Gapminder Data Analysis using Tidyverse

Shahbaz Alam

4/23/2021

Synopsis

This is a sample introduction of Data wrangling using R with the help of the most famous bundle of R packages named ***Tidyverse***. In this introductory document I'm trying to present some basic norms when a data analyst gets the data for the first time and want to grow some intuition about the data.

Environment Setup

So first things first, we need to do some initial setup, means setting environment, install/load required packages, explore the structure of the data and perform some exploratory analysis. I'll use the ***gapminder*** data-set for data wrangling and ***ggplot2*** for exploratory analysis. This data come with the ***gapminder*** package and this report is made in ***R Markdown*** . Most of the visualization codes are not displayed here to increase readability. Complete code for this report will be available on my ***Github Repository***. So, let there be light!

```
rm(list = ls())
knitr::opts_chunk$set(echo = TRUE)

list.of.packages <- c("rstudioapi","tidyverse", "ggplot2", "gapminder",
                      "ggalt", "ggpubr", "ggtext", "ggrepel")
new.packages <- list.of.packages[!(list.of.packages %in%
                                     installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

library(tidyverse)
library(gapminder)
options(dplyr.summarise.inform = FALSE)
```

Now, we load the *gapminder* data and see the structure of the data. This is the first thing we need to do, when we get any data. This will help us when we create new variables from the existing ones. Sometimes desired output cannot be obtained because of not knowing the data type/variable type. In that case, ***class()*** function helps a lot.

```
data(gapminder, package = "gapminder")
str(gapminder)
```

```
## tibble [1,704 x 6] (S3: tbl_df/tbl/data.frame)
##  $ country  : Factor w/ 142 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ continent: Factor w/ 5 levels "Africa","Americas",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ year     : int [1:1704] 1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
##  $ lifeExp  : num [1:1704] 28.8 30.3 32 34 36.1 ...
##  $ pop      : int [1:1704] 8425333 9240934 10267083 11537966 13079460 14880372 12881816 13867957 163
##  $ gdpPercap: num [1:1704] 779 821 853 836 740 ...
```

So, the *gapminder* data contains GDP, population and life expectancy information of 142 countries with their

continents from year 1952 to 2007. ***help()*** function provide some basic details about variables and more information can be obtained from gapminder.org. After loading the data, one very important function is ***View(data_name)*** to see all data in different tab.

Data Wrangling and exploratory Analysis(EDA)

To perform data wrangling, we'll use ***dplyr*** package. The ***dplyr*** package is part of the ***tidyverse***. It provides a number of very useful functions for manipulating *tibbles* (and their base-R cousin, the *data.frame*) in a way that will reduce repetition, reduce the probability of making errors, and probably even save you some time of typing. Here is a glance of this functions working :

- selecting variables with ***select()***
- subsetting observations with ***filter()***
- grouping observations with ***group_by()***
- generating summary statistics using ***summarize()***
- generating new variables using ***mutate()***
- Sorting tibbles using ***arrange()***
- chaining operations together using pipes *%>%*

Now, we try to do some exploratory analysis to get some valuable information. First, we want to know which 10 countries in Asia has the highest life expectancy in 2007.

```
asiaTopTenLifeExp <- gapminder %>%
  filter(continent == "Asia", year == 2007) %>%
  select(country, lifeExp) %>%
  arrange(desc(lifeExp)) %>%
  mutate(lifeExpRank = 1:n()) %>%
  slice(1:10)
# knitr::kable(asiaTopTenLifeExp)
# knitr::kable(asiaTopTenLifeExp, format = "latex", booktabs = TRUE)
print(asiaTopTenLifeExp)
```

```
## # A tibble: 10 x 3
##    country          lifeExp lifeExpRank
##    <fct>              <dbl>       <int>
##  1 Japan               82.6           1
##  2 Hong Kong, China    82.2           2
##  3 Israel              80.7           3
##  4 Singapore           80.0           4
##  5 Korea, Rep.         78.6           5
##  6 Taiwan              78.4           6
##  7 Kuwait              77.6           7
##  8 Oman                75.6           8
##  9 Bahrain             75.6           9
## 10 Vietnam             74.2          10
```

So, **Japan** has the highest life expectancy not only in Asia but also globally! What about *Bangladesh* in 2007? Lets see

```
bdLifeExpRank <- gapminder %>%
  filter(continent == "Asia", year == 2007) %>%
  select(country, lifeExp) %>%
  arrange(desc(lifeExp)) %>%
  mutate(lifeExpRank = 1:n()) %>%
  filter(country == "Bangladesh")

bdLifeExpWorldRank <- gapminder %>%
```

```
  filter(year == 2007) %>%
  select(country, lifeExp) %>%
  arrange(desc(lifeExp)) %>%
  mutate(lifeExpRank = 1:n()) %>%
  filter(country == "Bangladesh")

paste0(bdLifeExpRank$country[1]," ranked ",bdLifeExpRank$lifeExpRank,
       " in Asia and ", bdLifeExpWorldRank$lifeExpRank,
       " globally with life expectancy ", bdLifeExpRank$lifeExp," years in 2007")
```

```
## [1] "Bangladesh ranked 27 in Asia and 93 globally with life expectancy 64.062 years in 2007"
```

