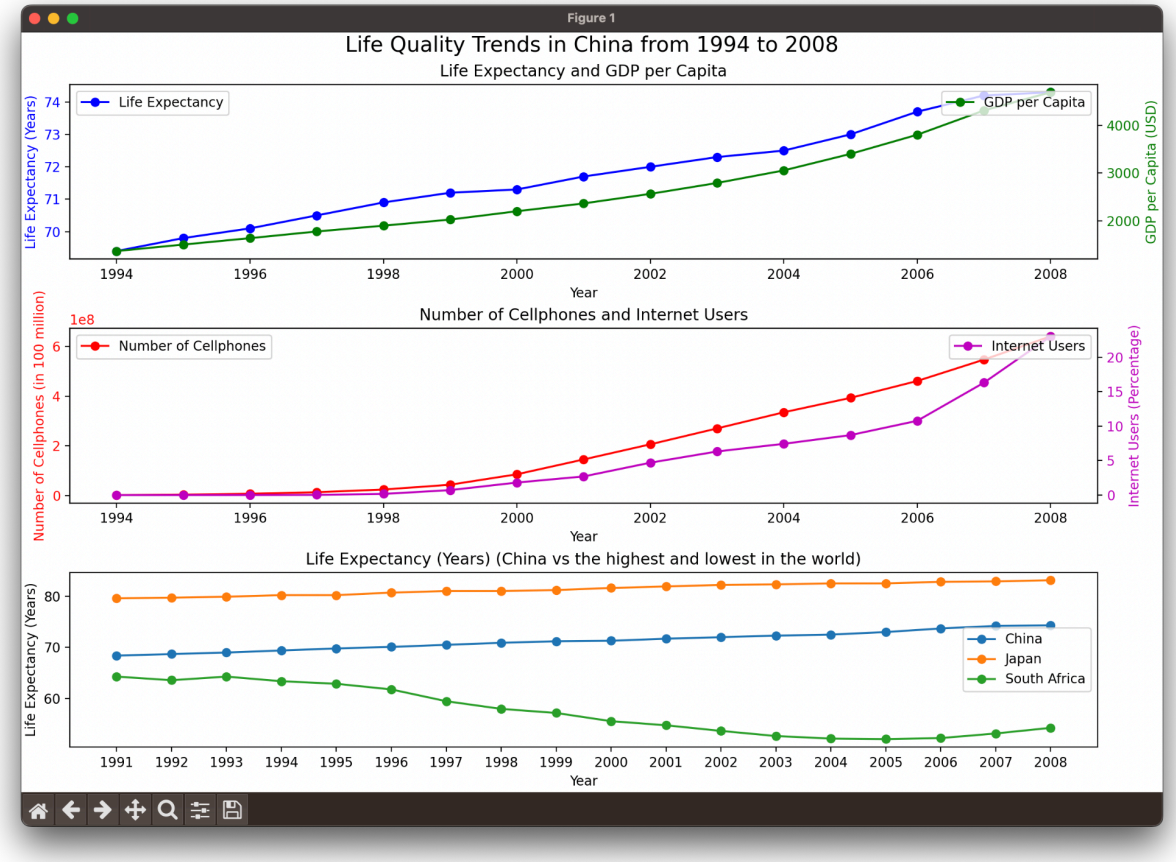
```
Please enter the country you want to analyze (Enter in displayed format): USA

Analyzing Economy data for USA:
Category specific aggregation and analysis:
       GDP_USD_Total    population  inflation_percent
count   1.800000e+01  1.800000e+01          18.000000
mean    1.319444e+13  2.797778e+08           2.218889
std     2.227495e+12  1.693432e+07           0.607579
min     9.800000e+12  2.520000e+08           1.130000
25%     1.122500e+13  2.667500e+08           1.852500
50%     1.350000e+13  2.805000e+08           2.195000
75%     1.495000e+13  2.932500e+08           2.602500
max     1.640000e+13  3.060000e+08           3.380000

Final dataframe saved as: 'output/df_export'
```

