Hw 5

Aylin Mumcular

16 10 2019

install.packages("tidyverse") install.packages("dplyr") install.packages("gapminder") install.packages("forcats") install.packages("here") install.packages("ggrepel") install.packages("ggpubr") —

Exercise 1: Explain the value of the here::here package

Here::Here package is quite useful because this package makes it robust to access codes, no matter the file paths of different users. It can automatically adapt to different operating systems and directories. Only specifying the paths from the root directory to sub-directory is sufficient for users. In other words, its ability to detect the root directory and work platform-independently allows users to exchange and run codes without creating the same directories with the code owners. Using this package enhances the reproducibility of codes.

Exercise 2: Factor management

Drop factor/levels

```
gapminder <- gapminder::gapminder

#Explore continent variable from the gapminder dataset

gapminder$continent %>% class #Continent feature is a factor
```

```
## [1] "factor"
```

```
gapminder$continent %>% levels #I will drop Oceania among these levels
```

```
## [1] "Africa"   "Americas" "Asia"     "Europe"   "Oceania"
```

```
gapminder$country %>% levels
```

```
##   [1] "Afghanistan"              "Albania"
##   [3] "Algeria"                  "Angola"
##   [5] "Argentina"                "Australia"
##   [7] "Austria"                  "Bahrain"
##   [9] "Bangladesh"               "Belgium"
##  [11] "Benin"                    "Bolivia"
##  [13] "Bosnia and Herzegovina"   "Botswana"
##  [15] "Brazil"                   "Bulgaria"
##  [17] "Burkina Faso"             "Burundi"
##  [19] "Cambodia"                 "Cameroon"
##  [21] "Canada"                   "Central African Republic"
##  [23] "Chad"                     "Chile"
##  [25] "China"                    "Colombia"
##  [27] "Comoros"                  "Congo, Dem. Rep."
##  [29] "Congo, Rep."              "Costa Rica"
##  [31] "Cote d'Ivoire"            "Croatia"
##  [33] "Cuba"                     "Czech Republic"
##  [35] "Denmark"                  "Djibouti"
##  [37] "Dominican Republic"       "Ecuador"
##  [39] "Egypt"                    "El Salvador"
##  [41] "Equatorial Guinea"        "Eritrea"
##  [43] "Ethiopia"                 "Finland"
```

```
##  [45] "France"                   "Gabon"
##  [47] "Gambia"                   "Germany"
##  [49] "Ghana"                    "Greece"
##  [51] "Guatemala"                "Guinea"
##  [53] "Guinea-Bissau"            "Haiti"
##  [55] "Honduras"                 "Hong Kong, China"
##  [57] "Hungary"                  "Iceland"
##  [59] "India"                    "Indonesia"
##  [61] "Iran"                     "Iraq"
##  [63] "Ireland"                  "Israel"
##  [65] "Italy"                    "Jamaica"
##  [67] "Japan"                    "Jordan"
##  [69] "Kenya"                    "Korea, Dem. Rep."
##  [71] "Korea, Rep."              "Kuwait"
##  [73] "Lebanon"                  "Lesotho"
##  [75] "Liberia"                  "Libya"
##  [77] "Madagascar"               "Malawi"
##  [79] "Malaysia"                 "Mali"
##  [81] "Mauritania"               "Mauritius"
##  [83] "Mexico"                   "Mongolia"
##  [85] "Montenegro"               "Morocco"
##  [87] "Mozambique"               "Myanmar"
##  [89] "Namibia"                  "Nepal"
##  [91] "Netherlands"              "New Zealand"
##  [93] "Nicaragua"                "Niger"
##  [95] "Nigeria"                  "Norway"
##  [97] "Oman"                     "Pakistan"
##  [99] "Panama"                   "Paraguay"
## [101] "Peru"                     "Philippines"
## [103] "Poland"                   "Portugal"
## [105] "Puerto Rico"              "Reunion"
## [107] "Romania"                  "Rwanda"
## [109] "Sao Tome and Principe"    "Saudi Arabia"
## [111] "Senegal"                  "Serbia"
## [113] "Sierra Leone"             "Singapore"
## [115] "Slovak Republic"          "Slovenia"
## [117] "Somalia"                  "South Africa"
## [119] "Spain"                    "Sri Lanka"
## [121] "Sudan"                    "Swaziland"
## [123] "Sweden"                   "Switzerland"
## [125] "Syria"                    "Taiwan"
## [127] "Tanzania"                 "Thailand"
## [129] "Togo"                     "Trinidad and Tobago"
## [131] "Tunisia"                  "Turkey"
## [133] "Uganda"                   "United Kingdom"
## [135] "United States"            "Uruguay"
## [137] "Venezuela"                "Vietnam"
## [139] "West Bank and Gaza"       "Yemen, Rep."
## [141] "Zambia"                   "Zimbabwe"
```