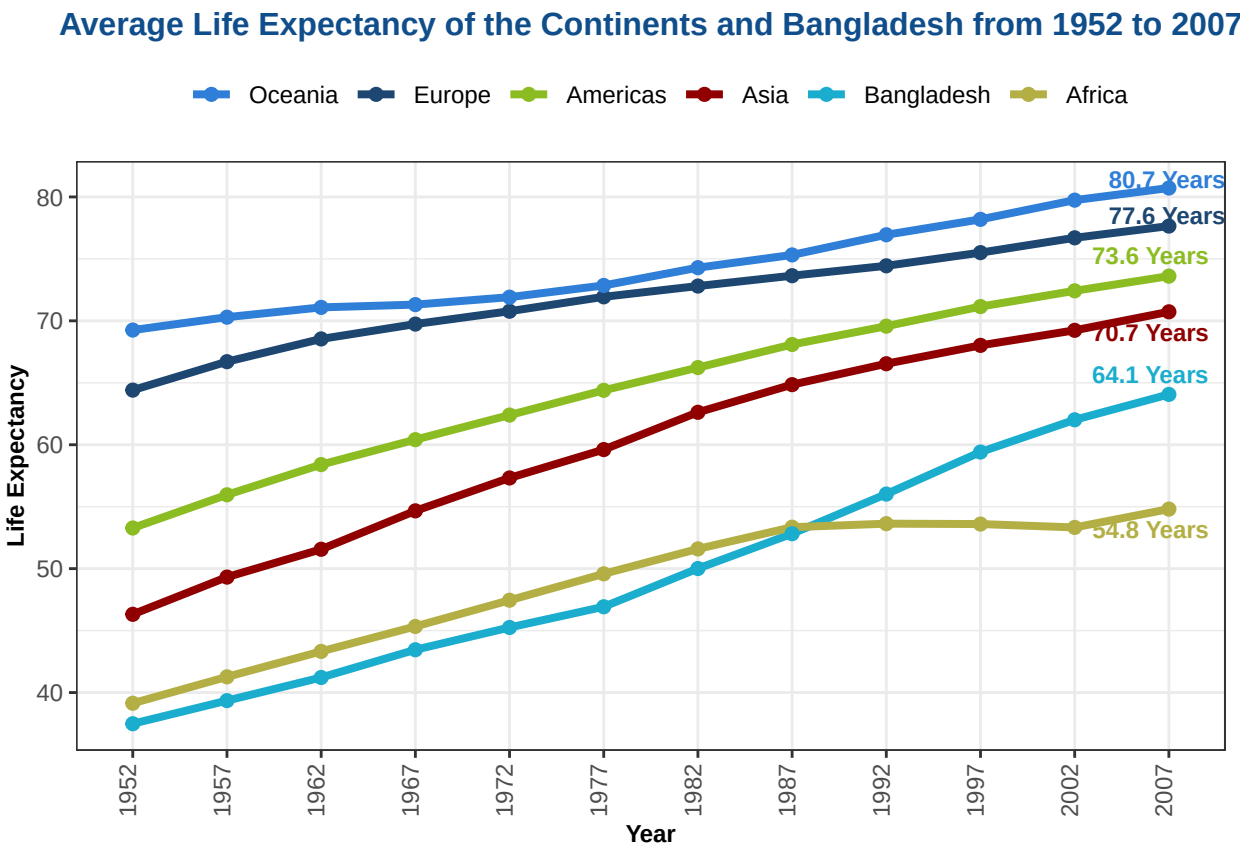
So, Bangladesh ranked 27 in Asia and 93 globally in 2007. But, in 2018, according to World Bank Bangladesh ranked 97 with life expectancy 72.32 years. Similarly, we know **Swaziland* has the lowest life expectancy 39.61 years in the world!

EDA: Life Expectancy at Birth

Now, let's draw some picture with more useful insights about life expectancy. We want to know average life expectancy of the continents since 1952 to 2007 and where our country belongs! In this report, I'll add only the data manipulation code. Code for graphics will be avoided to increase readability. Full codes will be available on the .rmd file.

```
# remove unnecessary file
rm(bdLifeExpRank, bdLifeExpWorldRank)

continentLifeExp <- gapminder %>%
  group_by(continent, year) %>%
  summarise(lifeExp = mean(lifeExp)) %>%
  ungroup() %>%
  # factor variable might cause problem, so make it as character variable
  mutate(region = as.character(continent)) %>%
  select(region, year, lifeExp)

#sepearte Bangladesh and then append with continents life expectancy data to see
# the position of our country globally
bdLifeExpYearly <- gapminder %>%
  filter(country == "Bangladesh") %>%
  select(country, year, lifeExp) %>%
  rename(region = country)

combined_Bd_Continent <- bind_rows(continentLifeExp, bdLifeExpYearly)
rm(continentLifeExp, bdLifeExpYearly)

# get the order of the continents and bangladesh to make visualisation more aesthetic
regionRank <- combined_Bd_Continent %>%
  group_by(region) %>%
  summarise(lifeExp = mean(lifeExp)) %>%
  ungroup() %>%
  arrange(desc(lifeExp))
orderedRegion <- regionRank$region
rm(regionRank)

combined_Bd_Continent <- combined_Bd_Continent %>%
  mutate(region = factor(region, levels = orderedRegion, ordered = TRUE))
```

```
lastLifeExp <- combined_Bd_Continent %>%
  group_by(region) %>%
  arrange(desc(year)) %>%
  slice(1) %>%
  arrange(desc(lifeExp))
```



Here,Legends are ordered from highest to lowest life expectancy at birth. Continent Oceania has highest life expectancy and Africa has the lowest throughout this time interval. Though Bangladesh has life expectancy lower than average in Asia, It shows a high increasing trend since 1977.