Pivot tables

	A	B	C	D
1	year	Japan	South Africa	Turkey
2	1991	79.6	64.3	69.3
3	1992	79.7	63.6	69.6
4	1993	79.9	64.3	70.1
5	1994	80.2	63.4	70.4
6	1995	80.2	62.9	70.8
7	1996	80.7	61.8	71.5
8	1997	81	59.5	72
9	1998	81	58	72.7
10	1999	81.2	57.2	72.8
11	2000	81.6	55.6	74.7
12	2001	81.9	54.8	75
13	2002	82.2	53.7	75.2
14	2003	82.3	52.7	75.7
15	2004	82.5	52.2	76.3
16	2005	82.5	52.1	76.8
17	2006	82.8	52.3	77
18	2007	82.9	53.2	77.1
19	2008	83.1	54.3	77
20				

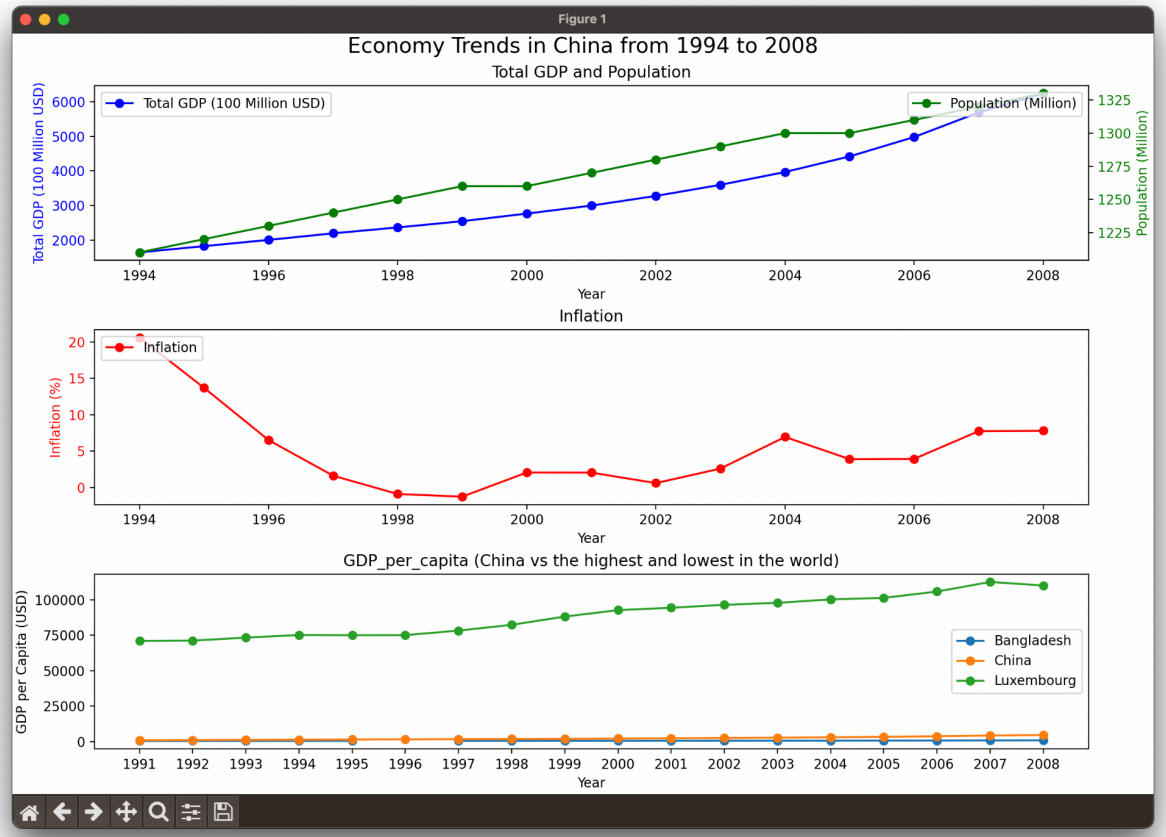
This is the pivot table that is created after selecting a topic and country. It provides values of other two countries to allow the user to make better comparisons.

