

Cancer incidence trends in young adults

Introduction

The idea of this project comes from a conversation I recently have with a friend of mine who worked in a cancer retreat center. The friend is a medical professional who worked at the center for many years. He noticed that over the past 10 years, there have been more relatively young patients in the center than it was before. The friend was inclined to attribute this observation to the deteriorating quality of air, food, and other environmental factors. I skeptically suggested that there are other possible explanations of this phenomenon. 10 years ago, the friend himself was younger and could perceive typical patients as older.

Cancer is a class of diseases in which some of the body's cells grow uncontrollably and spread to other organs and tissues. Most cancers form a tumor. According to modern concepts, cancer is caused by changes to genes that control the way our cells function, especially how they grow and divide. Despite intensive research and development of new treatments, cancer remains the leading causes of death worldwide. In 2018, there were 18.1 million new cases and 9.5 million cancer-related deaths worldwide. By 2040, the number of new cancer cases per year is expected to rise to 29.5 million and the number of cancer-related deaths to 16.4 million [1].

Cancer can be considered an age-related disease because the incidence of most cancers increases with age, rising more rapidly beginning in midlife. Despite the fact that the disease can occur at any age, more than half of all cancers occurred in adults aged ≥ 65 years [2]. Therefore, for a long time, cancer was considered a disease predominantly affecting the elderly. Given this age specificity, a possible increase in the number of diseases among younger age groups may be of great interest.

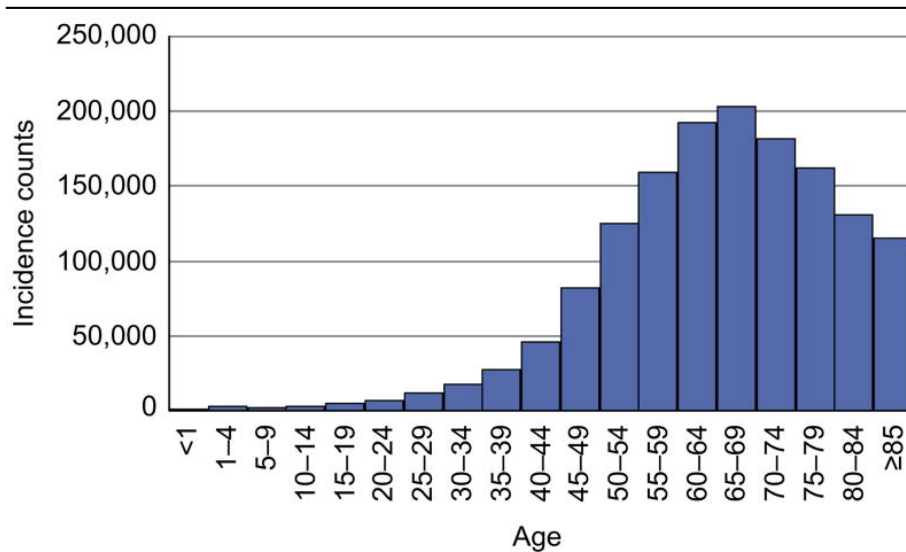


Figure 1. Invasive cancer incidence, by age, U.S., 2009 [2]

As a software developer I am interested in Digital health. The idea of using information technologies to enhance the efficiency of healthcare seems very promising. Collection and analysis of health data using the data-science approach could potentially improve our understanding of diseases such as cancer. Therefore, I decided to further investigate my friend's observation in this project. Using available data and data-science approach it seems possible to determine if young adults have become more often diagnosed with cancer in recent decades.

Literature Review

Before setting goals and objectives, I decided to study the available information on this topic. It is important to understand what has already been investigated and what has not yet been explored. I found several articles on the Internet in which the issue is studied. Below I provide a summary and key findings that are relevant in the context of this study.

Incidence trends for twelve cancers in younger adults—a rapid review. Br J Cancer 126, 1374–1386 (2022). [5]

This paper analyzed the epidemiological information of some types of cancer in young adult patients, and came to the following conclusion: "Overall, this review provides evidence that some cancers are increasingly being diagnosed in younger age groups, although the mechanisms remain unclear." [5] This is a meta-analysis of existing studies on different types of cancer, but my goal is to explore the big picture across all types of cancer. The sources used in the article were mainly originated from the United States.

Trends in Cancer Incidence in US Adolescents and Young Adults, 1973-2015 [6].

Some findings from this paper: "In this serial cross-sectional, US population-based study using cancer registry data from 497 452 AYAs, the rate of cancer increased by 29.6% from 1973 to 2015, with kidney carcinoma increasing at the greatest rate. Breast carcinoma and testicular cancer were the most common cancer diagnoses for female and male AYAs, respectively." [6]

The authors conclude: "In this cross-sectional, US population-based study, cancer in AYAs was shown to have a unique epidemiological pattern and is a growing health concern, with many cancer subtypes having increased in incidence from 1973 to 2015. Continued research on AYA cancers is important to understanding and addressing the distinct health concerns of this population." [6] AYA stands for Adolescents and Young Adults.

The findings from this paper also support the idea that the increase in the number of diseases among young people in the United States has natural causes. However, the article is again focused on the United States.

Cancer Stat Facts: Cancer Among Adolescents and Young Adults (AYAs) (Ages 15–39) [7].

The results of this work once again confirm the increase in the number of diseases among young adults in the United States: "Using statistical models for analysis, rates of new cancer cases of any site among AYAs have been rising on average 0.3% each year over 2010–2019, the last 10 years of available data." [7]. This study is based on a database provided by National Cancer Institute of US. This is a high fidelity database, but it covers only the U.S. This work uses great data visualization methods that I want to use for inspiration.

As a result of a review of the literature available on the Internet on this topic, I found that there is ample evidence of an increase in the number of diseases among young adults in the United States. However, there are no studies on other regions and on worldwide population.

Aims and Objectives

According to the available literature, there is a trend in the United States that young people are more likely to get cancer in recent decades. However, the question remained unexplored whether there is such a trend around the world. My goal within this project is to fill this gap. I want to know if this is a global trend or this is specific for certain regions.

The literature cited above suggests that the causes for the trend observed in the US remain unclear. If a similar trend takes place in the rest of the world, this may help in finding its causes. Since in this case the reasons may be common to the entire planet, and not specific to a particular region. However, the search for the causes is beyond the scope of this study.

For the purpose of this project, a young adult is considered to be between the ages of 15 and 44 (inclusive). This range was chosen because people younger than 15 have other types of cancer with different epidemiological dynamics that are outside the scope of this study [3]. The age group over 45 also has its own epidemiological dynamics. It has long seen an increase in morbidity, but it is attributed mainly to an increase in overall life expectancy and a decrease in mortality from other causes [4]. This is also outside the scope of this study.

Dataset

Requirements

To meet the goals of the project, the dataset must include worldwide cancer incidence statistics. Data must have at least two dimensions: the numbers should be broken down by year of diagnosis and a patient's age group. To pinpoint the trend, we need data for at least two decades. The data should be in machine-readable format.

Datasets considered

There are several organizations that collect statistics on cancer. In particular, the previously mentioned National Cancer Institute (NCI) provides SEER database. SEER (Surveillance, Epidemiology, and End Results) is an authoritative source for cancer statistics in the United States. As previously mentioned, this is an open and high fidelity database, but it covers only the U.S. so it is not suitable for our research. Another problem is that this dataset is not generally available in machine-readable format.

Another considered dataset is Cancer registration statistics for England provided by Office for National Statistics. The data includes cancer diagnoses and age-standardised incidence rates for all types of cancer by age and sex. It is available in a machine-readable format as Excel tables. Despite the fact that these are data for one geographic region, from these files I get an idea in what form such statistics can be provided at all.

Chosen dataset

After research of different sources concerning the subject I found that there is an organization International Agency for Research on Cancer (IARC) that is a part of the World Health Organization of the United Nations. Its role is to conduct and coordinate research into the causes of cancer. It also collects and publishes surveillance data regarding the occurrence of cancer worldwide.

This data comes in datasets called Cancer Incidence in Five Continents (CI5). CI5 is the result of a long collaboration between the International Agency for Research on Cancer and the International Association of Cancer Registries. The series of monographs, published approximately every five years, has become the reference source of data on the international incidence of cancer. [8]

The dataset is publicly available and can be used for research and educational purposes. On the home page it explicitly states "Materials (fact sheets, maps, estimates or data) may be used "as is" for research, educational or other non-commercial purposes, but the corresponding reference must be cited in all cases." Data is completely anonymized, it does not contain any personal information about patients.

