DIT407 Introduction to data science and AI - Assignment 1

Reynir Siik
reynir.siik@gmail.com

Franco Zambon
guszamfr@student.gu.se

2024-01-22

Abstract

This report will show how to compile a long series of population data into a diagram that visualises the dependency ratio [1] for the population of Sweden during 1860 to 2022. The data contained in a CSV-file will be loaded by Pandas. By slicing the data, age categories will be copied into numpy and the result will be plotted. A brief discussion of the graphs in the light of major events and general trends concerning population in western countries will attempt to interpret the fluctuations and trends in the plots.

Problem 1: Dependency ratio

1 Introduction

Abbreviated citation from assignment:

The dependency ratio is a demographic statistic that represents the ratio of dependent population and the productive population. The most commonly used definition (such as the one used by the World Health Organization [1], is as follows.

The population is divided into three classes by age: people aged 0–14 are considered children, people aged 15–64 are considered labor force, and people aged 65+ are considered the elderly. The children and the elderly make up the dependent population, whereas the labor force makes up the productive population.

The dependency ratio is usually expressed in units of the number of dependent people per 100 people in labor force, that is,

$$\text{Dependency ratio} = 100\,\frac{\#\text{children} + \#\text{elderly}}{\#\text{labor force}}$$

This statistic is of macro economical interest in understanding the pressure exhibited by the demographic structure of a country on national economy. The underlying assumption is that the productive population must sustain the dependent population through labor.

2 Reading and calculating the dataset

The data is stored in a comma separated values file. The file is read by using Pandas. Pandas organizes data in series. Series are a one-dimensional labeled array holding data

of any type [2].

Since the age column is to be used for comparison all values are converted to the datatype integer. Since the highest age value, 110, represents people of age 110 and higher, this value is represented by 110+. The plus sign prevents conversion of the value to integer so it gets special attention in the code in order to remove the plus sign.

```
df = pd.read_csv("swedish_population_by_year_and_sex_1860-2022.csv")
##
df['age'] = df['age'].replace('110+','110')
df['age'] = df['age'].astype(int)
```

Selection of series is done by choosing the corresponding series from age column and summing the columns for each year. The compiled series is stored in numpy matrices.

```
## Sum up population categorys                                          ##
## Children    ##############################################################
np_children = df[ df['age']<15].loc[:, first_yr:last_yr].sum()
## Elderly     ##############################################################
np_elderly = df[ df['age']>64].loc[:, first_yr:last_yr].sum()
## Labor force ##############################################################
np_labor = df[(df['age']<65) & (df['age']>14)].loc[:, first_yr:last_yr].sum()
## Total population ########################################################
np_total = df.loc[:, first_yr:last_yr].sum()
```

With the compiled series the calculation of dependency ratio for each year can be done and plotted in the diagram.

3 Graph of dependency ratio

The dependency ratio graph has a maximum in 1891, after which the ratio decreases and hits the minimum in 1941. After the minima the graph has an increasing trend with a few dips that will be discussed later.