Plots



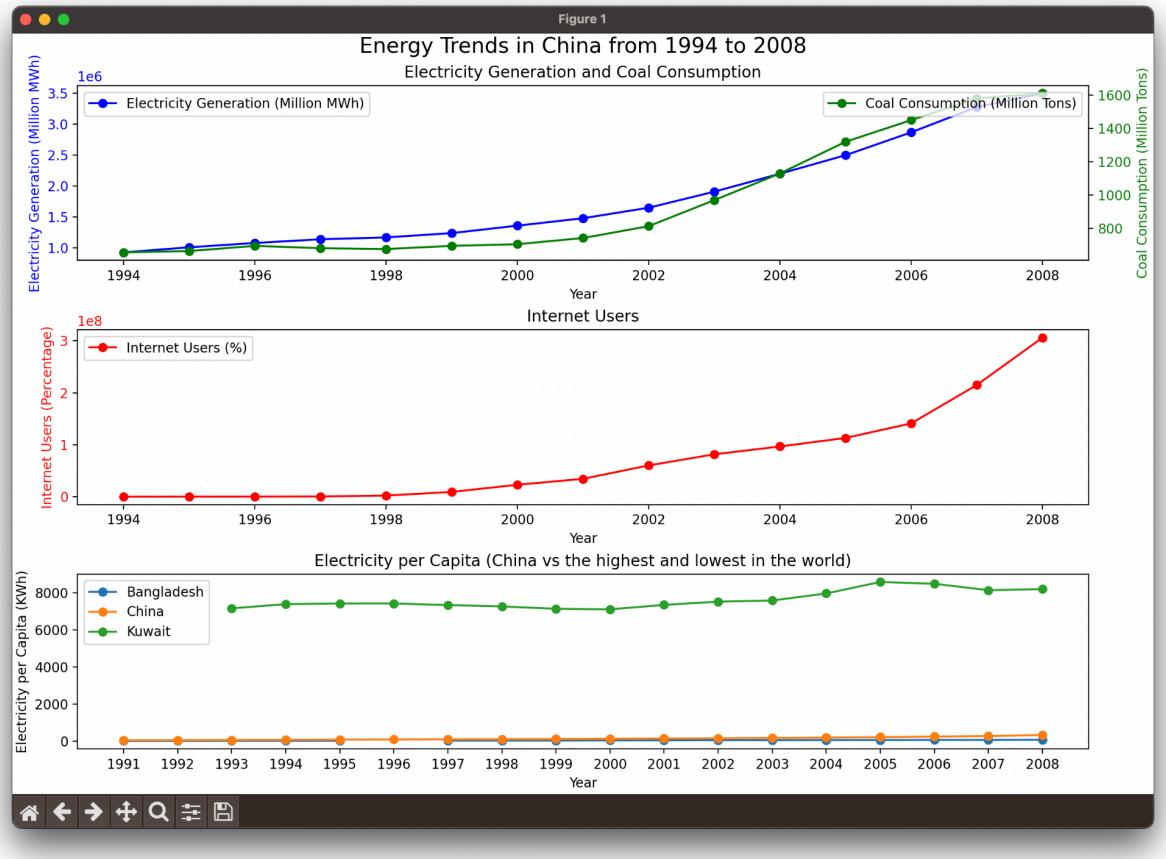
Overall Insight 1

Use the result generated by selecting `1. Life Quality ` and `China` as an example. The data reveals that positive trends in GDP per capita, cell phone usage, and internet access are correlated with improvements in life expectancy. Economic growth and technological advancements are key contributors to improving life quality in China. As the country's economy expands and technology becomes more integrated into daily life, the overall standard of living and health outcomes improve. In terms of life expectancy, China is slightly above the world average.