```
nrow(gapminder) #Number of rows before dropping Oceania
```

```
## [1] 1704
```

```
#Before dropping, there are 1704 rows, 5 levels for continents, and 142 levels of countries

gap_drop <- gapminder %>% filter(continent != "Oceania") %>% droplevels

gap_drop$continent %>% levels
```

```
## [1] "Africa"   "Americas" "Asia"     "Europe"
```

```
gap_drop$country %>% levels
```

```
##   [1] "Afghanistan"              "Albania"
##   [3] "Algeria"                  "Angola"
##   [5] "Argentina"                "Austria"
##   [7] "Bahrain"                  "Bangladesh"
##   [9] "Belgium"                  "Benin"
##  [11] "Bolivia"                  "Bosnia and Herzegovina"
##  [13] "Botswana"                 "Brazil"
##  [15] "Bulgaria"                 "Burkina Faso"
##  [17] "Burundi"                  "Cambodia"
##  [19] "Cameroon"                 "Canada"
##  [21] "Central African Republic" "Chad"
##  [23] "Chile"                    "China"
##  [25] "Colombia"                 "Comoros"
##  [27] "Congo, Dem. Rep."         "Congo, Rep."
##  [29] "Costa Rica"               "Cote d'Ivoire"
##  [31] "Croatia"                  "Cuba"
##  [33] "Czech Republic"           "Denmark"
##  [35] "Djibouti"                 "Dominican Republic"
##  [37] "Ecuador"                  "Egypt"
##  [39] "El Salvador"              "Equatorial Guinea"
##  [41] "Eritrea"                  "Ethiopia"
##  [43] "Finland"                  "France"
##  [45] "Gabon"                    "Gambia"
##  [47] "Germany"                  "Ghana"
##  [49] "Greece"                   "Guatemala"
##  [51] "Guinea"                   "Guinea-Bissau"
##  [53] "Haiti"                    "Honduras"
##  [55] "Hong Kong, China"         "Hungary"
##  [57] "Iceland"                  "India"
##  [59] "Indonesia"                "Iran"
##  [61] "Iraq"                     "Ireland"
##  [63] "Israel"                   "Italy"
##  [65] "Jamaica"                  "Japan"
##  [67] "Jordan"                   "Kenya"
##  [69] "Korea, Dem. Rep."         "Korea, Rep."
##  [71] "Kuwait"                   "Lebanon"
##  [73] "Lesotho"                  "Liberia"
##  [75] "Libya"                    "Madagascar"
##  [77] "Malawi"                   "Malaysia"
##  [79] "Mali"                     "Mauritania"
##  [81] "Mauritius"                "Mexico"
##  [83] "Mongolia"                 "Montenegro"
##  [85] "Morocco"                  "Mozambique"
##  [87] "Myanmar"                  "Namibia"
```

```
##  [89] "Nepal"                    "Netherlands"
##  [91] "Nicaragua"                "Niger"
##  [93] "Nigeria"                  "Norway"
##  [95] "Oman"                     "Pakistan"
##  [97] "Panama"                   "Paraguay"
##  [99] "Peru"                     "Philippines"
## [101] "Poland"                   "Portugal"
## [103] "Puerto Rico"              "Reunion"
## [105] "Romania"                  "Rwanda"
## [107] "Sao Tome and Principe"    "Saudi Arabia"
## [109] "Senegal"                  "Serbia"
## [111] "Sierra Leone"             "Singapore"
## [113] "Slovak Republic"          "Slovenia"
## [115] "Somalia"                  "South Africa"
## [117] "Spain"                    "Sri Lanka"
## [119] "Sudan"                    "Swaziland"
## [121] "Sweden"                   "Switzerland"
## [123] "Syria"                    "Taiwan"
## [125] "Tanzania"                 "Thailand"
## [127] "Togo"                     "Trinidad and Tobago"
## [129] "Tunisia"                  "Turkey"
## [131] "Uganda"                   "United Kingdom"
## [133] "United States"            "Uruguay"
## [135] "Venezuela"                "Vietnam"
## [137] "West Bank and Gaza"       "Yemen, Rep."
## [139] "Zambia"                   "Zimbabwe"
```

```
nrow(gap_drop)
```

```
## [1] 1680
```

```
#After dropping, there are 1680 rows, 4 levels for continents, and 140 levels of countries
```

