

# w6\_assignment

Frida Hæstrup

25/10/2020

Loading packages

```
library(pacman)
p_load(tidyverse, gganimate, gapminder, gifski, png, readr)
```

1)

Define a defensive function that calculates the Gross Domestic Product of a nation from the data available in the gapminder dataset. Using that function, calculate the GDP of Denmark in the following years: 1967, 1977, 1987, 1997, 2007, and 2017.

*# Inspecting data*

```
head(gapminder)
```

```
## # A tibble: 6 x 6
##   country    continent  year lifeExp      pop gdpPercap
##   <fct>      <fct>    <int>  <dbl>    <int>    <dbl>
## 1 Afghanistan Asia      1952   28.8  8425333    779.
## 2 Afghanistan Asia      1957   30.3  9240934    821.
## 3 Afghanistan Asia      1962   32.0 10267083    853.
## 4 Afghanistan Asia      1967   34.0 11537966    836.
## 5 Afghanistan Asia      1972   36.1 13079460    740.
## 6 Afghanistan Asia      1977   38.4 14880372    786.
```

*# Defining function for calculating GDP*

```
gdp_function <- function(nation, time){
  this_subset <- gapminder %>%
    subset(country == nation & year == time)

  if (!is.numeric(this_subset$pop)) {
    stop("pop must be a numeric vector.")
  }
  if (!is.numeric(this_subset$gdpPercap)) {
    stop("gdpPercap must be a numeric vector.")
  }

  this_subset <- this_subset %>% mutate(pop*gdpPercap)
```

```

GDP <- this_subset$`pop` * gdpPercap`[1]

  return(GDP)
}

# Calculating GDP in Denmark during the years 1967, 1977, 1987, 1997, 2007, and 2017
gdp_function('Denmark', 1967)
## [1] 77116977700

gdp_function('Denmark', 1977)
## [1] 103920280028

gdp_function('Denmark', 1987)
## [1] 128771236166

gdp_function('Denmark', 1997)
## [1] 157476118456

gdp_function('Denmark', 2007)
## [1] 192906627081

gdp_function('Denmark', 2017)
## [1] NA

```

In order to calculate the GDP of a nation, we can use the information provided by data set on the population and GDP per capita for each nation per year. By multiplying the 'gdpPercap' column with the 'pop' column, we get the GDP of the nation for that year. This is then put into a function and applied to Denmark for the relevant years. The results can be seen above.

## 2)

**Write a script that loops over each country in the gapminder dataset, tests whether the country starts with a 'B', and print out whether the life expectancy is smaller than 50, between 50 and 70, or greater than 70.**

```

B_df <- as.data.frame(matrix(0, nrow = 1, ncol = 8)) #new df to be filled with data from loop
colnames(B_df) <- c(colnames(gapminder), "starts_with_B", "range_life_exp")

for (i in gapminder$country){ #Looping through countries
  current_subset <- gapminder %>%
    subset(country == i)

```

```

for (j in current_subset$year){ #looping through years for that country
  new_current_subset <- current_subset %>%
    subset(year == j)

  new_current_subset$starts_with_B <- startswith('B', i) #returns 1 in new
column if country starts with B, 0 if not

  #determining size of life exp and adding this info to new column
  if (new_current_subset$lifeExp < 50){
    new_current_subset$range_life_exp <- "smaller than 50"
  }
  if (new_current_subset$lifeExp >= 50 & new_current_subset$lifeExp <= 70){
    new_current_subset$range_life_exp <- "between 50 and 70"
  }
  if (new_current_subset$lifeExp > 70){
    new_current_subset$range_life_exp <- "greater than 70"
  }

  B_df <- rbind(B_df, new_current_subset) #binding to B_df for each iteration of
the loop
}
}
head(B_df[-1,-2])

##      country year lifeExp      pop gdpPercap starts_with_B  range_life_exp
## 2 Afghanistan 1952  28.801  8425333   779.4453           0 smaller than 50
## 3 Afghanistan 1957  30.332  9240934   820.8530           0 smaller than 50
## 4 Afghanistan 1962  31.997 10267083   853.1007           0 smaller than 50
## 5 Afghanistan 1967  34.020 11537966   836.1971           0 smaller than 50
## 6 Afghanistan 1972  36.088 13079460   739.9811           0 smaller than 50
## 7 Afghanistan 1977  38.438 14880372   786.1134           0 smaller than 50

```

This loop returns a dataframe with information on every country for each year about the range of the lifeExp and whether country starts with 'B'.