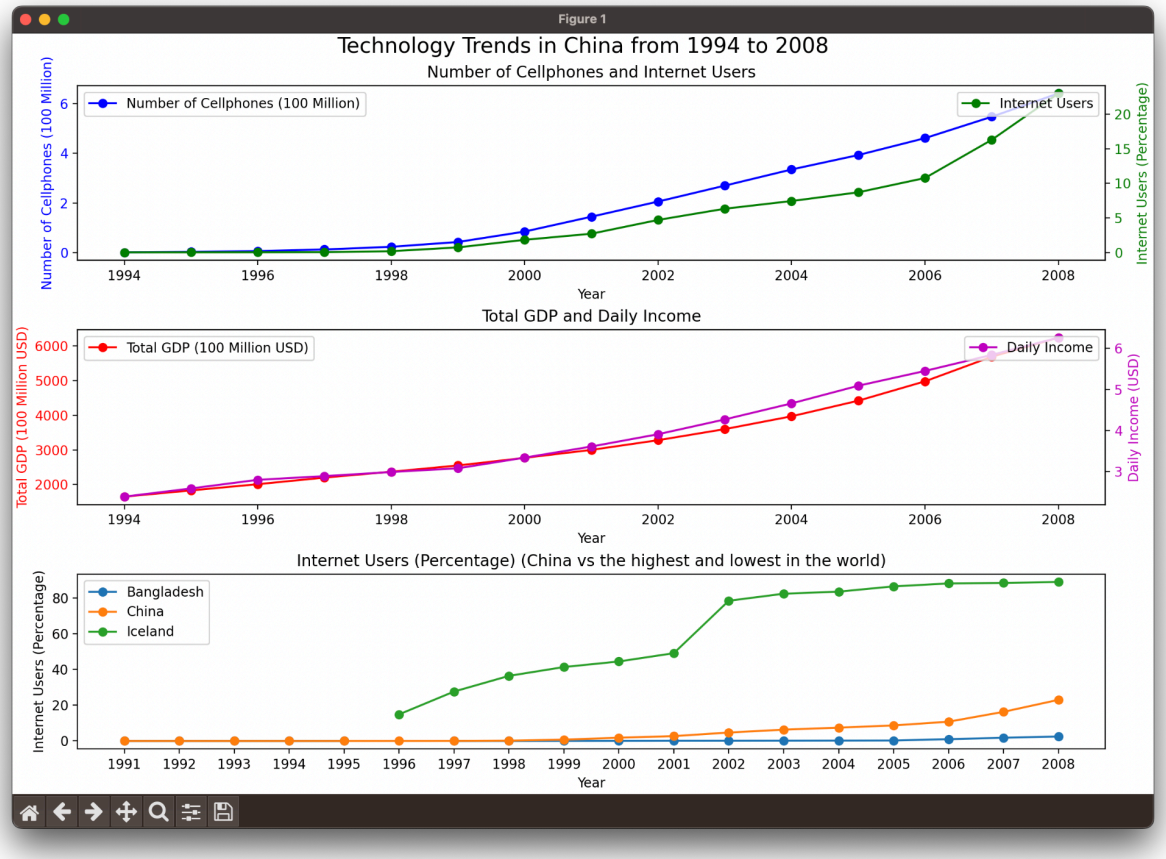
Overall Insight 2

Use the result generated by selecting `2. Economy` and `China` as an example. The analysis of economic indicators shows a steady increase in total GDP and population, indicating robust growth. Fluctuating but manageable inflation rates suggest a stable economic environment. We can notice that with inflation peaking each time, the population will also stagnate. The concurrent growth in GDP and population points to improved productivity and economic capacity, reflecting a positive economic trend. Regarding GDP per capita, China is sitting at the lower end of the world average.