The whole dataset consists of separate publications (volumes) with data for different periods. These volumes are identified by Roman numerals (V, VI, VII, etc). The first 6 volumes (V, VI, VII, VIII, IX, and X)[9][10][11][12][13][14] cover the period from 1973 till 2007. The last volume (XI)[15] covers the period from 2008 till 2012. These publications are PDF files with very detailed reports on the incidence of cancer in different countries. Files with detailed source data are also provided along with reports. They are of primary interest to us.

The first 6 volumes are considered archived, they are downloadable on this page. There are PDF reports and ZIP files with tabulated detailed (source) data. The latest volume is downloadable on this page. This dataset covers a large period of time from 1973 till 2012. The raw data is provided in machine-readable format (CSV, tabulated). There are detailed data on date of diagnosis, patient's age group, sex, geographic region, and cancer type. This is the most comprehensive source of information on cancer incidences in the world. Thus, this dataset is fully suitable for this study.

Method

After a more detailed examine of the dataset files, some problems were identified. The first problem is size of the dataset: more than 215 MB unzipped. This is much more than the stated limit. The second problem is the extreme heterogeneity of the data. Each of these seven volumes has a different data format. Some files are in the form CSV, some are tab-separated. Different field names and other differences.

To overcome these difficulties, I decided to divide the data analysis process into several stages:

Preprocessing

At this stage, we create a dataset from the raw CI5 files. The raw CI5 files are individual reports (volumes) that were published at different times. These volumes have different data format. So we have separate sections for each volume (Volume V, Volume VI, Volume VII, etc.) We download raw files, unzip, cleansing, aggregate, and save only necessary information to a CSV file *dataset.csv* that serve as a source for the further stages. This file is relatively small and can be cached, so there is no need to run this stage more than once.

Processing

At this stage I no longer touch the raw CI5 files. The only source of the information is the intermediate dataset we generated at the Preprocessing stage. It is more precise cleansing and preparation for analysis.

Analysis

I want to explore the dataset through different lenses, in particular:

- Dynamics of the total number of registered cancer cases in the world in the recent decades.
- Dynamics of registered cancer cases in the world in young adults in the recent decades.
- Dynamics of the percentage of young people among all cases in the recent decades.

My assumption is that the evaluation of these metrics will be enough to answer this project's question: Is there a world trend that young people are more likely to be diagnosed with cancer in the recent decades.

Disclaimer

This study touches on important topics in the diagnosis and epidemiology of cancer. Research in this topic usually has high requirements for data quality and fact checking. The framework for this research does not imply the level of verification normally expected from researches in public health. Considering the possible ethical issues, it is necessary to explicitly state that the results of this study and all the facts presented are provided for informational purposes only. The materials in this study are not intended for use in any clinical setting.

Import necessary libraries

```
!pip install pandas==1.4.2
!pip install matplotlib==3.5.2

import os
import requests
import zipfile
import re
import io
import codecs
from urllib.parse import urlparse
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

Preprocessing

At this stage, we create a dataset from the raw CI5 files [9][10][11][12][13][14][15].

As I mentioned above, the raw data consists of 7 separate zip files (volumes). The total size of these files are more than 215 MB unzipped which exceeds the allocated limit. At the preprocessing stage, I download the original CI5 files, remove all irrelevant information from them, and merge the result into one file *dataset.csv*. Thus, the result of this stage is one relatively small file *dataset.csv* that serve as a data source for the further stages.

Important: If you have dataset.csv you can skip the whole Preprocessing section and go right to the Processing section.

First, download "detailed data" files from <https://ci5.iarc.fr/ci5i-x/pages/download.aspx> and <https://ci5.iarc.fr/CI5-XI/Pages/download.aspx> for each volumes:

```
urls = (
    "https://ci5.iarc.fr/ci5i-x/old/vol5/CI5-V.zip",
    "https://ci5.iarc.fr/ci5i-x/old/vol6/CI5-VI.zip",
    "https://ci5.iarc.fr/ci5i-x/old/vol7/CI5-VIIId.zip",
    "https://ci5.iarc.fr/ci5i-x/old/vol8/CI5-VIIId.zip",
    "https://ci5.iarc.fr/ci5i-x/old/vol9/CI5-IXd.zip",
    "https://ci5.iarc.fr/CI5-X/CI5-Xd.zip",
    "https://ci5.iarc.fr/CI5-XI/CI5-XI.zip"
)

# create a new dir CI5 to save volume files
os.makedirs("CI5", exist_ok=True)

def download_if_not_exists(url, path):
    """Download a file from the url and save to the path only if file does not already exist"""
    if os.path.exists(path):
        print(f"File {path} exists in cache")
    else:
        print(f"Downloading {url}...")
        response = requests.get(url)
        # check for HTTP errors
        assert response.ok, f"Failed to download a file {url}. Please try again."

        print(f"Save to {path}")
        open(path, "wb").write(response.content)

def unzip(file_path, target_dir):
    """Extract ZIP file file_path to target_dir"""
    print(f"Extract {file_path} to {target_dir}")
```

```

with zipfile.ZipFile(file_path, "r") as zip_ref:
    zip_ref.extractall(target_dir)

# Download all files from urls, save it to CI5 folder, unzip
for url in urls:
    url_path = urlparse(url).path
    file_name = os.path.basename(url_path)
    file_path = os.path.join("CI5", file_name)
    volume_name = os.path.splitext(file_name)[0]
    volume_path = os.path.join("CI5", volume_name)

    if os.path.exists(volume_path):
        print(f"Volume {volume_name} exists in cache")
    else:
        download_if_not_exists(url, file_path)
        unzip(file_path, os.path.join("CI5", volume_name))

```

Volume V

This is the first CI5 volume covers period from 1973 till 1982. It is a zip archive with 4 files. The layout of the data files are:

- `cancers.txt` — List of diagnostic groups. It represents anatomical sites where cancer occurs.
- `registry.txt` — list of regions and years of observation
- `cases.csv` — Number of cases by sex, five-year age group and a cancer code from `cancers.txt`
- `pops.csv` — Number of people at risk by five-year age group

I will use Pandas dataframe to load and join data from `registry.txt` and `cases.csv`. Pandas looks like a suitable tool as it makes it easy to perform such operations on heterogeneous data. As a result of each volume processing step, we will get a dataframe in the following form:

Period	N0_4	N5_9	...	N85+	N_UNK
1973-1982	104	60	...	1513	50
1977-1981	97	40	...	10	50
...
1982-1982	55	22	...	436	18

- **Period** is a time interval arbitrary number of years. Periods can overlap with each other.
- **N0_4**, **N5_9** — are five-year age groups. There are the number of recorded cases for patients in each age groups per each period.
- **N85+** — is a special age group for patients over 85
- **N_UNK** — is a special age group for patients with unknown age

So, each row in this intermediate dataset is a time period (from__year-till__year).
Each column is a patient's age group.

```
# registry.txt is Tab-separated, while cases.csv is comma-separated
v_registry_df = pd.read_csv("CI5/CI5-V/registry.txt", sep="\t", index_col=0)
v_cases_df = pd.read_csv("CI5/CI5-V/cases.csv", index_col=0)

# it uses a registry ID as a primary key to join two dataframes
v_df = v_registry_df.join(v_cases_df, how="inner", lsuffix="_registry")

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
v_df["PERIOD"] = v_df["PERIOD_1"].astype(str) + '-' + v_df["PERIOD_2"].astype(str)

# remove columns with irrelevant information
v_df.drop(columns=['ETHNIC_GROUP_registry', 'PERIOD_1', 'PERIOD_2', 'NAME', 'ETHNIC_GROUP',
                  'SEX', 'CANCER', 'TOTAL'], inplace=True)

v_df = v_df.groupby(["PERIOD"]).sum()

# v_df # uncomment to display data for this volume
```

Volume VI

Reference: Curado. M. P., Edwards. B., Shin. H.R., Storm. H., Ferlay. J., Heanue. M. and Boyle. P., eds (2007) Cancer Incidence in Five Continents, Vol. IX IARC Scientific Publications, No. 160, Lyon, IARC.

The second volume covers a period from 1981 till 1989. Technically, this volume is almost the same as the previous. Except it has slightly different column names.

