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(country,year) %>%
  mutate(change_LE=lifeExp-lag(lifeExp)) %>%
  filter(change_LE<0)
```

```
## # A tibble: 102 x 7
## # Groups:   country [52]
##    country  continent  year lifeExp     pop gdpPercap change_LE
##    <fct>    <fct>     <int>   <dbl>   <int>     <dbl>     <dbl>
##  1 Albania  Europe     1992    71.6 3326498     2497.    -0.419
##  2 Angola   Africa     1987    39.9 7874230     2430.    -0.036
##  3 Benin    Africa     2002    54.4 7026113     1373.    -0.371
##  4 Botswana Africa     1992    62.7 1342614     7954.    -0.877
##  5 Botswana Africa     1997    52.6 1536536     8647.   -10.2
##  6 Botswana Africa     2002    46.6 1630347    11004.    -5.92
##  7 Bulgaria Europe     1977    70.8 8797022     7612.    -0.09
##  8 Bulgaria Europe     1992    71.2 8658506     6303.    -0.15
##  9 Bulgaria Europe     1997    70.3 8066057     5970.    -0.87
## 10 Burundi  Africa     1992    44.7 5809236      632.    -3.48
## # ... with 92 more rows
```

1.4 Max()

Choose one of the following:

Filter gapminder so that it shows the max GDP per capita experienced by each country. Hint: you might find the max() function useful here.

OR

Filter gapminder to contain six rows: the rows with the three largest GDP per capita, and the rows with the three smallest GDP per capita. Be sure to not create any intermediate objects when doing this (with, for example, the assignment operator). Hint: you might find the sort() function useful, or perhaps even the dplyr::slice() function.

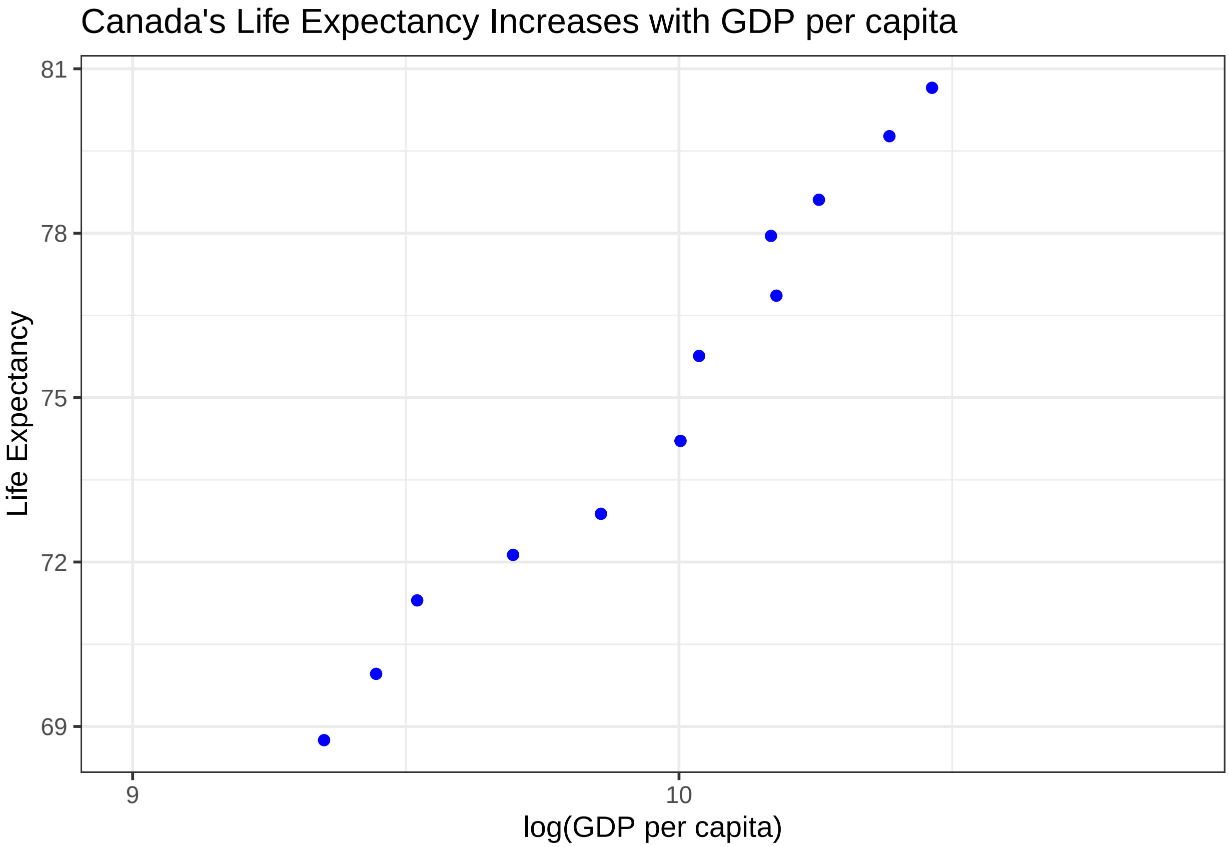
```
gapminder %>%
  group_by(country) %>%
  arrange(country,gdpPercap) %>%
  filter(gdpPercap==max(gdpPercap))
```