Reorder levels based on knowledge from data

```
#Let's choose 25th quantile of life expectancy as the summary statistics for reordering

func <- function(x) {return(quantile(x, 0.25))}

#Reorder continent levels

gap_drop2 <- gap_drop %>% mutate(continent = fct_reorder(continent, lifeExp, func, .desc = TRUE))


gap_drop2$continent %>% levels
```

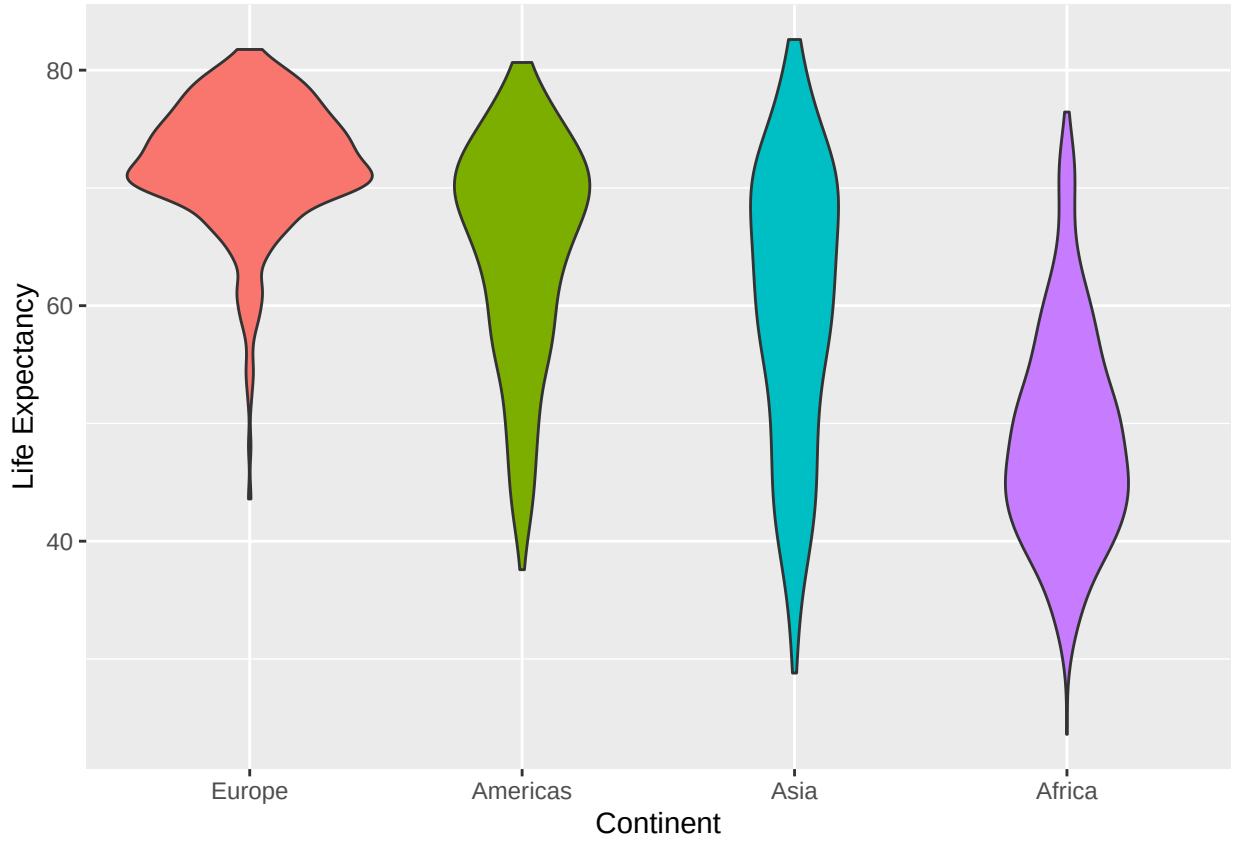
```
## [1] "Europe"   "Americas" "Asia"     "Africa"
```

```
#The order of the levels changed from "Africa" "Americas" "Asia" "Europe" to "Europe" "Americas" "Asia"
```

