Assignment 2

Due 02/08/2019 by beginning of class

Instructions:

You may work with others on these problems but you must turn in your own work. You may type your solutions in any way you like (LaTeX, Markdown, Office, etc...) as long as you present your work clearly and in an organized way.

Unless otherwise specified, you must hand in a printed copy of your work at the beginning of class.

Problems:

1 Read the sections on "Data Types" and "Vectors and Type Coercion" from the Software Carpentry lesson on Datastructures (`https://swcarpentry.github.io/r-novice-gapminder/04-data-structures-part1/index.html`). We will try to fix the erroneously entered data in the 'feline-data_v2.csv'. In this case, we might say it's fair to enter the average of the two values, 2.35.

There were a number of legitimate interpretations to this question. I will outline one that highlights the issues with "factors".

1.1 Assign 'feline-data_v2.csv' to the variable 'cats'. Try the conversion function 'as.double()' on the vector *cats$weight*. What is the output in the console when you enter

$$print(cats\$weight) \tag{1}$$

into the console? Explain why you get this output.

Using the function *as.double*() on *cats$weight* in the console yields:

$$as.double(cats\$weight)$$

$$1 \quad 4 \quad 3 \quad 2$$

Entering,

$$print(cats\$weight)$$

2.1   5   3.2   2.3 or 2.4

Levels: 2.1 2.3 or 2.4 3.2 5

we see that R has read the inhomogeneous data in the vector *cats$weight* as levels of a factor vector.

1.2 This doesn't seem to solve the issues with the feline data version 2. Sometimes it is easier to work with CHAR types of data. Try the conversion function '*as.character*()' on *cats$weight*. What is the output now when enter

$$print(cats\$weight) \tag{2}$$

into the console?

If at this moment we re-assign the value of $as.char(cats\$weight)$ back into $cats\$weight$ we obtain

$$\begin{aligned}&cats\$weight \leftarrow as.character(cats\$weight)\\&print(cats\$weight)\\&\text{“2.1”}\quad \text{“5”}\quad \text{“3.2”}\quad \text{“2.3 } or \text{ 2.4”}\end{aligned}$$

Apologies that this was phrased badly.

1.3 Try assigning the character vector to a variable we call ‘*weights*’. Try the following assignment statement of the DOUBLE type value,

$$weights[4] \leftarrow 2.35$$

Now, what type is the fourth entry of the vector ‘*weights*’? Why is this?

Making the following assignments,

$$\begin{aligned}&weights \leftarrow as.character(cats\$weight)\\&weights[4] \leftarrow 2.35\\&print(weights[4])\\&\text{“2.35”}\end{aligned}$$

Vectors must have homogeneous data. When entering a double into a character vector, it is forced into the character type.

1.4 Explain how you can finally get the fourth cat’s weight to be listed as 2.35 in the dataframe ‘cats’, and to be of the right type.

Assigning the variable weights from 1.3 with the conversion function,

$$cats\$weights \leftarrow as.double(weights)$$

will achieve this result.

2 We will explore the gapminder data once again. Save this data into the file called "gapminder_data.csv". Recall, the data can be downloaded from the address: `https://raw.githubusercontent.com/swcarpentry/r-novice-gapminder/gh-pages/_episodes_rmd/data/gapminder_data.csv` and saved into this file using commands directly into R.

2.1 We want to do some regional analysis. Write an R *script* that performs the following:

- read the gapminder data into the variable "*gapminder*";
- make a multi-panel plot, where each frame corresponds to one of the five regions: "Africa", "Americas", "Asia", "Europe", "Oceania";
- the vertical axis should correspond to life expectancy and the horizontal axis should correspond to the year;
- data points should be represented by dots, color coded by region with a color-key in the figure;
- include a trend-line using a linear model for each region along with the data points;
- the figure should include the following labels: x-axis = "Year"; y-axis = "Life expectancy"; Figure title = "Regional life expectancy trends"; color-key title = "Region".