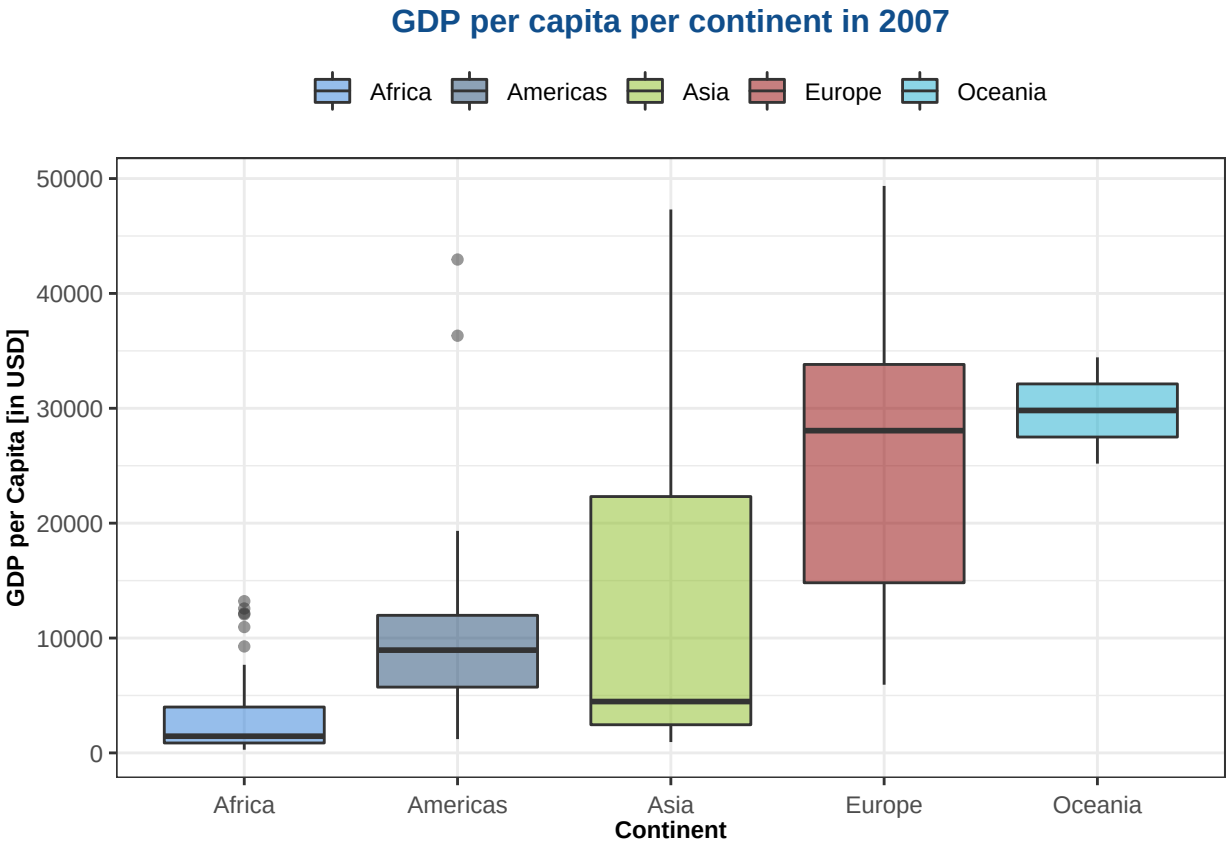
EDA: GDP per Capita

What about GDP, Let's explore some interesting facts with 1-D plot/density plot and line plot. Here is a box-plot of GDP per continents from 1952 to 2007.