```
# registry.txt is Tab-separated, while cases.csv is comma-separated
vi_registry_df = pd.read_csv("CI5/CI5-VI/registry.txt", sep="\t", index_col=0, names=["REGIS
vi_cases_df = pd.read_csv("CI5/CI5-VI/cases.csv", index_col=0)

# it uses a registry ID as a primary key to join two dataframes
vi_df = vi_registry_df.join(vi_cases_df, how="inner", lsuffix="_registry")

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
vi_df["PERIOD"] = vi_df["PERIOD_1"].astype(str) + '-' + vi_df["PERIOD_2"].astype(str)

# remove columns with irrelevant information
vi_df.drop(columns=['PERIOD_1', 'PERIOD_2', 'NAME', 'SEX', 'CANCER', 'TOTAL'], inplace=True)

vi_df = vi_df.groupby("PERIOD").sum()

# rename column names to a standard form
vi_df.rename(columns={"N85": "N85+", "N_unk": "N_UNK"}, inplace=True)
```

```
# vi_df # uncomment to display data for this volume
```

Volume VIIId

This volume covers a period from 1983 till 1994. It has a different format than the previous two.

```
viid_registry_df = pd.read_csv("CI5/CI5-VIIId/registry.txt", index_col=0, names=["REGISTRY",  
viid_cases_df = pd.read_csv("CI5/CI5-VIIId/CI5VII.csv", names=["REGISTRY", "SEX", "CANCER_NUM"
```

```
# remove the first row as it is a broken header
```

```
viid_registry_df = viid_registry_df[1:]
```

```
# Extract years from NAME column using Regex
```

```
viid_registry_df = viid_registry_df["NAME"].str.extract(r"(.)\s+\((\d+)-(\d+)\)", expand=True)  
                    .rename(columns= {1: "PERIOD_1", 2: "PERIOD_2"})
```

```
# We have different AGE codes in this file, so we need to convert it to the standard form
```

```
AGE_GROUP_CODES_MAPPING = {1: "N0_4", 2: "N5_9", 3: "N10_14", 4: "N15_19", 5: "N20_24", 6: "N25_29",  
                             8: "N35_39", 9: "N40_44", 10: "N45_49", 11: "N50_54", 12: "N55_59",  
                             14: "N65_69", 15: "N70_74", 16: "N75_79", 17: "N80_84", 18: "N85_89"}
```

```
viid_cases_df["AGE"].replace(AGE_GROUP_CODES_MAPPING, inplace=True)
```

```
# Convert a values in AGE to columns with aggregated values from CASES_COUNT, reset the index
```

```
viid_cases_df = viid_cases_df.groupby(["REGISTRY", "AGE"])["CASES_COUNT"].sum().to_frame().reset_index()  
                    .pivot(index="REGISTRY", columns="AGE", values="CASES_COUNT")
```

```
# Convert registry ID to int to be compatible with viid_cases_df's index to be able to join
```

```
viid_registry_df.index = viid_registry_df.index.astype(int)
```

```
# it uses a registry ID as a primary key to join two dataframes
```

```
viid_df = viid_registry_df.join(viid_cases_df, how="inner", lsuffix="_registry")
```

```
# join two fields with start and end years into one field PERIOD in a format "1995-1998"
```

```
viid_df["PERIOD"] = viid_df["PERIOD_1"].astype(str) + '-' + viid_df["PERIOD_2"].astype(str)
```

```
# remove columns with irrelevant information
```

```
viid_df.drop(columns=[0, "PERIOD_1", "PERIOD_2"], inplace=True)
```

```
viid_df = viid_df.groupby("PERIOD").sum()
```

```
# viid_df # uncomment to display data for this volume
```

Volume VIIId

This volume covers a period from 1991 till 1998. It has a similar format that the previous one. Except it has slightly different registry.txt format.

```
viiid_registry_df = pd.read_table("CI5/CI5-VIIId/registry.txt", index_col=0)
viiid_cases_df = pd.read_csv("CI5/CI5-VIIId/CI5-VIIId.csv", names=["REGISTRY", "SEX", "CANCE"]

# Extract years from index column using Regex
viiid_registry_df = viiid_registry_df.index.str.extract(r"\s*(\d+)\s+(.*)\((\d+)(-\d+)\)?\s")
viiid_registry_df = viiid_registry_df.rename(columns= {0: "REGISTRY", 1: "NAME", 2: "PERIOD"})

# Use of AGE_GROUP_CODES_MAPPING from the previous volume to convert AGE to the standard format
viiid_cases_df["AGE"].replace(AGE_GROUP_CODES_MAPPING, inplace=True)

# Convert a values in AGE to columns with aggregated values from CASES_COUNT, reset the index
viiid_cases_df = viiid_cases_df.groupby(["REGISTRY", "AGE"])["CASES_COUNT"].sum().to_frame()
                        .pivot(index="REGISTRY", columns="AGE", values="CASES_COUNT")

# Convert registry ID to int to be compatible with viiid_cases_df's index to be able to join
viiid_registry_df.index = viiid_registry_df.index.astype(int)

# It uses a registry ID as a primary key to join two dataframes
viiid_df = viiid_registry_df.join(viid_cases_df, how="inner", lsuffix="_registry")

# Fix illegal values in the record for Taiwan
viiid_df.loc[81, "PERIOD_1"] = 1993
viiid_df.loc[81, "PERIOD_2"] = 1997

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
viiid_df["PERIOD"] = viiid_df["PERIOD_1"].astype(str) + '-' + viiid_df["PERIOD_2"].astype(str)

viiid_df = viiid_df.groupby(["PERIOD"]).sum()

# viiid_df # uncomment to display data for this volume
```

Volume IXd

This volume covers a period from 1996 till 2002. It uses a different data format when cancer cases from different registers stored in different files *CI5/CI5-IXd/{registry_id}.csv*.

```
ixd_registry_df = pd.read_table("CI5/CI5-IXd/registry.txt", names=["REGISTRY", "NAME"], index_col=0)

# Extract years from index column using Regex
ixd_registry_df = ixd_registry_df["NAME"].str.extract(r"\s*(.*)\s*\((\d+)-(\d+)\s)", expand=True)
                        .rename(columns= {0: "NAME", 1: "PERIOD_1", 2: "PERIOD_2"})
```

```

# Load case record files for each registry and append it to registry_dfs array
registry_dfs = []
for registry in idx_registry_df.index:
    df = pd.read_csv(f"CI5/CI5-IXd/{registry}.csv", names=["SEX", "CANCER_NUMBER", "AGE", "CANCER_TYPE"])
    df['REGISTRY'] = registry
    registry_dfs.append(df)

# Merge all register files into one dataframe
idx_cases_df = pd.concat(registry_dfs)

# Use of AGE_GROUP_CODES_MAPPING from the previous volume to convert AGE to the standard format
idx_cases_df["AGE"].replace(AGE_GROUP_CODES_MAPPING, inplace=True)

# Convert a values in AGE to columns with aggregated values from CASES_COUNT, reset the index
idx_cases_df = idx_cases_df.groupby(["REGISTRY", "AGE"])["CASES_COUNT"].sum().to_frame().reset_index()
idx_cases_df.pivot(index="REGISTRY", columns="AGE", values="CASES_COUNT")

# It uses a registry ID as a primary key to join two dataframes
idx_df = idx_registry_df.join(idx_cases_df, how="inner", lsuffix="_registry")

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
idx_df["PERIOD"] = idx_df["PERIOD_1"].astype(str) + '-' + idx_df["PERIOD_2"].astype(str)

idx_df = idx_df.groupby("PERIOD").sum()

# idx_df # uncomment to display data for this volume

```

Volume Xd

This volume covers a period from 2003 till 2007. It has a similar format that the previous one.

```

# A file CI5/CI5-XI/registry.txt contains unicode errors which cause error when read it directly
with codecs.open("CI5/CI5-Xd/registry.txt", 'r', 'utf8', errors="ignore") as ff:
    content = ff.read()

# load a dataframe from the string
xd_registry_df = pd.read_table(io.StringIO(content), names=["REGISTRY", "NAME"], index_col=0)

# Extract years from index column using Regex
xd_registry_df = xd_registry_df["NAME"].str.extract(r"\s*(.*)\s*((\d+)-(?:\d+,\d+-)?(\d+)\s+)", expand=True)
xd_registry_df.rename(columns={0: "NAME", 1: "PERIOD_1", 2: "PERIOD_2", 3: "PERIOD_3"}, inplace=True)

# Load case record files for each registry and append it to registry_dfs array
registry_dfs = []

```

```

for registry in xd_registry_df.index:
    df = pd.read_csv(f"CI5/CI5-Xd/{registry}.csv", names=["SEX", "CANCER_NUMBER", "AGE", "CA
    df['REGISTRY'] = registry
    registry_dfs.append(df)

# Merge all register files into one dataframe
xd_cases_df = pd.concat(registry_dfs)

# Use of AGE_GROUP_CODES_MAPPING from the previous volume to convert AGE to the standard for
xd_cases_df["AGE"].replace(AGE_GROUP_CODES_MAPPING, inplace=True)

# Convert a values in AGE to columns with agregated values from CASES_COUNT, reset the index
xd_cases_df = xd_cases_df.groupby(["REGISTRY", "AGE"])["CASES_COUNT"].sum().to_frame().reset
    .pivot(index="REGISTRY", columns="AGE", values="CASES_COUNT")

# It uses a registry ID as a primary key to join two dataframes
xd_df = xd_registry_df.join(xd_cases_df, how="inner", lsuffix="_registry")

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
xd_df["PERIOD"] = xd_df["PERIOD_1"].astype(str) + '-' + xd_df["PERIOD_2"].astype(str)

xd_df = xd_df.groupby(["PERIOD"]).sum()

# xd_df # uncomment to display data for this volume

```

Volume XI

The is the latest volume covers the period from 2008 till 2012. Technicaly, it differs from the previous one, uses the format of the old volumes.