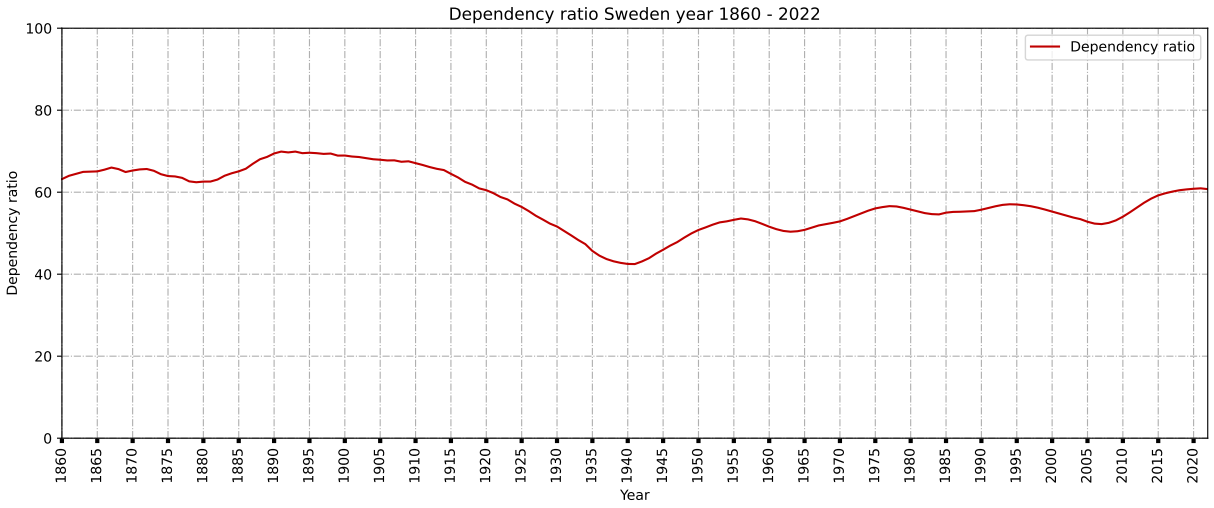

Figure 1: Dependency ratio

4 Graphs of fractions of population

The calculation of fractions of dependent people results in three graphs in figure 2, and shows some overall trends. The fraction of children is decreasing, with a significant dip in 1941. The fraction of elderly is increasing, interrupted by a steady interval between 1987 and 2005. After 2005 the fraction of elderly is larger than the fraction of children.

When the graphs of children and elderly are summed up the resulting graph resembles the graph of dependency ratio.

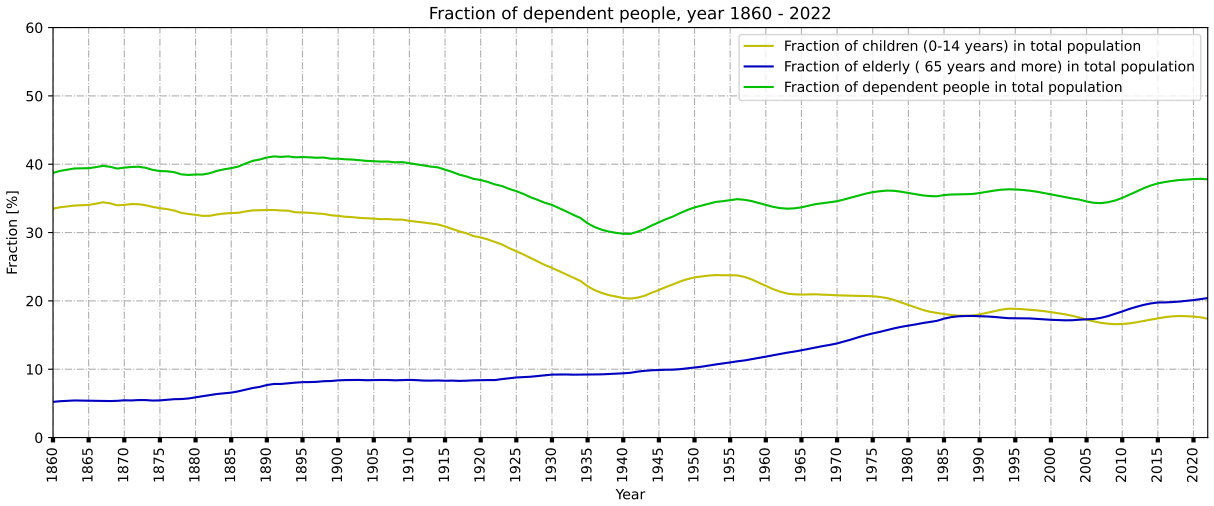

Figure 2: Fractions of population [%]

5 Discussion

The graphs of the Swedish population will be discussed in connection to three major events that have had an effect on the Swedish population.

The first event was the huge migration from Sweden between the years 1851 and 1910. Of those who were born in the later half of the 19th century 20% of the male and 15% of female population migrated from Sweden to America [3].

The second event was the first world war that started in 1914 [4]. Sweden experiences food shortage bordering to famine due to bad harvest in 1917, and the embargo against Germany, that effects Sweden also. The graph of fraction of children shows a decreasing trend. This trend continues until 1940.

The third event, the second world war. During the war the trend of lower fraction of children is turned and starts to increase [5]. The so called baby boom have caused ripple on the graph since then.

The dip of childbirths that can be observed in 1941 causes subsequent dips with increasing interval. That may have two reasons. First and foremost that the people who are children during 1941 will have their children the years to come. That causes a dip to recur since there are fewer people that can give birth. The dip also comes with increasing interval, which may be caused by the trend of having children later in life then before [6].

References

[1] W. H. Organization, *The global health observatory indicator metadata registry list: Dependency ratio*, Retrieved 2023-10-25, 2023. [Online]. Available: https://www.who.int/data/gho/indicator-metadata-registry/imr-details/1119.

[2] Pandas, *10 minutes to pandas*, Date: Jan 20, 2024 Version: 2.2.0. Retrieved 2024-01-22, 2024. [Online]. Available: https://pandas.pydata.org/docs/user_guide/10min.html.