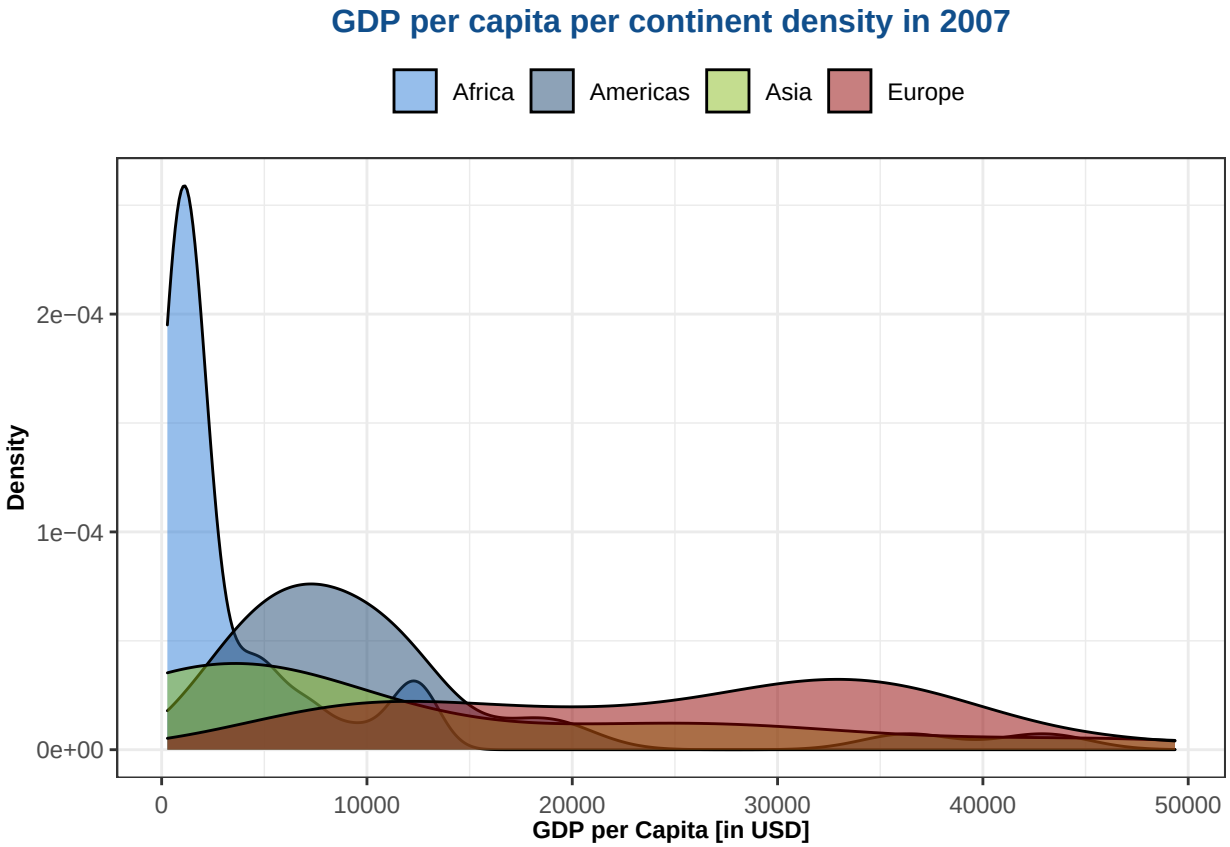
```
continentGdp <- gapminder %>%
  group_by(continent, year) %>%
  summarise(gdpPercap = mean(gdpPercap)) %>%
  ungroup() %>%
  # factor variable might cause problem, so make it as character variable
  mutate(region = as.character(continent)) %>%
  select(region, year, gdpPercap)

#sepearte Bangladesh and then append with continents life expectancy data to see
# the position of our country globally
```

```r
bdGdpYearly <- gapminder %>%
  filter(country == "Bangladesh") %>%
  select(country, year, gdpPercap) %>%
  rename(region = country)

gdp_combined_Bd_Continent <- bind_rows(continentGdp, bdGdpYearly) %>%
  mutate(gdpPercap = round(gdpPercap,0))
rm(bdGdpYearly, continentGdp)

# get the order of the continents and bangladesh to make visualisation more aesthetic
regionRankGdp <- gdp_combined_Bd_Continent %>%
  group_by(region) %>%
  summarise(gdpPercap = round(mean(gdpPercap), 0)) %>%
  ungroup() %>%
  arrange(desc(gdpPercap))
orderedRegion <- regionRankGdp$region
rm(regionRankGdp)

gdp_combined_Bd_Continent <- gdp_combined_Bd_Continent %>%
  mutate(region = factor(region, levels = orderedRegion, ordered = TRUE))

lastGdp<- gdp_combined_Bd_Continent %>%
  group_by(region) %>%
  arrange(desc(year)) %>%
  slice(1) %>%
  arrange(desc(gdpPercap))
```