```

# A file CI5/CI5-XI/registry.txt contains unicode errors which cause error when read it dire
with codecs.open("CI5/CI5-XI/registry.txt", 'r', 'utf8', errors="ignore") as ff:
    content = ff.read()

# Here we have one file cases.csv as in the older volumes
xi_cases_df = pd.read_csv("CI5/CI5-XI/cases.csv", index_col=0)

# load a dataframe from the string
xi_registry_df = pd.read_table(io.StringIO(content), names=["REGISTRY", "NAME"], index_col=0)

# Extract years from index column using Regex
xi_registry_df = xi_registry_df["NAME"].str.extract(r"\s*(.*)\s*((\d+)-(?:\d+,\d+-)?(\d+)\s*\s*
    .rename(columns= {0: "NAME", 1: "PERIOD_1", 2: "PERIOD_2", 3: "PERIOD_3", 4: "PERIOD_4", 5: "PERIOD_5", 6: "PERIOD_6", 7: "PERIOD_7", 8: "PERIOD_8", 9: "PERIOD_9", 10: "PERIOD_10", 11: "PERIOD_11", 12: "PERIOD_12", 13: "PERIOD_13", 14: "PERIOD_14", 15: "PERIOD_15", 16: "PERIOD_16", 17: "PERIOD_17", 18: "PERIOD_18", 19: "PERIOD_19", 20: "PERIOD_20", 21: "PERIOD_21", 22: "PERIOD_22", 23: "PERIOD_23", 24: "PERIOD_24", 25: "PERIOD_25", 26: "PERIOD_26", 27: "PERIOD_27", 28: "PERIOD_28", 29: "PERIOD_29", 30: "PERIOD_30", 31: "PERIOD_31", 32: "PERIOD_32", 33: "PERIOD_33", 34: "PERIOD_34", 35: "PERIOD_35", 36: "PERIOD_36", 37: "PERIOD_37", 38: "PERIOD_38", 39: "PERIOD_39", 40: "PERIOD_40", 41: "PERIOD_41", 42: "PERIOD_42", 43: "PERIOD_43", 44: "PERIOD_44", 45: "PERIOD_45", 46: "PERIOD_46", 47: "PERIOD_47", 48: "PERIOD_48", 49: "PERIOD_49", 50: "PERIOD_50", 51: "PERIOD_51", 52: "PERIOD_52", 53: "PERIOD_53", 54: "PERIOD_54", 55: "PERIOD_55", 56: "PERIOD_56", 57: "PERIOD_57", 58: "PERIOD_58", 59: "PERIOD_59", 60: "PERIOD_60", 61: "PERIOD_61", 62: "PERIOD_62", 63: "PERIOD_63", 64: "PERIOD_64", 65: "PERIOD_65", 66: "PERIOD_66", 67: "PERIOD_67", 68: "PERIOD_68", 69: "PERIOD_69", 70: "PERIOD_70", 71: "PERIOD_71", 72: "PERIOD_72", 73: "PERIOD_73", 74: "PERIOD_74", 75: "PERIOD_75", 76: "PERIOD_76", 77: "PERIOD_77", 78: "PERIOD_78", 79: "PERIOD_79", 80: "PERIOD_80", 81: "PERIOD_81", 82: "PERIOD_82", 83: "PERIOD_83", 84: "PERIOD_84", 85: "PERIOD_85", 86: "PERIOD_86", 87: "PERIOD_87", 88: "PERIOD_88", 89: "PERIOD_89", 90: "PERIOD_90", 91: "PERIOD_91", 92: "PERIOD_92", 93: "PERIOD_93", 94: "PERIOD_94", 95: "PERIOD_95", 96: "PERIOD_96", 97: "PERIOD_97", 98: "PERIOD_98", 99: "PERIOD_99", 100: "PERIOD_100", 101: "PERIOD_101", 102: "PERIOD_102", 103: "PERIOD_103", 104: "PERIOD_104", 105: "PERIOD_105", 106: "PERIOD_106", 107: "PERIOD_107", 108: "PERIOD_108", 109: "PERIOD_109", 110: "PERIOD_110", 111: "PERIOD_111", 112: "PERIOD_112", 113: "PERIOD_113", 114: "PERIOD_114", 115: "PERIOD_115", 116: "PERIOD_116", 117: "PERIOD_117", 118: "PERIOD_118", 119: "PERIOD_119", 120: "PERIOD_120", 121: "PERIOD_121", 122: "PERIOD_122", 123: "PERIOD_123", 124: "PERIOD_124", 125: "PERIOD_125", 126: "PERIOD_126", 127: "PERIOD_127", 128: "PERIOD_128", 129: "PERIOD_129", 130: "PERIOD_130", 131: "PERIOD_131", 132: "PERIOD_132", 133: "PERIOD_133", 134: "PERIOD_134", 135: "PERIOD_135", 136: "PERIOD_136", 137: "PERIOD_137", 138: "PERIOD_138", 139: "PERIOD_139", 140: "PERIOD_140", 141: "PERIOD_141", 142: "PERIOD_142", 143: "PERIOD_143", 144: "PERIOD_144", 145: "PERIOD_145", 146: "PERIOD_146", 147: "PERIOD_147", 148: "PERIOD_148", 149: "PERIOD_149", 150: "PERIOD_150", 151: "PERIOD_151", 152: "PERIOD_152", 153: "PERIOD_153", 154: "PERIOD_154", 155: "PERIOD_155", 156: "PERIOD_156", 157: "PERIOD_157", 158: "PERIOD_158", 159: "PERIOD_159", 160: "PERIOD_160", 161: "PERIOD_161", 162: "PERIOD_162", 163: "PERIOD_163", 164: "PERIOD_164", 165: "PERIOD_165", 166: "PERIOD_166", 167: "PERIOD_167", 168: "PERIOD_168", 169: "PERIOD_169", 170: "PERIOD_170", 171: "PERIOD_171", 172: "PERIOD_172", 173: "PERIOD_173", 174: "PERIOD_174", 175: "PERIOD_175", 176: "PERIOD_176", 177: "PERIOD_177", 178: "PERIOD_178", 179: "PERIOD_179", 180: "PERIOD_180", 181: "PERIOD_181", 182: "PERIOD_182", 183: "PERIOD_183", 184: "PERIOD_184", 185: "PERIOD_185", 186: "PERIOD_186", 187: "PERIOD_187", 188: "PERIOD_188", 189: "PERIOD_189", 190: "PERIOD_190", 191: "PERIOD_191", 192: "PERIOD_192", 193: "PERIOD_193", 194: "PERIOD_194", 195: "PERIOD_195", 196: "PERIOD_196", 197: "PERIOD_197", 198: "PERIOD_198", 199: "PERIOD_199", 200: "PERIOD_200", 201: "PERIOD_201", 202: "PERIOD_202", 203: "PERIOD_203", 204: "PERIOD_204", 205: "PERIOD_205", 206: "PERIOD_206", 207: "PERIOD_207", 208: "PERIOD_208", 209: "PERIOD_209", 210: "PERIOD_210", 211: "PERIOD_211", 212: "PERIOD_212", 213: "PERIOD_213", 214: "PERIOD_214", 215: "PERIOD_215", 216: "PERIOD_216", 217: "PERIOD_217", 218: "PERIOD_218", 219: "PERIOD_219", 220: "PERIOD_220", 221: "PERIOD_221", 222: "PERIOD_222", 223: "PERIOD_223", 224: "PERIOD_224", 225: "PERIOD_225", 226: "PERIOD_226", 227: "PERIOD_227", 228: "PERIOD_228", 229: "PERIOD_229", 230: "PERIOD_230", 231: "PERIOD_231", 232: "PERIOD_232", 233: "PERIOD_233", 234: "PERIOD_234", 235: "PERIOD_235", 236: "PERIOD_236", 237: "PERIOD_237", 238: "PERIOD_238", 239: "PERIOD_239", 240: "PERIOD_240", 241: "PERIOD_241", 242: "PERIOD_242", 243: "PERIOD_243", 244: "PERIOD_244", 245: "PERIOD_245", 246: "PERIOD_246", 247: "PERIOD_247", 248: "PERIOD_248", 249: "PERIOD_249", 250: "PERIOD_250", 251: "PERIOD_251", 252: "PERIOD_252", 253: "PERIOD_253", 254: "PERIOD_254", 255: "PERIOD_255", 256: "PERIOD_256", 257: "PERIOD_257", 258: "PERIOD_258", 259: "PERIOD_259", 260: "PERIOD_260", 261: "PERIOD_261", 262: "PERIOD_262", 263: "PERIOD_263", 264: "PERIOD_264", 265: "PERIOD_265", 266: "PERIOD_266", 267: "PERIOD_267", 268: "PERIOD_268", 269: "PERIOD_269", 270: "PERIOD_270", 271: "PERIOD_271", 272: "PERIOD_272", 273: "PERIOD_273", 274: "PERIOD_274", 275: "PERIOD_275", 276: "PERIOD_276", 277: "PERIOD_277", 278: "PERIOD_278", 279: "PERIOD_279", 280: "PERIOD_280", 281: "PERIOD_281", 282: "PERIOD_282", 283: "PERIOD_283", 284: "PERIOD_284", 285: "PERIOD_285", 286: "PERIOD_286", 287: "PERIOD_287", 288: "PERIOD_288", 289: "PERIOD_289", 290: "PERIOD_290", 291: "PERIOD_291", 