```
## # A tibble: 142 x 6
## # Groups:   country [142]
##    country     continent  year lifeExp       pop gdpPercap
##    <fct>       <fct>     <int>   <dbl>     <int>     <dbl>
##  1 Afghanistan Asia       1982    39.9  12881816      978.
##  2 Albania     Europe     2007    76.4   3600523     5937.
##  3 Algeria     Africa     2007    72.3  33333216     6223.
##  4 Angola      Africa     1967    36.0   5247469     5523.
##  5 Argentina   Americas   2007    75.3  40301927    12779.
##  6 Australia   Oceania    2007    81.2  20434176    34435.
##  7 Austria     Europe     2007    79.8   8199783    36126.
##  8 Bahrain     Asia       2007    75.6    708573    29796.
##  9 Bangladesh  Asia       2007    64.1 150448339     1391.
## 10 Belgium     Europe     2007    79.4  10392226    33693.
## # ... with 132 more rows
```

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.

```
gapminder %>%
  filter(country == "Canada") %>%
  ggplot(aes(x=log(gdpPercap),lifeExp)) +
  scale_x_log10(limits=c(9,11)) +
  geom_point(colour="blue") +
  labs(x="log(GDP per capita)",
    y="Life Expectancy",
    title="Canada's Life Expectancy Increases with GDP per capita") +
  theme_bw()
```



Exercise 2

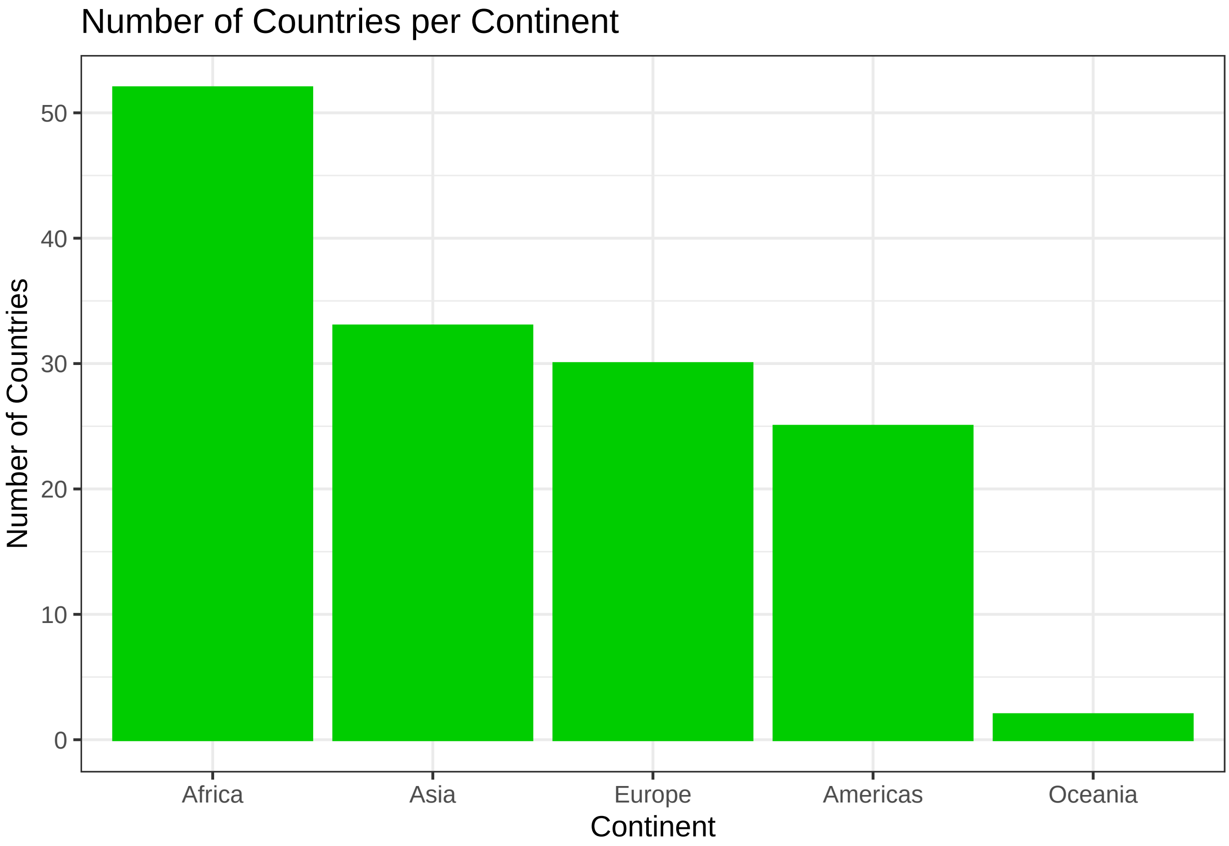
Pick one categorical variable and one quantitative variable to explore. Answer the following questions in whichever way you think is appropriate, using dplyr: What are possible values (or range, whichever is appropriate) of each variable? What values are typical? What's the spread? What's the distribution? Etc., tailored to the variable at hand. Feel free to use summary stats, tables, figures.

For this exercise, we will use `continent` as a categorical variable and `pop` as quantitative variable from the `gapminder` data set.

2.1 Gapminder Continent

We can see that there are 5 continents: **Africa, Asia, Europe, Americas, and Oceania** Africa has the most number of continents and Oceania with the fewest countries.