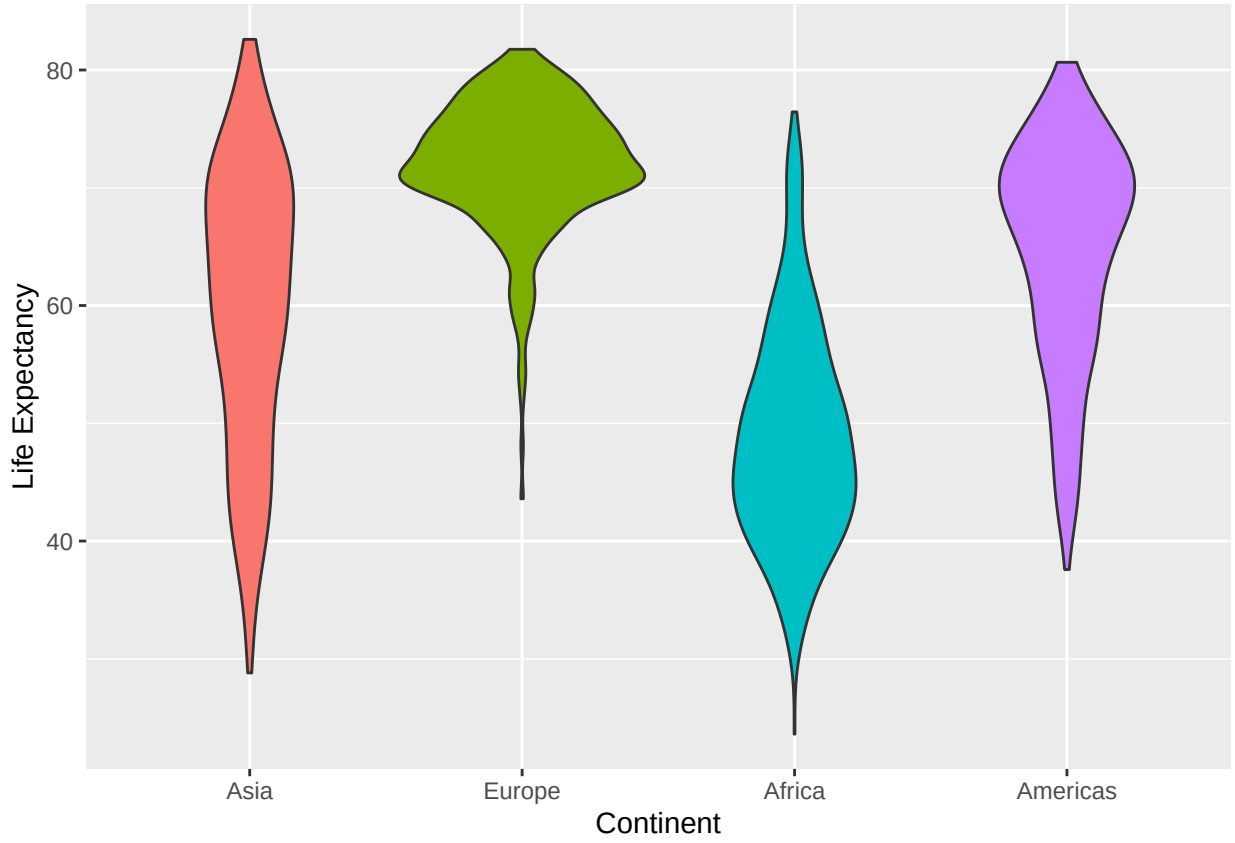
Explore the effects of re-leveling a factor in a tibble

```
#Before re-leveling

gd2 <- gap_drop2 %>% arrange(continent)

gd2 %>%
    ggplot( aes(x = continent, y = lifeExp, fill = continent)) +
```

```
    geom_violin() +
    xlab("Continent") +
    theme(legend.position="none") +
    ylab("Life Expectancy")
```




```
#After re-leveling

gap_drop3 <- gap_drop2 %>%
                mutate(continent = fct_relevel(continent, c("Asia", "Europe", "Africa", "Americas")))

gd3 <- gap_drop3 %>% arrange(continent)

gd3 %>%
    ggplot( aes(x = continent, y = lifeExp, fill = continent)) +
    geom_violin() +
    xlab("Continent") +
    theme(legend.position="none") +
    ylab("Life Expectancy")
```