292: "PERIOD_292", 293: "PERIOD_293", 294: "PERIOD_294", 295: "PERIOD_295", 296: "PERIOD_296", 297: "PERIOD_297", 298: "PERIOD_298", 299: "PERIOD_299", 300: "PERIOD_300", 301: "PERIOD_301", 302: "PERIOD_302", 303: "PERIOD_303", 304: "PERIOD_304", 305: "PERIOD_305", 306: "PERIOD_306", 307: "PERIOD_307", 308: "PERIOD_308", 309: "PERIOD_309", 310: "PERIOD_310", 311: "PERIOD_311", 312: "PERIOD_312", 313: "PERIOD_313", 314: "PERIOD_314", 315: "PERIOD_315", 316: "PERIOD_316", 317: "PERIOD_317", 318: "PERIOD_318", 319: "PERIOD_319", 320: "PERIOD_320", 321: "PERIOD_321", 322: "PERIOD_322", 323: "PERIOD_323", 324: "PERIOD_324", 325: "PERIOD_325", 326: "PERIOD_326", 327: "PERIOD_327", 328: "PERIOD_328", 329: "PERIOD_329", 330: "PERIOD_330", 331: "PERIOD_331", 332: "PERIOD_332", 333: "PERIOD_333", 334: "PERIOD_334", 335: "PERIOD_335", 336: "PERIOD_336", 337: "PERIOD_337", 338: "PERIOD_338", 339: "PERIOD_339", 340: "PERIOD_340", 341: "PERIOD_341", 342: "PERIOD_342", 343: "PERIOD_343", 344: "PERIOD_344", 345: "PERIOD_345", 346: "PERIOD_346", 347: "PERIOD_347", 348: "PERIOD_348", 349: "PERIOD_349", 350: "PERIOD_350", 351: "PERIOD_351", 352: "PERIOD_352", 353: "PERIOD_353", 354: "PERIOD_354", 355: "PERIOD_355", 356: "PERIOD_356", 357: "PERIOD_357", 358: "PERIOD_358", 359: "PERIOD_359", 360: "PERIOD_360", 361: "PERIOD_361", 362: "PERIOD_362", 363: "PERIOD_363", 364: "PERIOD_364", 365: "PERIOD_365", 366: "PERIOD_366", 367: "PERIOD_367", 368: "PERIOD_368", 369: "PERIOD_369", 370: "PERIOD_370", 371: "PERIOD_371", 372: "PERIOD_372", 373: "PERIOD_373", 374: "PERIOD_374", 375: "PERIOD_375", 376: "PERIOD_376", 377: "PERIOD_377", 378: "PERIOD_378", 379: "PERIOD_379", 380: "PERIOD_380", 381: "PERIOD_381", 382: "PERIOD_382", 383: "PERIOD_383", 384: "PERIOD_384", 385: "PERIOD_385", 386: "PERIOD_386", 387: "PERIOD_387", 388: "PERIOD_388", 389: "PERIOD_389", 390: "PERIOD_390", 391: "PERIOD_391", 392: "PERIOD_392", 393: "PERIOD_393", 394: "PERIOD_394", 395: "PERIOD_395", 396: "PERIOD_396", 397: "PERIOD_397", 398: "PERIOD_398", 399: "PERIOD_399", 400: "PERIOD_400", 401: "PERIOD_401", 402: "PERIOD_402", 403: "PERIOD_403", 404: "PERIOD_404", 405: "PERIOD_405", 406: "PERIOD_406", 407: "PERIOD_407", 408: "PERIOD_408", 409: "PERIOD_409", 410: "PERIOD_410", 411: "PERIOD_411", 412: "PERIOD_412", 413: "PERIOD_413", 414: "PERIOD_414", 415: "PERIOD_415", 416: "PERIOD_416", 417: "PERIOD_417", 418: "PERIOD_418", 419: "PERIOD_419", 420: "PERIOD_420", 421: "PERIOD_421", 422: "PERIOD_422", 423: "PERIOD_423", 424: "PERIOD_424", 425: "PERIOD_425", 426: "PERIOD_426", 427: "PERIOD_427", 428: "PERIOD_428", 429: "PERIOD_429", 430: "PERIOD_430", 431: "PERIOD_431", 432: "PERIOD_432", 433: "PERIOD_433", 434: "PERIOD_434", 435: "PERIOD_435", 436: "PERIOD_436", 437: "PERIOD_437", 438: "PERIOD_438", 439: "PERIOD_439", 440: "PERIOD_440", 441: "PERIOD_441", 442: "PERIOD_442", 443: "PERIOD_443", 444: "PERIOD_444", 445: "PERIOD_445", 446: "PERIOD_446", 447: "PERIOD_447", 448: "PERIOD_448", 449: "PERIOD_449", 450: "PERIOD_450", 451: "PERIOD_451", 452: "PERIOD_452", 453: "PERIOD_453", 454: "PERIOD_454", 455: "PERIOD_455", 456: "PERIOD_456", 457: "PERIOD_457", 458: "PERIOD_458", 459: "PERIOD_459", 460: "PERIOD_460", 461: "PERIOD_461", 462: "PERIOD_462", 463: "PERIOD_463", 464: "PERIOD_464", 465: "PERIOD_465", 466: "PERIOD_466", 467: "PERIOD_467", 468: "PERIOD_468", 469: "PERIOD_469", 470: "PERIOD_470", 471: "PERIOD_471", 472: "PERIOD_472", 473: "PERIOD_473", 474: "PERIOD_474", 475: "PERIOD_475", 476: "PERIOD_476", 477: "PERIOD_477", 478: "PERIOD_478", 479: "PERIOD_479", 480: "PERIOD_480", 481: "PERIOD_481", 482: "PERIOD_482", 483: "PERIOD_483", 484: "PERIOD_484", 485: "PERIOD_485", 486: "PERIOD_486", 487: "PERIOD_487", 488: "PERIOD_488", 489: "PERIOD_489", 490: "PERIOD_490", 491: "PERIOD_491", 492: "PERIOD_492", 493: "PERIOD_493", 494: "PERIOD_494", 495: "PERIOD_495", 496: "PERIOD_496", 497: "PERIOD_497", 498: "PERIOD_498", 499: "PERIOD_499", 500: "PERIOD_500", 501: "PERIOD_501", 502: "PERIOD_502", 503: "PERIOD_503", 504: "PERIOD_504", 505: "PERIOD_505", 506: "PERIOD_506", 507: "PERIOD_507", 508: "PERIOD_508", 509: "PERIOD_509", 510: "PERIOD_510", 511: "PERIOD_511", 512: "PERIOD_512", 513: "PERIOD_513", 514: "PERIOD_514", 515: "PERIOD_515", 516: "PERIOD_516", 517: "PERIOD_517", 518: "PERIOD_518", 519: "PERIOD_519", 520: "PERIOD_520", 521: "PERIOD_521", 522: "PERIOD_522", 523: "PERIOD_523", 524: "PERIOD_524", 525: "PERIOD_525", 526: "PERIOD_526", 527: "PERIOD_527", 528: "PERIOD_528", 529: "PERIOD_529", 530: "PERIOD_530", 531: "PERIOD_531", 532: "PERIOD_532", 533: "PERIOD_533", 534: "PERIOD_534", 535: "PERIOD_535", 536: "PERIOD_536", 537: "PERIOD_537", 538: "PERIOD_538", 539: "PERIOD_539", 540: "PERIOD_540", 541: "PERIOD_541", 542: "PERIOD_542", 543: "PERIOD_543", 544: "PERIOD_544", 545: "PERIOD_545", 546: "PERIOD_546", 547: "PERIOD_547", 548: "PERIOD_548", 549: "PERIOD_549", 550: "PERIOD_550", 551: "PERIOD_551", 552: "PERIOD_552", 553: "PERIOD_553", 554: "PERIOD_554", 555: "PERIOD_555", 556: "PERIOD_556", 557: "PERIOD_557", 558: "PERIOD_558", 559: "PERIOD_559", 560: "PERIOD_560", 561: "PERIOD_561", 562: "PERIOD_562", 563: "PERIOD_563", 564: "PERIOD_564", 565: "PERIOD_565", 566: "PERIOD_566", 567: "PERIOD_567", 568: "PERIOD_568", 569: "PERIOD_569", 570: "PERIOD_570", 571: "PERIOD_571", 572: "PERIOD_572", 573: "PERIOD_573", 574: "PERIOD_574", 575: "PERIOD_575", 576: "PERIOD_576", 577: "PERIOD_577", 578: "PERIOD_578", 579: "PERIOD_579", 580: "PERIOD_580", 581: "PERIOD_581", 582: "PERIOD_582", 583: "PERIOD_583", 584: "PERIOD_584", 585: "PERIOD_585", 586: "PERIOD_586", 587: "PERIOD_587", 588: "PERIOD_588", 589: "PERIOD_589", 590: "PERIOD_590", 591: "PERIOD_591", 592: "PERIOD_592", 593: "PERIOD_593", 594: "PERIOD_594", 595: "PERIOD_595", 596: "PERIOD_596", 597: "PERIOD_597", 598: "PERIOD_598", 599: "PERIOD_599", 600: "PERIOD_600", 601: "PERIOD_601", 602: "PERIOD_602", 603: "PERIOD_603", 604: "PERIOD_604", 605: "PERIOD_605", 606: "PERIOD_606", 607: "PERIOD_607", 608: "PERIOD_608", 609: "PERIOD_609", 610: "PERIOD_610", 611: "PERIOD_611", 612: "PERIOD_612", 613: "PERIOD_613", 614: "PERIOD_614", 615: "PERIOD_615", 616: "PERIOD_616", 617: "PERIOD_617", 618: "PERIOD_618", 619: "PERIOD_619", 620: "PERIOD_620", 621: "PERIOD_621", 622: "PERIOD_622", 623: "PERIOD_623", 624: "PERIOD_624", 625: "PERIOD_625", 626: "PERIOD_626", 627: "PERIOD_627", 628: "PERIOD_628", 629: "PERIOD_629", 630: "PERIOD_630", 631: "PERIOD_631", 632: "PERIOD_632", 633: "PERIOD_633", 634: "PERIOD_634", 635: "PERIOD_635", 636: "PERIOD_636", 637: "PERIOD_637", 638: "PERIOD_638", 639: "PERIOD_639", 640: "PERIOD_640", 641: "PERIOD_641", 642: "PERIOD_642", 643: "PERIOD_643", 644: "PERIOD_644", 645: "PERIOD_645", 646: "PERIOD_646", 647: "PERIOD_647", 648: "PERIOD_648", 649: "PERIOD_649", 650: "PERIOD_650", 651: "PERIOD_651", 652: "PERIOD_652", 653: "PERIOD_653", 654: "PERIOD_654", 655: "PERIOD_655", 656: "PERIOD_656", 657: "PERIOD_657", 658: "PERIOD_658", 659: "PERIOD_659", 660: "PERIOD_660", 661: "PERIOD_661", 662: "PERIOD_662", 663: "PERIOD_663", 664: "PERIOD_664", 665: "PERIOD_665", 666: "PERIOD_666", 667: "PERIOD_667", 668: "PERIOD_668", 669: "PERIOD_669", 670: "PERIOD_670", 671: "PERIOD_671", 672: "PERIOD_672", 673: "PERIOD_673", 674: "PERIOD_674", 675: "PERIOD_675", 676: "PERIOD_676", 677: "PERIOD_677", 678: "PERIOD_678", 679: "PERIOD_679", 680: "PERIOD_680", 681: "PERIOD_681", 682: "PERIOD_682", 683: "PERIOD_683", 684: "PERIOD_684", 685: "PERIOD_685", 686: "PERIOD_686", 687: "PERIOD_687", 688: "PERIOD_688", 689: "PERIOD_689", 690: "PERIOD_690", 691: "PERIOD_691", 692: "PERIOD_692", 693: "PERIOD_693", 694: "PERIOD_694", 695: "PERIOD_695", 696: "PERIOD_696", 697: "PERIOD_697", 698: "PERIOD_698", 699: "PERIOD_699", 700: "PERIOD_700", 701: "PERIOD_701", 702: "PERIOD_702", 703: "PERIOD_703", 704: "PERIOD_704", 705: "PERIOD_705", 706: "PERIOD_706", 707: "PERIOD_707", 708: "PERIOD_708", 709: "PERIOD_709", 710: "PERIOD_710", 711: "PERIOD_711", 712: "PERIOD_712", 713: "PERIOD_713", 714: "PERIOD_714", 715: "PERIOD_715", 716: "PERIOD_716", 717: "PERIOD_717", 718: "PERIOD_718", 719: "PERIOD_719", 720: "PERIOD_720", 721: "PERIOD_721", 722: "PERIOD_722", 723: "PERIOD_723", 724: "PERIOD_724", 725: "PERIOD_725", 726: "PERIOD_726", 727: "PERIOD_727", 728: "PERIOD_728", 729: "PERIOD_729", 730: "PERIOD_730", 731: "PERIOD_731", 732: "PERIOD_732", 733: "PERIOD_733", 734: "PERIOD_734", 735: "PERIOD_735", 736: "PERIOD_736", 737: "PERIOD_737", 738: "PERIOD_738", 739: "PERIOD_739", 740: "PERIOD_740", 741: "PERIOD_741", 742: "PERIOD_742", 743: "PERIOD_743", 744: "PERIOD_744", 745: "PERIOD_745", 746: "PERIOD_746", 747: "PERIOD_747", 748: "PERIOD_748", 749: "PERIOD_749", 750: "PERIOD_750", 751: "PERIOD_751", 752: "PERIOD_752", 753: "PERIOD_753", 754: "PERIOD_754", 755: "PERIOD_755", 756: "PERIOD_756", 757: "PERIOD_757", 758: "PERIOD_758", 759: "PERIOD_759", 760: "PERIOD_760", 761: "PERIOD_761", 762: "PERIOD_762", 763: "PERIOD_763", 764: "PERIOD_764", 765: "PERIOD_765", 766: "PERIOD_766", 767: "PERIOD_767", 768: "PERIOD_768", 769: "PERIOD_769", 770: "PERIOD_770", 771: "PERIOD_771", 772: "PERIOD_772", 773: "PERIOD_773", 774: "PERIOD_774", 775: "PERIOD_775", 776: "PERIOD_776", 777: "PERIOD_777", 778: "PERIOD_778", 779: "PERIOD_779", 780: "PERIOD_780", 781: "PERIOD_781", 782: "PERIOD_782", 783: "PERIOD_783", 784: "PERIOD_784", 785: "PERIOD_785", 786: "PERIOD_786", 787: "PERIOD_787", 788: "PERIOD_788", 789: "PERIOD_789", 790: "PERIOD_790", 791: "PERIOD_791", 792: "PERIOD_792", 793: "PERIOD_793", 794: "PERIOD_794", 795: "PERIOD_795", 796: "PERIOD_796", 797: "PERIOD_797", 798: "PERIOD_798", 799: "PERIOD_799", 800: "PERIOD_800", 801: "PERIOD_801", 802: "PERIOD_802", 803: "PERIOD_803", 804: "PERIOD_804", 805: "PERIOD_805", 806: "PERIOD_806", 807: "PERIOD_807", 808: "PERIOD_808", 809: "PERIOD_809", 810: "PERIOD_810", 811: "PERIOD_811", 812: "PERIOD_812", 813: "PERIOD_813", 814: "PERIOD_814", 815: "PERIOD_815", 816: "PERIOD_816", 817: "PERIOD_817", 818: "PERIOD_818", 819: "PERIOD_819", 820: "PERIOD_820", 821: "PERIOD_821", 822: "PERIOD_822", 823: "PERIOD_823", 824: "PERIOD_824", 825: "PERIOD_825", 826: "PERIOD_826", 827: "PERIOD_827", 828: "PERIOD_828", 829: "PERIOD_829", 830: "PERIOD_830", 831: "PERIOD_831", 832: "PERIOD_832", 833: "PERIOD_833", 834: "PERIOD_834", 835: "PERIOD_835", 836: "PERIOD_836", 837: "PERIOD_837", 838: "PERIOD_838", 839: "PERIOD_839", 840: "PERIOD_840", 841: "PERIOD_841", 842: "PERIOD_842", 843: "PERIOD_843", 844: "PERIOD_844", 845: "PERIOD_845", 846: "PERIOD_846", 847: "PERIOD_847", 848: "PERIOD_848", 849: "PERIOD_849", 850: "PERIOD_850", 851: "PERIOD_851", 852: "PERIOD_852", 853: "PERIOD_853", 854: "PERIOD_854", 855: "PERIOD_855", 856: "PERIOD_856", 857: "PERIOD_857", 858: "PERIOD_858", 859: "PERIOD_859", 860: "PERIOD_860", 861: "PERIOD_861", 862: "PERIOD_862", 863: "PERIOD_863", 864: "PERIOD_864", 865: "PERIOD_865", 866: "PERIOD_866", 867: "PERIOD_867", 868: "PERIOD_868", 869: "PERIOD_869", 870: "PERIOD_870", 871: "PERIOD_871", 872: "PERIOD_872", 873: "PERIOD_873", 874: "PERIOD_874", 875: "PERIOD_875", 876: "PERIOD_876", 877: "PERIOD_877", 878: "PERIOD_878", 879: "PERIOD_879", 880: "PERIOD_880", 881: "PERIOD_881", 882: "PERIOD_882", 883: "PERIOD_883", 884: "PERIOD_884", 885: "PERIOD_885", 886: "PERIOD_886", 887: "PERIOD_887", 888: "PERIOD_888", 889: "PERIOD_889", 890: "PERIOD_890", 891: "PERIOD_891", 892: "PERIOD_892", 893: "PERIOD_893", 894: "PERIOD_894", 895: "PERIOD_895", 896: "PERIOD_896", 897: "PERIOD_897", 898: "PERIOD_898", 899: "PERIOD_899", 900: "PERIOD_900", 901: "PERIOD_901", 902: "PERIOD_902", 903: "PERIOD_903", 904: "PERIOD_904", 905: "PERIOD_905", 906: "PERIOD_906", 907: "PERIOD_907", 908: "PERIOD_908", 909: "PERIOD_909", 910: "PERIOD_910", 911: "PERIOD_911", 912: "PERIOD_912", 913: "PERIOD_913", 914: "PERIOD_914", 915: "PERIOD_915", 916: "PERIOD_916",
```

