R Take-Home Assignment 1: Gapminder (Wrangling & Visualization)

Your Name

2025-09-30

Overview

In this first assignment, you will work with the Gapminder dataset, a classic resource for studying global development. Originally compiled by the Gapminder Foundation, this dataset brings together health, population, and economic data for 142 countries between 1952 and 2007, measured at five-year intervals.

Each record in the dataset contains:

- Life expectancy (`lifeExp`): the average number of years a newborn is expected to live, given current mortality rates.
- Population size (`pop`): the total number of inhabitants in the country that year.
- GDP per capita (`gdpPercap`): the average economic output per person, adjusted for inflation and expressed in US dollars.
- `Country` and `continent` information: identifiers for grouping and comparison.

The dataset is widely used because it allows us to explore broad questions: How did life expectancy change across different continents over the second half of the 20th century? Do wealthier countries consistently have higher life expectancy? And how much variation exists within regions?

In this assignment, you will practice the core steps of data analysis in R: importing data, exploring its structure, creating subsets, calculating summaries, and producing visualisations. These are fundamental skills you will use again and again in your own research.

Source

`https://github.com/resbaz/r-novice-gapminder-files/raw/master/data/gapminder-FiveYearData.csv`

Skills

`loading CSVs, subsetting, summarising, plotting`

Preparation

You will need the following R packages:

- `dplyr`, `ggplot2`, `readr`

```
# Task: load the required packages (hint: use library())
# library(dplyr)
# library(ggplot2)
# library(readr)
```

1. Load the data

Every analysis begins with bringing the data into R. Here, you will import the Gapminder dataset directly from its online source.

Task: Load the Gapminder dataset into R and store it as `gapminder`.

```
# Task: load dataset here (hint: read_csv() from readr)
```

2. Explore the dataset

Before we can analyse, we need to understand what the dataset looks like. How many countries are covered, what years are available, and what variables can we work with?

Task: Report the number of countries, the range of years, and the variable names.

```
# Task: explore the dataset (hint: n_distinct(), range(), names())
```

Write your answers here:

```
Countries: ...
Years: ...
Variables: ...
```

3. Subset the data

Analyses are often limited to a particular region of interest. Suppose you are asked to focus only on Europe.

Task: Create a subset containing only the rows where `continent` is `"Europe"`. Save this as `europe`.

```
# Task: subset the data (hint: filter(continent == "Europe"))
```

4. Summarise life expectancy

To see how regions compare, we want to calculate average life expectancy for each continent over time. This will let us track differences across the decades.

Task: Calculate the mean life expectancy per continent per year. Save the result as `lifeexp_summary`.

```
# Task: summarise data (hint: group_by() + summarise())
```

5. Visualise trends

Summaries are useful, but a plot makes patterns easier to see. We will now compare how life expectancy has changed in different regions.

Task: Produce a line plot of life expectancy over time for **two continents of your choice**. Label the axes and add a title.

```
# Task: plot trends (hint: ggplot() + geom_line())
```

6. Reflection

Finally, consider what your analysis shows. Think about whether continents improved at the same pace, and whether any regions lag behind.

Task: Write 3–4 sentences describing the patterns you see across continents in life expectancy trends.

Write your reflection here. . .

Appendix

```r
sessionInfo()
```

```
## R version 4.5.1 (2025-06-13)
## Platform: aarch64-apple-darwin24.4.0
## Running under: macOS Sequoia 15.6.1
##
## Matrix products: default
## BLAS:   /opt/homebrew/Cellar/openblas/0.3.30/lib/libopenblasp-r0.3.30.dylib
## LAPACK: /opt/homebrew/Cellar/r/4.5.1/lib/R/lib/libRlapack.dylib;  LAPACK version 3.12.1
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: Europe/Amsterdam
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
##  [1] compiler_4.5.1    fastmap_1.2.0     cli_3.6.5         tools_4.5.1
##  [5] htmltools_0.5.8.1 yaml_2.3.10       rmarkdown_2.30    knitr_1.50
##  [9] xfun_0.52         digest_0.6.37     rlang_1.1.6       evaluate_1.0.4
```