Exercise 3: File input/output

```
new_data <- gapminder[2:30, ] #A subset of the gapminder data

write_csv(new_data, here("Homework 5", "new_file.csv")) #Export

imported_data <- read_csv(here("Homework 5", "new_file.csv")) #Reload
```

```
## Parsed with column specification:
## cols(
##   country = col_character(),
##   continent = col_character(),
##   year = col_double(),
##   lifeExp = col_double(),
##   pop = col_double(),
##   gdpPercap = col_double()
## )
```

```
imported_data
```

```
## # A tibble: 29 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <chr>       <chr>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 Afghanistan Asia       1957    30.3  9240934      821.
##  2 Afghanistan Asia       1962    32.0 10267083      853.
##  3 Afghanistan Asia       1967    34.0 11537966      836.
##  4 Afghanistan Asia       1972    36.1 13079460      740.
##  5 Afghanistan Asia       1977    38.4 14880372      786.
```

```
##  6 Afghanistan Asia       1982    39.9 12881816      978.
##  7 Afghanistan Asia       1987    40.8 13867957      852.
##  8 Afghanistan Asia       1992    41.7 16317921      649.
##  9 Afghanistan Asia       1997    41.8 22227415      635.
## 10 Afghanistan Asia       2002    42.1 25268405      727.
## # ... with 19 more rows
```

```
#Data is reloaded successfully

gd4 <- imported_data %>%
         mutate(country = factor(country), #Convert from char to factor
         continent = factor(continent),
         continent = fct_reorder(continent, pop, mean, .desc = TRUE)) %>%
         arrange(continent)

gd4
```

```
## # A tibble: 29 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fct>       <fct>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 Afghanistan Asia       1957    30.3  9240934      821.
##  2 Afghanistan Asia       1962    32.0 10267083      853.
##  3 Afghanistan Asia       1967    34.0 11537966      836.
##  4 Afghanistan Asia       1972    36.1 13079460      740.
##  5 Afghanistan Asia       1977    38.4 14880372      786.
##  6 Afghanistan Asia       1982    39.9 12881816      978.
##  7 Afghanistan Asia       1987    40.8 13867957      852.
##  8 Afghanistan Asia       1992    41.7 16317921      649.
##  9 Afghanistan Asia       1997    41.8 22227415      635.
## 10 Afghanistan Asia       2002    42.1 25268405      727.
## # ... with 19 more rows
```