```

# join two fields with start and end years into one field PERIOD in a format "1995-1998"
xi_df["PERIOD"] = xi_df["PERIOD_1"].astype(str) + '-' + xi_df["PERIOD_2"].astype(str)

xi_df = xi_df.groupby(["PERIOD"]).sum()

# remove columns with irrelevant information
xi_df.drop(columns=["SEX", "CANCER", "TOTAL"], inplace=True)

# rename column names to a standard form
xi_df.rename(columns={"N85": "N85+", "N_unk": "N_UNK"}, inplace=True)

xi_df # uncomment to display data for this volume

```

Save results

Merge data from all volumes and save to one intermediate CSV file that serve as a source for the further stages. This file is relatively small and can be cached, so there is no need to run Preprocessing stage more than once.

```

df = pd.concat([v_df, vi_df, viid_df, viiid_df, ixd_df, xd_df, xi_df])
df.to_csv("dataset.csv")
df

```

Processing

At this stage I no longer touch the raw CI5 files. The only source of the information is a dataset we generated at the previous stage: *dataset.csv*. Each row in this dataset is a time period (from_year-till_year). Each column of this dataset is a patient's age group (N0_4, N5_9, etc.) This format reflects the original form of the data. In the raw CI5 files we did not have a breakdown by year, but there were periods of several years. Some periods overlap each other.

This data format is difficult to visualize and analyze because we don't have information for every specific year. The goal of this stage is to recast the existing dataset to a form suitable for analysis at the next stage. For this purpose, I decided to apply a very simple interpolation. If we have the number of cases over a period of several years, then we simply divide this number by the number of years and take into account the quotient for each year in the period: $Y_i = \frac{C}{i_{max} - i_{min}}$ where C is a total number of cases in a period, Y_i is the number of cases for a specific year in a period.

```

# Check if the dataset is ready
assert os.path.exists("dataset.csv"), "dataset.csv does not exist, please run Preprocessing

df = pd.read_csv("dataset.csv", index_col=0)

# Get a number of years covered by each period, it is inclusive, i.e. 2001-2003 = 3 years

```

```

years = df.index.str.extract("(\\d+)-(\\d+)", expand=True)
years_count = (years[1].astype(int) - years[0].astype(int) + 1)
years_count.set_axis(df.index, inplace=True)

# divide a number of cancer cases in each period by a number of years covered by each period
df = df.floordiv(years_count, axis=0)
df.reset_index(inplace=True)

# Convert periods in a form YYYY-YYYY to a list of years YYYY, add it to a new column YEAR
df.insert(1, "YEAR", df["PERIOD"].str.extract("(\\d+)-(\\d+)", expand=True) \
    .apply(lambda x : list(range(int(x[0]), int(x[1])+1)), axis=1))
df = df.explode("YEAR")
df.reset_index(inplace=True)

# remove irrelevant columns
df.drop(columns=["index", "PERIOD"], inplace=True)

df = df.groupby("YEAR").sum()

```

Data validation

Let's check the resulting dataset:

```

df.head()

df.tail()

```

It seems pretty legit. Except that the records of the first years have unrealistically small values. This data comes from the first volume (V). I've checked the original data and came to a conclusion that this is a feature of the dataset and not a processing error. However, due to the low value of this data, I decided to delete it from the dataset:

```

df = df.loc[1978:]

```

Let's check the boundary values for years:

```

(df.index.min(), df.index.max())

```

So this resulting dataset covers a period of 34 years from 1978 till 2012. Let's check the boundary values for number of cases:

```

pd.concat([df.max().rename("MAX"), df.min().rename("MIN")], axis=1)

```

Fortunately, no extreme big or negative values in the dataset. All looks pretty legit. Let's check the distribution of cancer cases by age groups:

```

# disable scientific notation for Y values
plt.ticklabel_format(axis="y", style="plain")
plt.title("Figure 2. Number of cancer cases by age groups", fontdict={"fontsize": 16})
plt.ylabel("Number of cases", fontsize=16)

```

```
plt.xlabel("Age group", fontsize=16)
```

```
df.sum().plot(kind="bar", figsize=(15,7), fontsize=16)
```

Here we see a picture very similar to Figure 1. Invasive cancer incidence, by age, U.S., 2009 [2]. Cancer incidence rises rapidly with age. Since it has the same property, it also confirms the veracity of this dataset.

Analysis

Let's visualize dynamics of the total number of registered cancer cases in the world:

```
total_cases = df.sum(axis=1)
```

```
# disable scientific notation for Y values
```

```
plt.ticklabel_format(axis="y", style="plain")
```

```
plt.title("Figure 3. Number of cancer cases reported in CI5 dataset", fontdict={"fontsize": 16})
```

```
plt.ylabel("Number of cases", fontsize=16)
```

```
plt.xlabel("Year", fontsize=16)
```

```
total_cases.plot.bar(figsize=(15,7), fontsize=16)
```

As we can see from this graph, the number of reported cancer cases increases in steps over time. Remember, our dataset consists of individual reports (volumes) that were published at different times. Different reports had different coverage of regions. These steps show the boundaries of different volumes. As we can see, over time the coverage has improved and this was reflected in more reported cases on the right side of the graph. Unfortunately, due to this feature, we cannot rely on absolute values in our dataset.

Let's visualize dynamics of registered cancer cases in the world in young adults in the recent decades:

```
young_adults_cases = df["N15_19"] + df["N20_24"] + df["N25_29"] + df["N30_34"] + df["N35_39"]
```

```
# disable scientific notation for Y values
```

```
plt.ticklabel_format(axis="y", style="plain")
```

```
plt.title("Figure 4. Number of cancer cases reported in CI5 dataset for 15-44 years old patients", fontdict={"fontsize": 16})
```

```
plt.ylabel("Number of cases", fontsize=16)
```

```
plt.xlabel("Year", fontsize=16)
```

```
young_adults_cases.plot.bar(stacked=True, figsize=(15,7), fontsize=16)
```

Here we see almost the same figure as previous one. From this we can conclude that more cases of cancer have been reported among younger patients in recent decades. However, this feature of the dataset may not reflect the actual dynamics. To better understand this data, you need to look at relative numbers.

Let's visualize dynamics of the percentage of young people among all cases in

the recent decades.