```
  gapminder %>%
  arrange(country)%>%
  filter(year == 1952) %>%
  mutate(continent = fct_infreq(continent)) %>%
  ggplot(aes(continent)) +
  geom_bar(colour="3",fill="3") +
  labs(x="Continent",
    y="Number of Countries",
    title="Number of Countries per Continent") +
  theme_bw()
```



We can also view this data as a table:

```
gapminder %>%
  filter(year == 1952) %>%
  group_by(continent) %>%
  summarize(number_of_countries = mean(length(country)))
```

```
## # A tibble: 5 x 2
##   continent number_of_countries
##   <fct>                   <dbl>
## 1 Africa                          52
## 2 Americas                        25
## 3 Asia                            33
## 4 Europe                          30
## 5 Oceania                          2
```

2.2 Gapminder Pop

The mean population of the continents over the years are shown below:

```
gapminder %>%
  group_by(continent) %>%
  summarize(mean_popuplation=mean(pop))
```

```
## # A tibble: 5 x 2
##   continent mean_popuplation
##   <fct>                <dbl>
## 1 Africa            9916003.
## 2 Americas         24504795.
## 3 Asia             77038722.
## 4 Europe           17169765.
## 5 Oceania           8874672.
```

The standard error of the population per country is shown below:

```
gapminder %>%
  group_by(continent) %>%
  summarize(stdE_pop = sd(pop)/sqrt(n()))
```

```
## # A tibble: 5 x 2
##   continent  stdE_pop
##   <fct>         <dbl>
## 1 Africa      620133.
## 2 Americas   2943299.
## 3 Asia      10396373.
## 4 Europe     1081469.
## 5 Oceania    1328102.
```

The ranges of population for each continent is shown below:

```
gapminder %>%
  group_by(continent) %>%
  summarize(min(pop),max(pop))
```

```
## # A tibble: 5 x 3
##   continent `min(pop)` `max(pop)`
##   <fct>          <int>      <int>
## 1 Africa         60011  135031164
## 2 Americas      662850  301139947
## 3 Asia          120447 1318683096
## 4 Europe        147962   82400996
## 5 Oceania      1994794   20434176
```

This shows that Asia has the most population at 1,318,683,096 and Africa had the least at 60,011.

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.