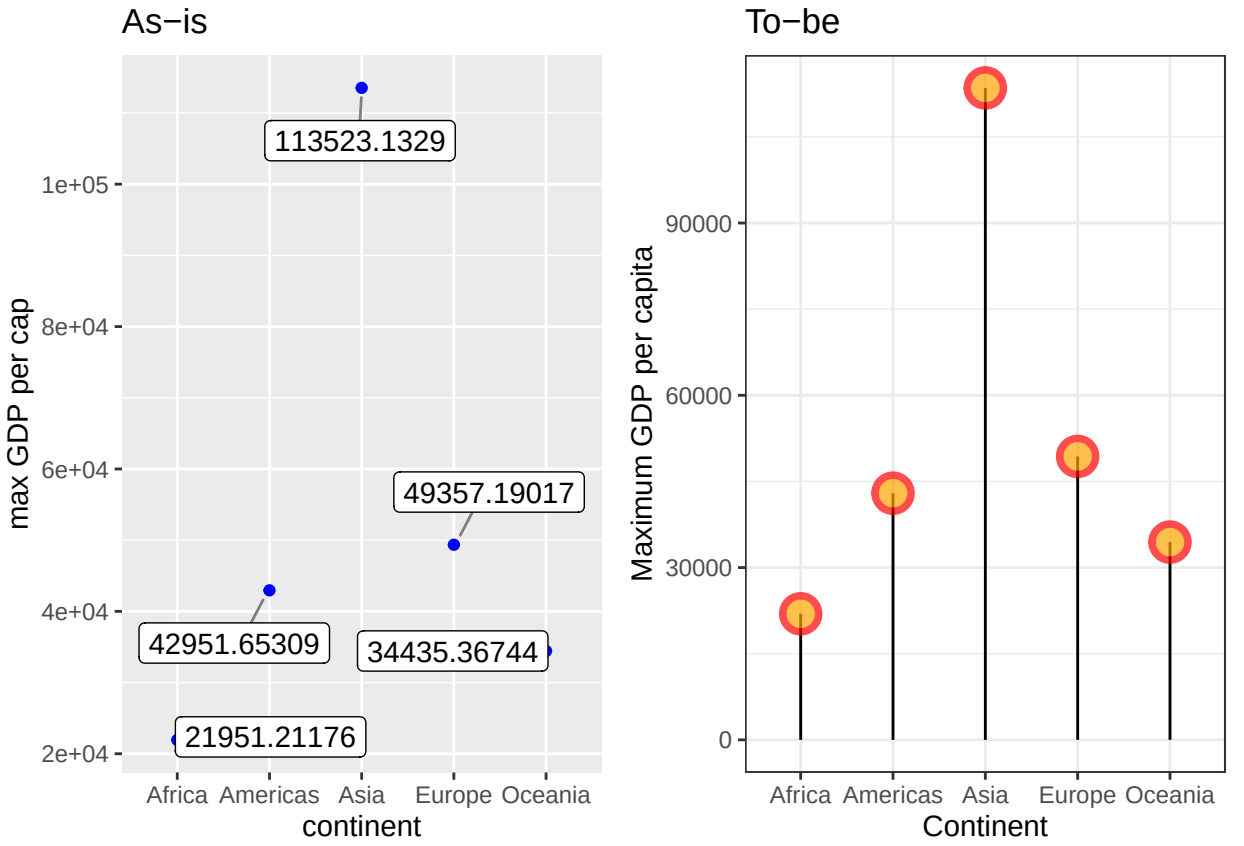
```
gd4$continent %>% levels
```

```
## [1] "Asia"   "Africa" "Europe"
```