```
young_adults_percentage = (young_adults_cases / total_cases * 100)
```

```
# disable scientific notation for Y values
```

```
plt.ticklabel_format(axis="y", style="plain")
```

```
plt.title("Figure 5. Percentage of 15-44 years old patients among all reported cancer cases")
```

```
plt.ylabel("%", fontsize=18)
```

```
plt.xlabel("Year", fontsize=16)
```

```
# draw a mean line over the graph
```

```
mean = young_adults_percentage.mean()
```

```
plt.plot(young_adults_percentage.index, [mean]*len(young_adults_cases.index), color='grey',
```

```
plt.text(young_adults_percentage.index[-5], mean+0.05, f'Mean = {round(mean, 2)}%', color='g'
```

```
# annotate the maximum
```

```
plt.annotate(f'A max of {round(young_adults_percentage.max(),2)}% in {young_adults_percentag
```

```
xy=(young_adults_percentage.idxmax(), young_adults_percentage.max()),
```

```
xycoords='data',
```

```
xytext=(0.2, 0.95),
```

```
textcoords='axes fraction',
```

```
arrowprops=dict(facecolor='black', shrink=0.05),
```

```
horizontalalignment='right',
```

```
verticalalignment='top')
```

```
# annotate the minimum
```

```
plt.annotate(f'A min of {round(young_adults_percentage.min(),2)}% in {young_adults_percentag
```

```
xy=(young_adults_percentage.idxmin(), young_adults_percentage.min()),
```

```
xycoords='data',
```

```
xytext=(0.7, 0.1),
```

```
textcoords='axes fraction',
```

```
arrowprops=dict(facecolor='black', shrink=0.05),
```

```
horizontalalignment='right',
```

```
verticalalignment='top')
```

```
young_adults_percentage.plot(figsize=(15,7), fontsize=16, color='blue')
```

The graph shows that the percentage of young patients among all recorded cancer cases is around 9% and not very volatile over time. There is no trend for a significant increase in the proportion of young patients in recent years.

Discussion and conclusion

In the reviewed literature, there is evidence of an increase in the incidence of cancer among young patients in some regions, in particular in the USA. The aim of this project was to analyze the data for the entire world to understand whether such a trend is taking place worldwide, or if it is a local phenomenon.

As a result of the analysis, I found that over the past decades, the number of reported cancer cases has increased significantly, from around 5,000,000 cases in 1978 to around 40,000,000 cases in 2012. It's x8 growth. While the world population increased from 4 billion to 7 billion for the same period. Such a disproportional increase in registered cases, as well as the stepwise nature of this growth, indicates that it is not a natural increase in the incidence. More likely that this shows a gradual increase in the territory's coverage for CI5 dataset that serve as a source for the analysis.

Inconsistency in area coverage limits the methods of analysis that can be applied to our dataset. In particular, it is not possible to calculate the incidence per 100,000 persons at risk. This metric is the gold standard in measuring incidence and assessing epidemiological trends.

However, we can assume that the data is not biased against the covered age groups. This assumption is supported by the fact that Figure 2. "Number of cancer cases by age groups" that I derive from the considered dataset has the same characteristics as in other studies (for example, Figure 1). Therefore, I decided to evaluate the ratio between young patients and all patients over time.

The Figure 5 shows that the percentage of young cancer patients little changed over time. It hits the maximum value of 10.02% in 1991 then declined. It reached a minimum of 7.8% in 2008. As can be seen from the graph, small fluctuations were in both directions within the ambit of 2%. It leads to the conclusion that on a global scale there is no trend towards an increase in the proportion of young people among all cases of cancer. This result may mean that the causes for the increase in cancer cases in the considered regions are of a local nature and are not the result of global changes, such as climate change.

References

- [1] "What Is Cancer?" by National Cancer Institute (2021, May 5) [Online]. Available: <https://www.cancer.gov/about-cancer/understanding/what-is-cancer>
- [2] "Age and Cancer Risk" Am J Prev Med. 2014 Mar; 46(3 0 1): S7–15. [Online]. Available: <https://doi.org/10.1016/j.amepre.2013.10.029>
- [3] "Childhood Cancers" by National Cancer Institute (2021, April 12) [Online]. Available: <https://www.cancer.gov/types/childhood-cancers>
- [4] "The Challenging Landscape of Cancer and Aging: Charting a Way Forward" by Norman E. Sharpless, M.D. (2018, January 24) [Online]. Available: <https://www.cancer.gov/news-events/cancer-currents-blog/2018/sharpless-aging-cancer-research>
- [5] di Martino, E., Smith, L., Bradley, S.H. et al. Incidence trends for twelve cancers in younger adults—a rapid review. Br J Cancer 126, 1374–1386 (2022). [Online]. Available: <https://doi.org/10.1038/s41416-022-01704-x>

- [6] Scott AR, Stoltzfus KC, Tchelebi LT, et al. Trends in Cancer Incidence in US Adolescents and Young Adults, 1973-2015. *JAMA Netw Open.* 2020;3(12):e2027738. [Online]. Available: <https://doi.org/10.1001/jamanetworkopen.2020.27738>
- [7] Cancer Stat Facts: Cancer Among Adolescents and Young Adults (AYAs) (Ages 15–39) by National Cancer Institute (2022) [Online]. Available: <https://seer.cancer.gov/statfacts/html/aya.html>
- [8] CI5: CANCER INCIDENCE IN FIVE CONTINENTS by International Agency for Research on Cancer (IARC) [Online]. Available: <https://ci5.iarc.fr/Default.aspx>
- [9] Muir, C.S., Waterhouse, J., Mack, T., Powell, J., Whelan, S.L., eds (1987) *Cancer Incidence in Five Continents, Vol. V* IARC Scientific Publications, No. 88, Lyon, IARC. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [10] Parkin, D.M., Muir, C.S., Whelan, S.L., Gao, Y.-T., Ferlay, J., Powell, J., eds (1992) *Cancer Incidence in Five Continents, Vol. VI* IARC Scientific Publications, No. 120, Lyon, IARC. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [11] Parkin, D.M., Whelan, S.L., Ferlay, J., Raymond, L., and Young, J., eds (1997) *Cancer Incidence in Five Continents, Vol. VII* IARC Scientific Publications, No. 143, Lyon, IARC. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [12] Parkin, D.M., Whelan, S.L., Ferlay, J., Teppo, L., and Thomas, D.B., eds (2002) *Cancer Incidence in Five Continents, Vol. VIII* IARC Scientific Publications, No. 155, Lyon, IARC. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [13] Curado. M. P., Edwards. B., Shin. H.R., Storm. H., Ferlay. J., Heanue. M. and Boyle. P., eds (2007) *Cancer Incidence in Five Continents, Vol. IX* IARC Scientific Publications, No. 160, Lyon, IARC. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [14] Forman D, Bray F, Brewster DH, Gombe Mbalawa C, Kohler B, Piñeros M, Steliarova-Foucher E, Swaminathan R and Ferlay J, editors (2014) *Cancer Incidence in Five Continents, Vol. X* IARC Scientific Publication No. 164. Lyon: International Agency for Research on Cancer. [Online]. Available: <https://ci5.iarc.fr/CI5I-X/Pages/download.aspx>
- [15] Bray F, Colombet M, Mery L, Piñeros M, Znaor A, Zanetti R and Ferlay J, editors (2017) *Cancer Incidence in Five Continents, Vol. XI* (electronic version). Lyon: International Agency for Research on Cancer. Available from: <https://ci5.iarc.fr>, accessed 25.06.2022. The printed version of this volume (IARC Scientific Publication No. 166) is also available for download in PDF format at the IARC Publications website (<https://publications.iarc.fr/597>).