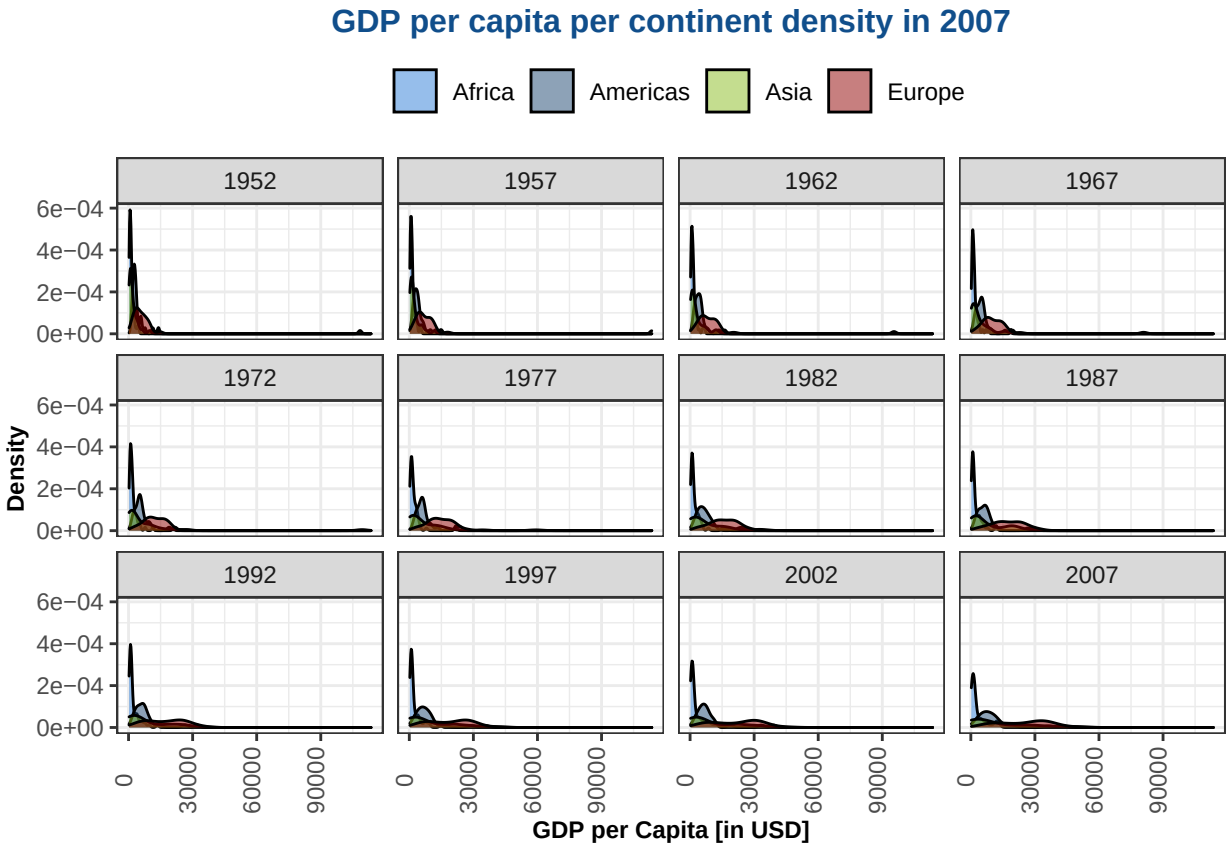
```r
col_code <- c("#2f7ed8","#1d4670", "#8bbc21", "#910000", "#1aadce", "#b3af44")

# library(ggrepel)
gap_box_plot <- gapminder %>%
  filter(year == 2007) %>%
  ggplot(aes(x = continent, y = gdpPercap, fill = continent)) +
  geom_boxplot(alpha = 0.5) +
  labs(x = 'Continent', y = 'GDP per Capita [in USD]',
       title = "GDP per capita per continent in 2007") +
  theme_bw()+
  theme(
        plot.title = element_text(size= 12, face='bold', color = '#104E8B',
                                  hjust = 0.5),
        axis.title = element_text(size= 9, face="bold"),
        axis.text.x = element_text(angle = 0, hjust = 0.5, vjust = .2),
        legend.title = element_blank(), legend.position = "top") +
  guides(color = guide_legend(nrow = 1)) +
  scale_fill_manual(values = col_code)

print(gap_box_plot)
```



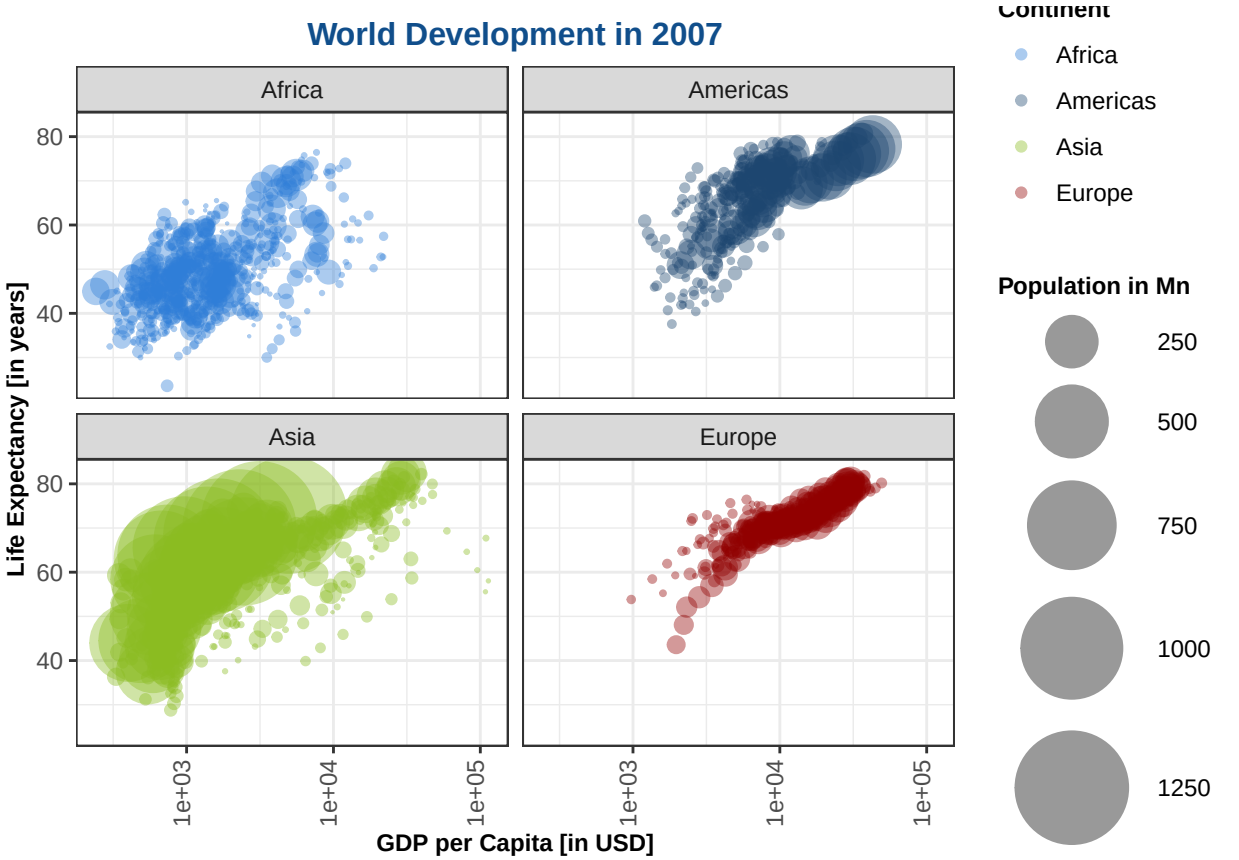
This plot suggests, African countries has the lowest GDP compared to the other continents. Continent Asia shows a right-tail distribution indicating that around 50% countries have GDP lower than 5000. It also tells a story of income inequality among Asian countries. Oceania has highest GDP per capita but since it only contain two countries, we'll remove it from further EDA. The lower 50% country of Europe has higher GDP than Americas except two outlier. Now what about density of GDP for each continents in 2007



