HW2: Explore Gapminder and use dplyr

Carleena Ortega

27/09/2019

Exercise 1

1.1 Filter

Use `filter()` to subset the `gapminder` data to three countries of your choice in the 1970's.

```
filtered <- gapminder %>%
  arrange(year) %>%
  filter(year > 1969, year < 1980, country == "Canada" | country == "Mexico"
    | country == "Brazil") %>%
  arrange(country)
```

1.2 Pipe Operator

Use the pipe operator %>% to select "country" and "gdpPercap" from your filtered dataset in 1.1.

```
filtered %>%
  select(country, gdpPercap)
```

```
## # A tibble: 6 x 2
##   country gdpPercap
##   <fct>       <dbl>
## 1 Brazil      4986.
## 2 Brazil      6660.
## 3 Canada     18971.
## 4 Canada     22091.
## 5 Mexico      6809.
## 6 Mexico      7675.
```

1.3 Drop in Life Expectancy

Filter gapminder to all entries that have experienced a drop in life expectancy. Be sure to include a new variable that's the increase in life expectancy in your tibble. Hint: you might find the lag() or diff() functions useful

```
gapminder %>%
  group_by(country) %>%
  arrange(year) %>%
  mutate(ch_LE=lifeExp-first(lifeExp))
```

```
## # A tibble: 1,704 x 7
## # Groups:   country [142]
##    country     continent  year lifeExp      pop gdpPercap ch_LE
```

```
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779.     0
##  2 Albania     Europe     1952    55.2  1282697     1601.     0
##  3 Algeria     Africa     1952    43.1  9279525     2449.     0
##  4 Angola      Africa     1952    30.0  4232095     3521.     0
##  5 Argentina   Americas   1952    62.5 17876956     5911.     0
##  6 Australia   Oceania    1952    69.1  8691212    10040.     0
##  7 Austria     Europe     1952    66.8  6927772     6137.     0
##  8 Bahrain     Asia       1952    50.9   120447     9867.     0
##  9 Bangladesh  Asia       1952    37.5 46886859      684.     0
## 10 Belgium     Europe     1952    68    8730405     8343.     0
## # ... with 1,694 more rows
```

1.4

Choose one of the following:

```
Filter gapminder so that it shows the max GDP per capita experienced by each country. Hint: you might fi
```

OR

```
Filter gapminder to contain six rows: the rows with the three largest GDP per capita, and the rows with
```

1.5

Produce a scatterplot of Canada's life expectancy vs. GDP per capita using ggplot2, without defining a new variable. That is, after filtering the gapminder data set, pipe it directly into the ggplot() function. Ensure GDP per capita is on a log scale.

Exercise 2

Pick one categorical variable and one quantitative variable to explore. Answer the following questions in whichever way you think is appropriate, using dplyr:

```
What are possible values (or range, whichever is appropriate) of each variable?
What values are typical? What's the spread? What's the distribution? Etc., tailored to the variable at h
Feel free to use summary stats, tables, figures.
```

Exercise 3

Make two plots that have some value to them. That is, plots that someone might actually consider making for an analysis. Just don't make the same plots we made in class – feel free to use a data set from the datasets R package if you wish.

```
A scatterplot of two quantitative variables.
One other plot besides a scatterplot.
```

You don't have to use all the data in every plot! It's fine to filter down to one country or a small handful of countries.

Bonus

Bonus 1

For people who want to take things further.

Evaluate this code and describe the result. Presumably the analyst's intent was to get the data for Rwanda and Afghanistan. Did they succeed? Why or why not? If not, what is the correct way to do this?

Bonus 2

Present numerical tables in a more attractive form using knitr::kable() for small tibbles (say, up to 10 rows), and DT::datatable() for larger tibbles.