Name Chukwunonso Otekeiwebia

LISUM30

01/03/2024

Analysing the Crude death Rates and Standardised death rates

```
into and excel sheet and convert to CSV.
            We can see that that data is per 100,000 meaning that we operate as if there were 100,000 individuals in the population, Hence,
            Population = 100,000 times the number of age groups
            To get the crude death rates, we summed up the death rates for both countries and divided by their population (per 100,000)
            respectively. rounding off their rates to 3 decimal places
           Futher research into an Age standardised death rates shows that for each age group in the standard population distribution, we need to know the proportion of the population that falls within that age group
            With this additional information, we can calculate the age-standardized death rate using a weighted average approach.
           We would multiply the death rate in each age group by the corresponding weight in the standard population distribution, sum these values across all age groups, and then divide by the total population of the standard population. This process ensures that the calculated rate is adjusted for differences in the age distribution between populations and allows for meaningful comparisons between populations with different age structures.
           these disparities in rates can be due to disparities in disease awareness and management, Health care Infrastructures and more.
In [1]: import pandas as pd
            import numpy as np
In [3]: df = pd.read_csv(r"C:\Users\Tekno\Downloads\book i.csv")
In [35]: df.head(2)
Out[35]:
                  Age group (years) Death rate, United States, 2019 Death rate, Uganda, 2019 Per(100,000)
                                                                            0.40
                                            0.04
              0 0-4
                                                                                                           100000
              1
                                 5-9
                                                                   0.02
                                                                                               0.17
                                                                                                            100000
 In [6]: df['Age group (years)'] = df['Age group (years)'].str.replace(',', '-')
In [41]: # Derive the total death rates
total_deaths_us = sum(df['Death rate, United States, 2019'])
total_deaths_ug = sum(df['Death rate, Uganda, 2019'])
In [17]: total deaths us
Out[17]: 2164.74
In [16]: total_deaths_ug
Out[16]: 2058.62
In [18]: population us = 100000
              population_ug = 100000
In [44]: # define the crude death rates and round off to 2 decimal place
             crude_death_rate_us = total_deaths_us / population_us
crude_death_rate_ug = total_deaths_ug / population_ug
crude_death_rate_ug = round(0.021647399999999997,2)
crude_death_rate_ug = round(0.0205862,2)
In [45]: print(f'The crude death rates are {crude_death_rate_US} for the United States and {crude_death_rate_ug} for Uganda')
              The crude death rates are 0.02 for the United States and 0.02 for Uganda
In [36]: # Calculate the weighted average of death rates for each country
weighted_avg_us = (df['Death rate, United States, 2019'] * df['Per(100,000)']).sum() / df['Per(100,000)'].sum()
weighted_avg_uganda = (df['Death rate, Uganda, 2019'] * df['Per(100,000)']).sum() / df['Per(100,000)'].sum()
             print("Age-Standardized Death Rate (US):", round(weighted_avg_us, 1))
print("Age-Standardized Death Rate (Uganda):", round(weighted_avg_uganda, 1))
             Age-Standardized Death Rate (US): 120.3
Age-Standardized Death Rate (Uganda): 114.4
```

First we researched information about age death rates and age standardised death rates, their meanings and differences we place

```
Flask.py > ...
    from flask import Flask
    app = Flask(__name__)
    @app.route('/')
    def hello_flask():
        return 'Hello welcome to my Flask'
    app.run(host='0.0.0.0',port=80)
```

Hello welcome to my Flask

```
OPEN EDITORS
                                                                            1 from flask import Flask, render_template
                                                                                     import os
      o index.html Images
                                                                                     app = Flask(_name_)
picFolder = os.path.join('static','pics')
app.config['UPLOAD_FOLDER'] = picFolder
       ≡ Extension: Live Server
WEEK-4
                                                                                     @app.route('/')
                                                                                     def index():
    image_1 = os.path.join(app.config['UPLOAD_FOLDER'],1.jpg)
    return render_template('index.html',user_image = 1)
🖾 1.png
2.jpg
🖾 flask 1.png
                                                                                     @app.route('/index')
🖾 flaskk.jpg
                                                                                     def hello_flask():
    return 'Hello welcome to my Flask'
                                                                         PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
                                                                                                                                                                                                                                        ∑ Code + ∨ □
                                                                              Use a production WSGI server instead.
                                                                         Use a production WSGI server instead.

* Debug mode: off

* Running on all addresses.

WARNING: This is a development server. Do not use it in a production deployment.

* Running on http://192.168.1.205:80/ (Press CTRL+C to quit)

192.168.1.205 - [01/Mar/2024 12:52:49] "GET / HTTP/1.1" 200 -

192.168.1.205 - [01/Mar/2024 12:52:49] "GET / Favicon.ico HTTP/1.1" 404 -

192.168.1.205 - [01/Mar/2024 13:23:40] "GET / HTTP/1.1" 200 -
```