Density plot show high peak for Africa with lower GDP. Asia clearly shows heavy right tail indicating how widely scattered these countries GDP. Europe shows a comparatively flat distribution of higher GDP's. Now let's see how GDP distribution for each continents changes over time



Another useful visualization is a bubble chart. Here, we use four variables at a time for visualization, *population size* presented by bubble radius, GDP in y-axis and life expectancy in x-axis. Circles in right upper plot area represent countries with higher GDP and higher life expectancy for each continents and a large circle indicates bigger population.



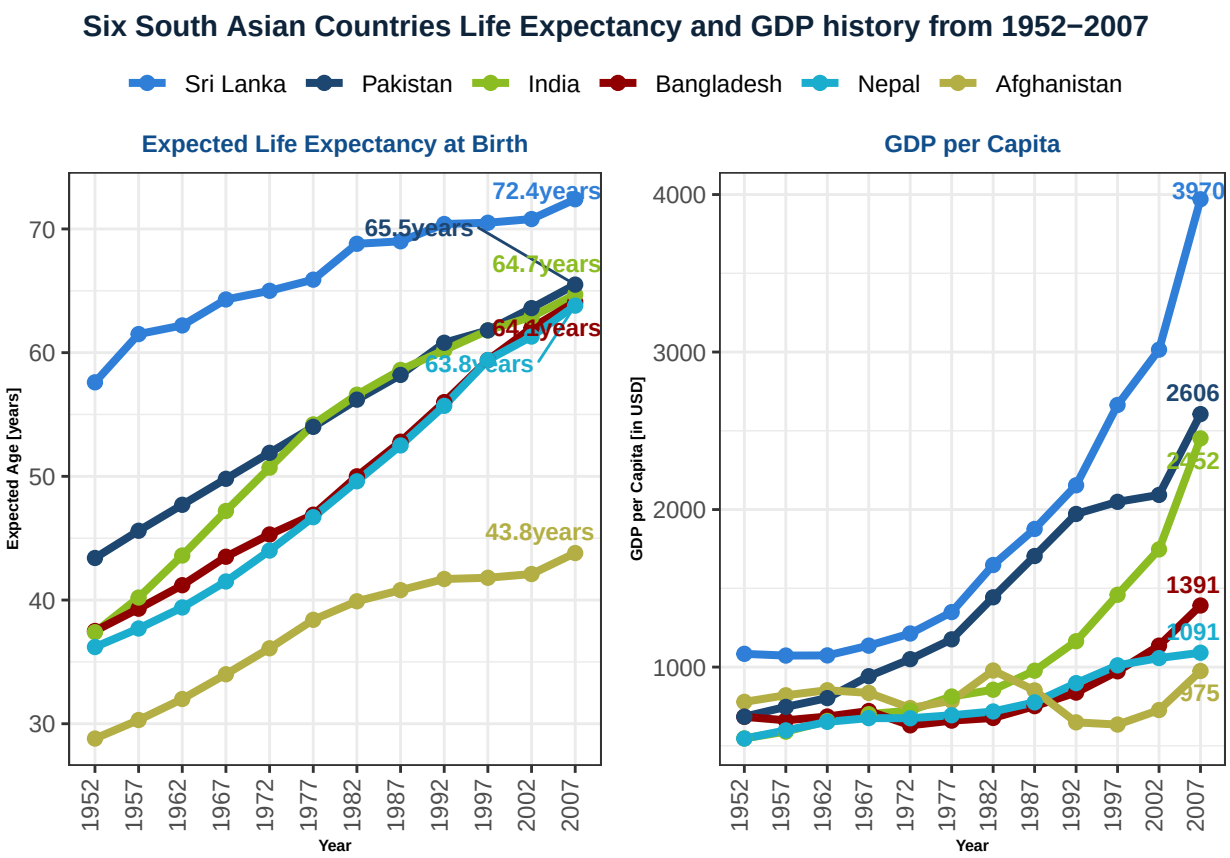
We can see, in Asia most of the countries have lower GDP with medium life expectancy but bigger population. Africa shows lower value in both GDP and life expectancy for most of the countries. Europe has country wise less population but high GDP and high life expectancy. Americas shows higher values even for countries with bigger population.