Exercise 4: Visualization design

```
#Max GDP per capita for each continent plot from Homework 3

p1 <- gapminder %>%
group_by(continent) %>%
summarize(maxGDPpercap = max(gdpPercap),
minGDPpercap = min(gdpPercap)) %>%
ggplot(aes(continent, maxGDPpercap)) +
geom_point(colour = "blue") +
geom_label_repel(aes(label = maxGDPpercap),
box.padding = 0.35,
point.padding = 0.5,
segment.color = 'grey50') +
ylab("max GDP per cap") + ggtitle("As-is")

#New plot

p2 <- gapminder %>%
group_by(continent) %>%
```

```r
summarize(maxGDPpercap = max(gdpPercap)) %>%
ggplot(aes(x=continent, y=maxGDPpercap)) +
geom_segment(aes(x=continent, xend=continent, y=0, yend=maxGDPpercap)) +
geom_point( size=5, color="red", fill=alpha("orange", 0.3), alpha=0.7, shape=21, stroke=2)+
xlab("Continent") + ylab("Maximum GDP per capita") + ggtitle("To-be") +
theme_bw()


ggarrange(p1, p2, nrow = 1, ncol = 2) #Combine two plots
```



- In the first plot, it is very difficult to read the values corresponding to the y-axis. To solve this problem, I added labels showing the actual values. However, this made the first plot very cluttered. In the second plot, the y-axis is scaled according to the possible range.
- In the first plot, the background is gray. Having a colorful background makes the plot harder to interpret because of its distractive nature. To solve this problem, I changed the background color to white in the second plot.
- In the first plot, I used abbreviations to name the y-axis. In the second plot, I changed how I named the axes by openly writing the labels and making them more understandable for someone does not know the data.

Exercise 5: Writing figures to file

```r
ggsave(here("Homework 5", "To-be.png"), plot = p2, width = 4, height = 4) #Save the new plot as png
```

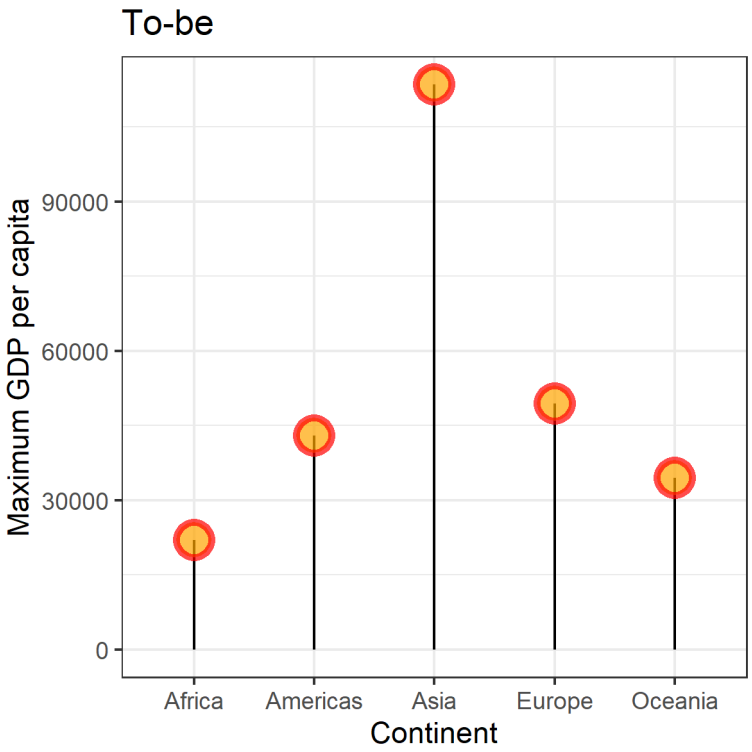


Figure 1: *New Plot*