For credit, you must submit both the figure and your script.
Hint: You can produce this with a combination of the functions used in the Software Carpentry lesson "Creating Publication-Quality Graphics with ggplot2"

We produce Figure 1 and save it to a .png file using the script in Figure 2:

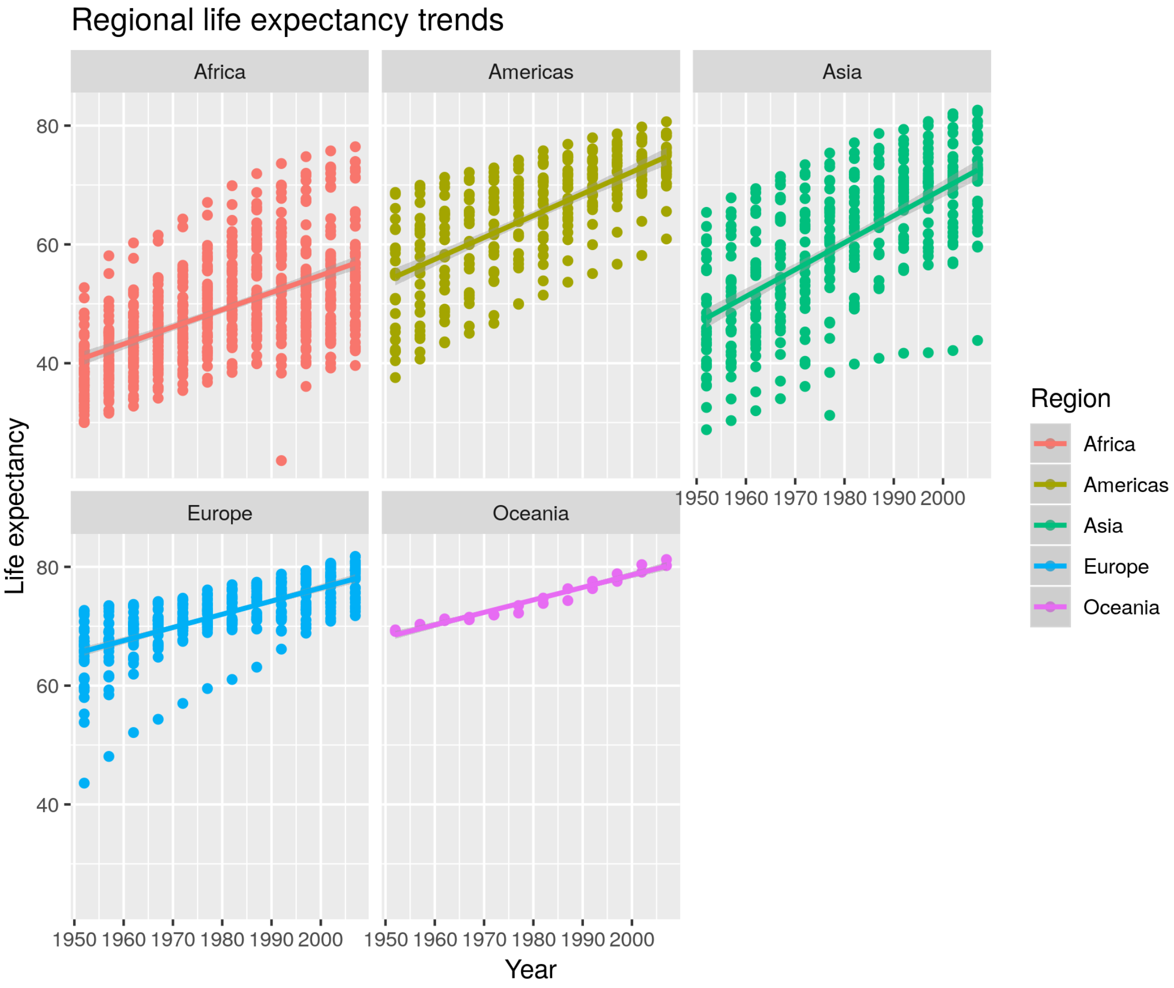
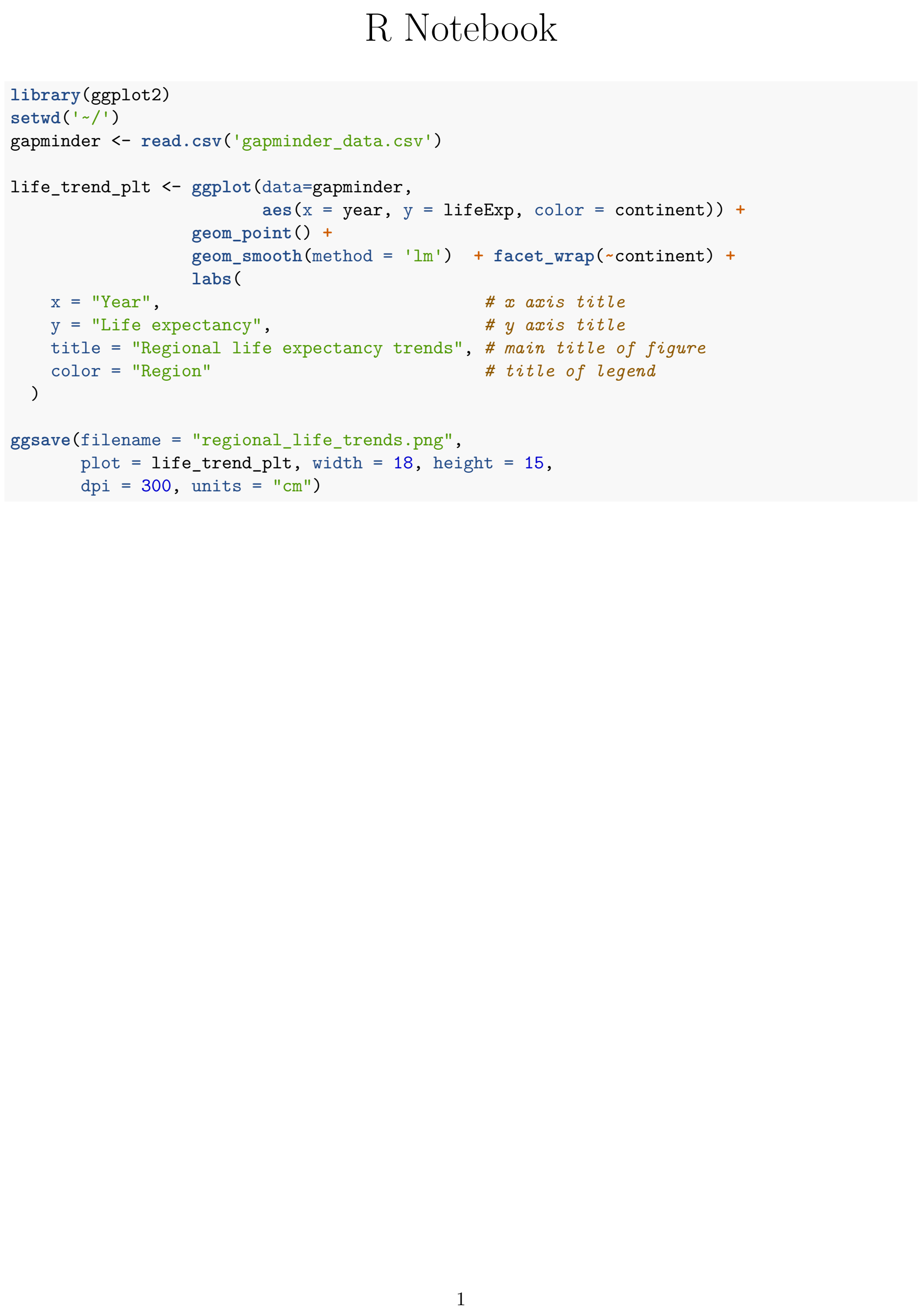


Figure 1: Regional life expectancy trends.



R Notebook

```r
library(ggplot2)
setwd('~/')
gapminder <- read.csv('gapminder_data.csv')

life_trend_plt <- ggplot(data=gapminder,
                         aes(x = year, y = lifeExp, color = continent)) +
                  geom_point() +
                  geom_smooth(method = 'lm')  + facet_wrap(~continent) +
                  labs(
    x = "Year",                                # x axis title
    y = "Life expectancy",                     # y axis title
    title = "Regional life expectancy trends", # main title of figure
    color = "Region"                           # title of legend
  )

ggsave(filename = "regional_life_trends.png",
       plot = life_trend_plt, width = 18, height = 15,
       dpi = 300, units = "cm")
```

1

Figure 2: Plotting script including save command.