EDA: South Asian Countries

Now, let's focus on our neighboring countries. We are interested to know how we stands among our neighbors. Note that, Bhutan and Maldives are missing from the *gapminder* data set. First, we need to do data wrangling to separate those countries data that we are interested, then we do some insightful visualization.

```
# life expectancy

sacLe <- gapminder %>%
  filter(country %in% c("Bangladesh", "Afghanistan", "India",
                        "Pakistan", "Sri Lanka", "Nepal")) %>%
  # factor variable might cause problem, so make it as character variable
  mutate(country = as.character(country),
         gdpPercap = round(gdpPercap, 0)) %>%
  select(country, year, lifeExp)


# get the order of the continents and bangladesh to make visualisation more aesthetic
sacRankLe <- sacLe %>%
  group_by(country) %>%
  summarise(lifeExp = round(mean(lifeExp), 1)) %>%
  ungroup() %>%
```

```
  arrange(desc(lifeExp))
orderedcountryLe <- sacRankLe$country
rm(sacRankLe)

sacLe <- sacLe %>%
  mutate(country = factor(country, levels = orderedcountryLe, ordered = TRUE),
         lifeExp = round(lifeExp, 1))

# creating value label in year 2007
sacLastLe<- sacLe %>%
  group_by(country) %>%
  arrange(desc(year)) %>%
  slice(1) %>%
  arrange(desc(lifeExp))

# GDP calculation
sacGdp <- gapminder %>%
  filter(country %in% c("Bangladesh", "Afghanistan", "India",
                        "Pakistan", "Sri Lanka", "Nepal")) %>%
  # factor variable might cause problem, so make it as character variable
  mutate(country = as.character(country),
         gdpPercap = round(gdpPercap, 0)) %>%
  select(country, year, gdpPercap)

# get the order of the countries to make visualization more aesthetic
sacRankGdp <- sacGdp %>%
  group_by(country) %>%
  summarise(gdpPercap = round(mean(gdpPercap), 0)) %>%
  ungroup() %>%
  arrange(desc(gdpPercap))
orderedCountryGdp <- sacRankGdp$country
rm(sacRankGdp)

sacGdp <- sacGdp %>%
  mutate(country = factor(country, levels = orderedcountryLe, ordered = TRUE))

sacLastGdp<- sacGdp %>%
  group_by(country) %>%
  arrange(desc(year)) %>%
  slice(1) %>%
  arrange(desc(gdpPercap))
```



In 2007, Sri Lanka has both high GDP and high life expectancy among our neighbor countries. Bangladesh was at fourth place in both parameters and Afghanistan cames last. Besides Sri Lanka and Afghanistan, other four has similar life expectancy in 2007 but shows a higher difference in GDP.

Data Modelling

Now we perform some basic modeling using linear and second order polynomial regression. Model adequacy checking are beyond the scope of this report. We want to predict the life expectancy of Bangladesh in 2011 and 2021

```
gapminder_bd <- gapminder %>%
  filter(country == "Bangladesh")

gpAfter70 <- gapminder %>%
  filter(country == "Bangladesh", year>=1970)

#ggplot(data = gapminder_bd, aes(x = year, y = gdpPercap)) + geom_point()
#ggplot(data = gpAfter70, aes(x = year, y = gdpPercap)) + geom_point()

# life expextency modelling
bd_lifeExp_model <- lm(lifeExp~year, data = gapminder_bd)
bd_lifeExp_model_squared <- lm(lifeExp~year + I(year^2), data = gapminder_bd)
summary(bd_lifeExp_model)
```

```
##
## Call:
## lm(formula = lifeExp ~ year, data = gapminder_bd)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.66576 -0.77482  0.02824  0.75655  1.34851
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -936.21577   32.33636  -28.95 5.63e-11 ***
## year           0.49813    0.01633   30.50 3.37e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9767 on 10 degrees of freedom
## Multiple R-squared:  0.9894, Adjusted R-squared:  0.9883
## F-statistic: 929.9 on 1 and 10 DF,  p-value: 3.37e-11
```

