assignment-1

September 9, 2024

1 Problem 1

1.1 Setup: importing libraries and reading the dataset

[23]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

[24]:
```
df = pd.read_csv("../data/swedish_population_by_year_and_sex_1860-2022.csv")
df
```

[24]:

	age	sex	1860	1861	1862	1863	1864	1865	1866	1867	…	\
0	0	men	60589	59797	62371	61515	61931	60998	63036	58645	…	
1	0	women	58837	58136	60041	59384	60100	59622	60874	56586	…	
2	1	men	56001	54544	52933	55776	57346	57776	57017	59071	…	
3	1	women	54833	53762	52282	54500	55823	56641	56263	57539	…	
4	2	men	52502	54062	51613	50710	53743	55227	55696	55254	…	
..	…	…	…	…	…	…	…	…	…	…	…	
217	108	women	0	0	0	0	0	0	0	0	…	
218	109	men	0	0	0	0	0	0	0	0	…	
219	109	women	0	0	0	0	0	0	0	0	…	
220	110+	men	0	0	0	0	0	0	0	0	…	
221	110+	women	0	0	0	0	0	0	0	0	…	

	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
0	58649	59584	59994	61005	59899	60032	59476	58485	58692	54095
1	55359	56296	55884	58018	56715	56807	55907	55104	55971	51091
2	59039	59489	60640	61352	62531	60973	60993	60058	59195	59411
3	55884	56083	57292	57216	59444	57789	57783	56533	55855	56712
4	58721	59807	60292	61817	62258	63361	61598	61348	60596	59723
..	…	…	…	…	…	…	…	…	…	…
217	6	8	5	3	3	10	7	9	4	8
218	0	0	0	0	3	0	0	0	0	0
219	4	2	6	3	3	1	6	2	6	1
220	1	1	1	1	0	3	0	0	0	0
221	2	4	2	5	2	2	1	2	1	3

[222 rows x 165 columns]

1.2 Part I: Dependency ratio

Creating a new dataframe containing the number of people of a specific age for a specific year:

[25]:
```
age_df = df.drop(columns='sex').groupby("age").sum()
```

Generating a support vector with all the ages in the dataset:

[26]:
```
age = pd.to_numeric(age_df.index.str.replace("+", ""))
```

Calculating the number of children by summing every age under 15 years old:

[27]:
```
children_count = age_df[age <= 14].sum()
```

Doing the same thing for the elderly:

[28]:
```
elderly_count = age_df[age >= 65].sum()
```

And for the labor force:

[29]:
```
labor_count = age_df[(age >= 15) & (age <= 64)].sum()
```

Calculating the dependency ratio as:

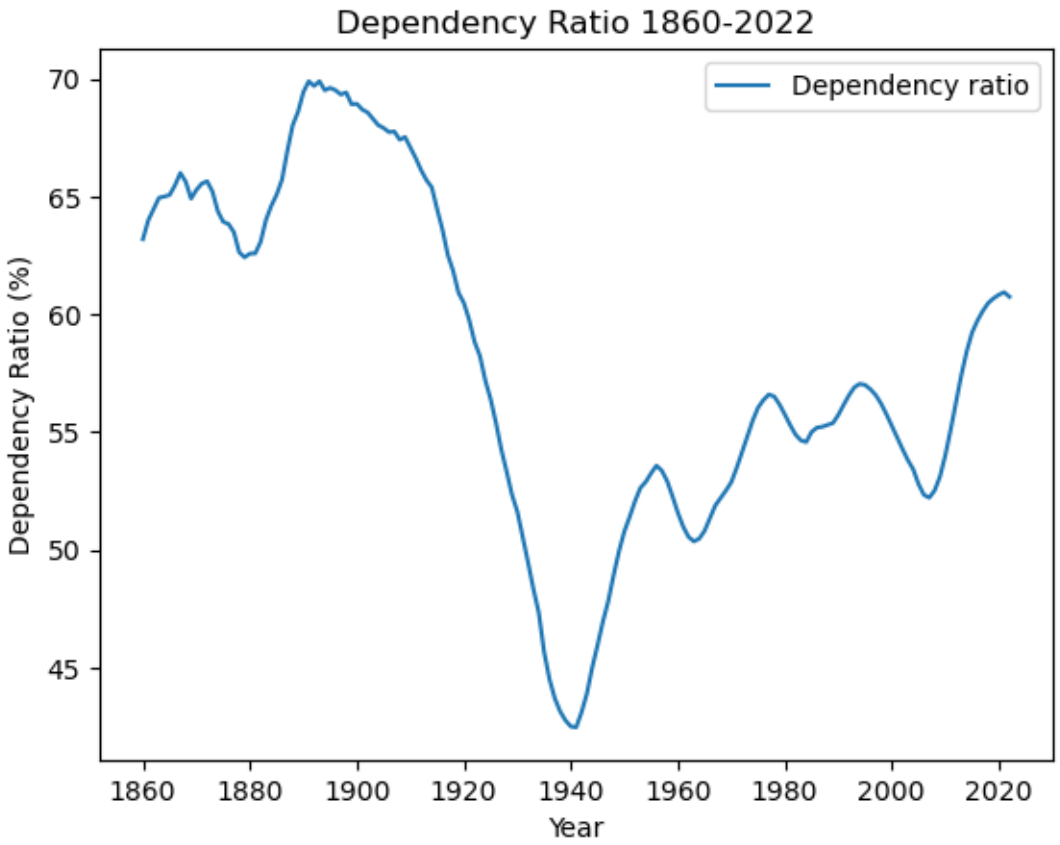
$$100 * \frac{\text{children\_count} + \text{elderly\_count}}{\text{labor\_count}}$$

[30]:
```
dependent_count = children_count + elderly_count
dependency_ratio = (dependent_count / labor_count)*100
```

Plotting the ratio:

[31]:
```
_, ax = plt.subplots()
ax.plot(dependency_ratio, label="Dependency ratio")
ax.set_title('Dependency Ratio 1860-2022')
ax.set_xlabel('Year')
ax.set_ylabel('Dependency Ratio (%)')
ax.set_xticks(range(0, len(dependency_ratio), 20))
ax.legend()

plt.savefig("dependency_ratio.png")
```



1.3 Part II: Population fractions

Calculating the total amount of a particular age group relative to the total population:

[32]:
```
total_count = age_df.sum()
children_fraction = children_count / total_count
elderly_fraction = elderly_count / total_count
dependent_fraction = dependent_count / total_count
```

Plotting the fractions:

[33]:
```
_, ax = plt.subplots()
ax.plot(children_fraction, label="Children")
ax.plot(elderly_fraction, label="Elderly")
ax.plot(dependent_fraction, label="Dependent Population")
ax.set_title('Swedish Population Categories Fractions 1860-2022')
ax.set_xlabel('Year')
ax.set_ylabel('Fraction')
ax.set_xticks(range(0, len(children_fraction), 20))
ax.legend(loc='lower right')
```

```
plt.savefig("population_fractions.png")
```