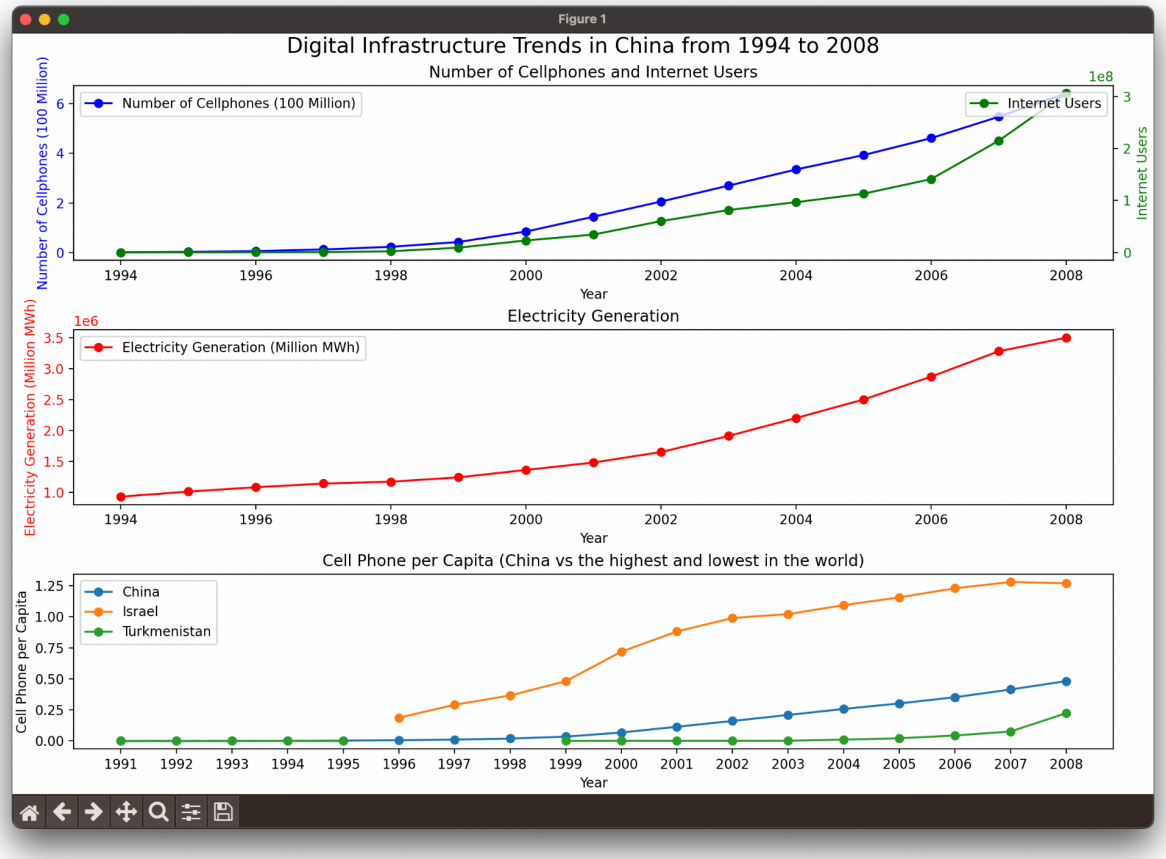
Overall Insight 3

Use the result generated by selecting `3. Energy` and `China` as an example. The energy sector in China has shown significant growth in electricity generation, indicating the country's expanding energy infrastructure and increasing energy demands. The continued dependence on coal as a primary energy source is evident from the rise in coal consumption. Additionally, the increase in internet penetration rates suggests improved access to digital infrastructure, aligning with trends in energy consumption. In terms of electricity per capita, China is sitting at the lower end of the world average.



Overall Insight 4

Use the result generated by selecting `4. Technology` and `China` as an example. The rapid increase in the number of cell phones and internet users illustrates widespread technological adoption and improved connectivity. The correlation between GDP growth and technological advancements indicates that technological progress is a key driver of economic development. Furthermore, rising daily income levels suggest enhanced standards of living, likely facilitated by the interplay of technological and economic advancements. In terms of the percentage of internet users, China is catching up with the world average.



Overall Insight 5

Use the result generated by selecting `5. Digital Infrastructure` and `China` as an example. The increasing number of cell phones and internet users highlights improved digital connectivity. The rise in electricity generation supports the infrastructure needed for this growth. Enhanced digital infrastructure is vital for broader economic and social development, as evidenced by rising internet penetration rates. In terms of cell phones per capita, China is catching up with the world average.

Sources

Datasets:

[1] Gapminder, “Cell Phones (Total),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 14, 2024).

[2] Gapminder, “Coal Consumption (Total),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 14, 2024).

[3] Gapminder, “Electricity Generation (Total),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[4] Gapminder, “Residential Electricity Use (Total),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[5] Gapminder, “Population,” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[6] Gapminder, “Inflation (Annual %),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[7] Gapminder, “Total GDP (US$, inflation-adjusted)” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[8] Gapminder, “Average Daily Income, $/Person/Day, Inflation- & Price Adjusted,” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[9] Gapminder, “Life Expectancy (At Birth),” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

[10] Gapminder, “Number Of People Using Internet,” Gapminder, https://www.gapminder.org/data/ (accessed Jun. 1, 2021).

Code reference:

[1]“Converting K and M to numerical form in Pandas DataFrame,” *www.skytowner.com*. https://www.skytowner.com/explore/converting_k_and_m_to_numerical_form_in_pandas_dataframe (accessed Jun. 17, 2024).