[3] S. centralbyrån, *Så påverkade utvandringen till amerika sveriges befolkning*, Retreived 2024-01-23, 2013. [Online]. Available: https://www.scb.se/hitta-statistik/artiklar/2013/sa-paverkade-utvandringen-till-amerika-sveriges-befolkning/.

[4] Wikipedia, *Sverige under första världskriget — wikipedia*, [Online; hämtad 23-januari-2024], 2023. [Online]. Available: //sv.wikipedia.org/w/index.php?title=Sverige_under_f%C3%B6rsta_v%C3%A4rldskriget&oldid=54227950.

[5] S. centralbyrån, *Sveriges folkmängd från 1749 och fram till idag*, Retreived 2024-01-23, 2017. [Online]. Available: https://www.scb.se/hitta-statistik/artiklar/2017/sveriges-folkmangd-fran-1749-och-fram-till-idag/.

[6] F. Youcefi, *Förstföderskor blir allt äldre*, Retreived 2024-01-23, Sveriges Television AB, May 24, 2017. [Online]. Available: https://www.svt.se/nyheter/lokalt/vast/forstfoderskor-blir-allt-aldre.

A Complete source code

This is the complete listing of the source code.

```
# Reynir Siik, Franco Zambon
# DIT407 lp3 2024-01-15
# Assignment 1
# Problem 1
# Dependency ratio
#
# data file: swedish_population_by_year_and_sex_1860-2022.csv
#########################################################################
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator # to change ticks of x-axis

import numpy  as np
import pandas as pd
#########################################################################
## Set timespan and load data
##
first_yr = '1860'
last_yr  = '2022'
df = pd.read_csv("swedish_population_by_year_and_sex_1860-2022.csv")

#########################################################################
## Change datatype in 'age' column to int in order to be able compare
##
## This could be done in a more general way, but for this dataset the
## following approach does the job...
##
df['age'] = df['age'].replace('110+','110')
df['age'] = df['age'].astype(int)

#########################################################################
## Sum up population categorys                                         ##
## Children    ###############################################################
np_children = df[ df['age']<15].loc[:, first_yr:last_yr].sum()

## Elderly     ###############################################################
np_elderly = df[ df['age']>64].loc[:, first_yr:last_yr].sum()

## Labor force ###############################################################
np_labor = df[(df['age']<65) & (df['age']>14)].loc[:, first_yr:last_yr].sum()

## Total population ##########################################################
np_total = df.loc[:, first_yr:last_yr].sum()

## Make diagrams    ##########################################################
##
## Figure 1
fig, ax = plt.subplots(figsize=(12,5), layout='constrained')

ax.grid(True, linestyle='-.')
ax.tick_params(axis='x', rotation=90, labelsize='medium', width=3)
ax.xaxis.set_major_locator(MultipleLocator(5)) # change ticks of x-axis

ax.set_xlim(0, 10 * ((int(last_yr) / 10) - int(first_yr)/10))
ax.set_ylim(0, 100)

ax.set_xlabel("Year")
ax.set_ylabel("Dependency ratio")

plt.title("Dependency ratio Sweden year " + first_yr + " - " + last_yr)
ax.plot(100 * (np_children + np_elderly) / np_labor,
```

```
61      c="#C00000", label='Dependency␣ratio')
62  ax.legend()
63
64  ## Figure 2
65
66  fig2, ax2 = plt.subplots(figsize=(12,5), layout='constrained')
67  ax2.grid(True, linestyle='-.')
68  ax2.tick_params(axis='x', rotation=90, labelsize='medium', width=3)
69  ax2.xaxis.set_major_locator(MultipleLocator(5)) # change ticks of x-axis
70
71  ax2.set_xlim(0, 10 * ((int(last_yr) / 10) - int(first_yr)/10))
72  ax2.set_ylim(0, 60)
73
74  ax2.set_xlabel("Year")
75  ax2.set_ylabel("Fraction␣[%]")
76
77  plt.title("Fraction␣of␣dependent␣people,␣year␣" + first_yr + "␣-␣" + last_yr)
78
79  ax2.plot(100 * np_children / np_total,
80      c="#C0C000",
81      label='Fraction␣of␣children␣(0-14␣years)␣in␣total␣population')
82
83  ax2.plot(100 * np_elderly / np_total,
84      c="#0000C0",
85      label='Fraction␣of␣elderly␣(␣65␣years␣and␣more)␣in␣total␣population')
86
87  ax2.plot(100 * (np_children + np_elderly) / np_total,
88      c="#00C000",
89      label='Fraction␣of␣dependent␣people␣in␣total␣population')
90
91  ax2.legend()
92
93  plt.show()
```