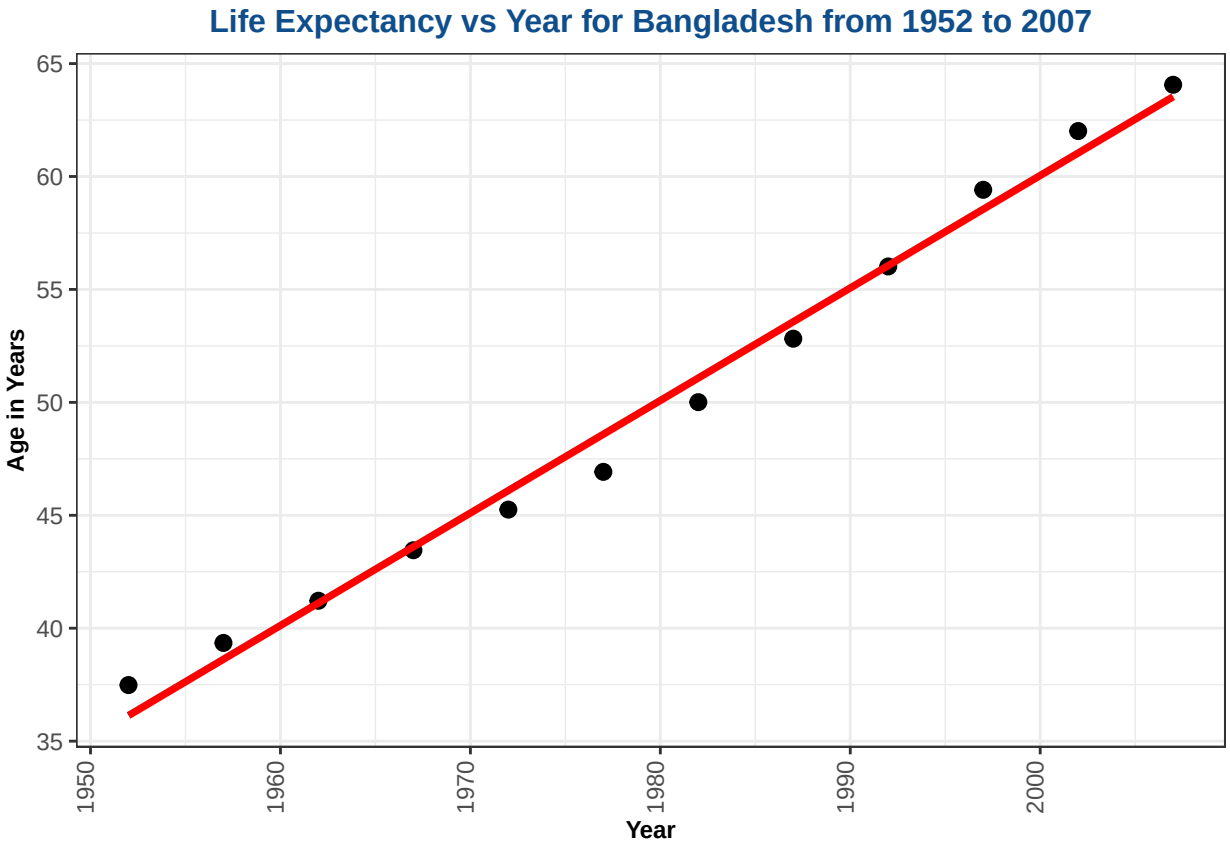
```
summary(bd_lifeExp_model_squared)
```

```
##
## Call:
## lm(formula = lifeExp ~ year + I(year^2), data = gapminder_bd)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81495 -0.16076  0.04321  0.28491  0.83632
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.049e+04  2.234e+03   4.696 0.001126 **
## year        -1.105e+01  2.258e+00  -4.895 0.000854 ***
## I(year^2)    2.917e-03  5.702e-04   5.116 0.000632 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5208 on 9 degrees of freedom
## Multiple R-squared:  0.9973, Adjusted R-squared:  0.9967
## F-statistic:  1648 on 2 and 9 DF,  p-value: 2.867e-12
```

```
predict(bd_lifeExp_model, newdata = data.frame(year = c(2011, 2021)))
```

```
##        1        2
## 65.52520 70.50651
```

As we can see, life expectancy prediction in 2011 is 65.52 years and 74.66 years in 2021. Both are very close to the actual values. Life expectancy with prediction line are given below. Both linear and polynomial model shows high r-squared value means both model captures most of the variability of the predictors. Though, polynomial gives slightly better result but we choose linear model because of its simplicity and interpretablity.



Now, let's try to model GDP per capita in Bangladesh. Again, complex models are beyond the scope of this report,so we use linear and first order polynomial regression to predict GDP per Capita in 2011 and 2021 in Bangladesh. We use a trick here, we first model using all data and then we use data since 1970 because after liberation war, there is a high possibility that more stability will be gain

```
# gdp modelling
bd_gdp_model <- lm(gdpPercap~year, data = gapminder_bd)
bd_gdp_model_squared <- lm(gdpPercap~year + I(year^2), data = gapminder_bd)
# summary(bd_gdp_model)
summary(bd_gdp_model_squared)
```

```
##
## Call:
## lm(formula = gdpPercap ~ year + I(year^2), data = gapminder_bd)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -61.43 -30.61 -17.16  17.15 101.99
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.835e+06  2.330e+05   7.876 2.51e-05 ***
## year        -1.864e+03  2.354e+02  -7.917 2.40e-05 ***
## I(year^2)    4.734e-01  5.945e-02   7.962 2.30e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 54.3 on 9 degrees of freedom
## Multiple R-squared:  0.9563, Adjusted R-squared:  0.9466
## F-statistic: 98.58 on 2 and 9 DF,  p-value: 7.589e-07
```

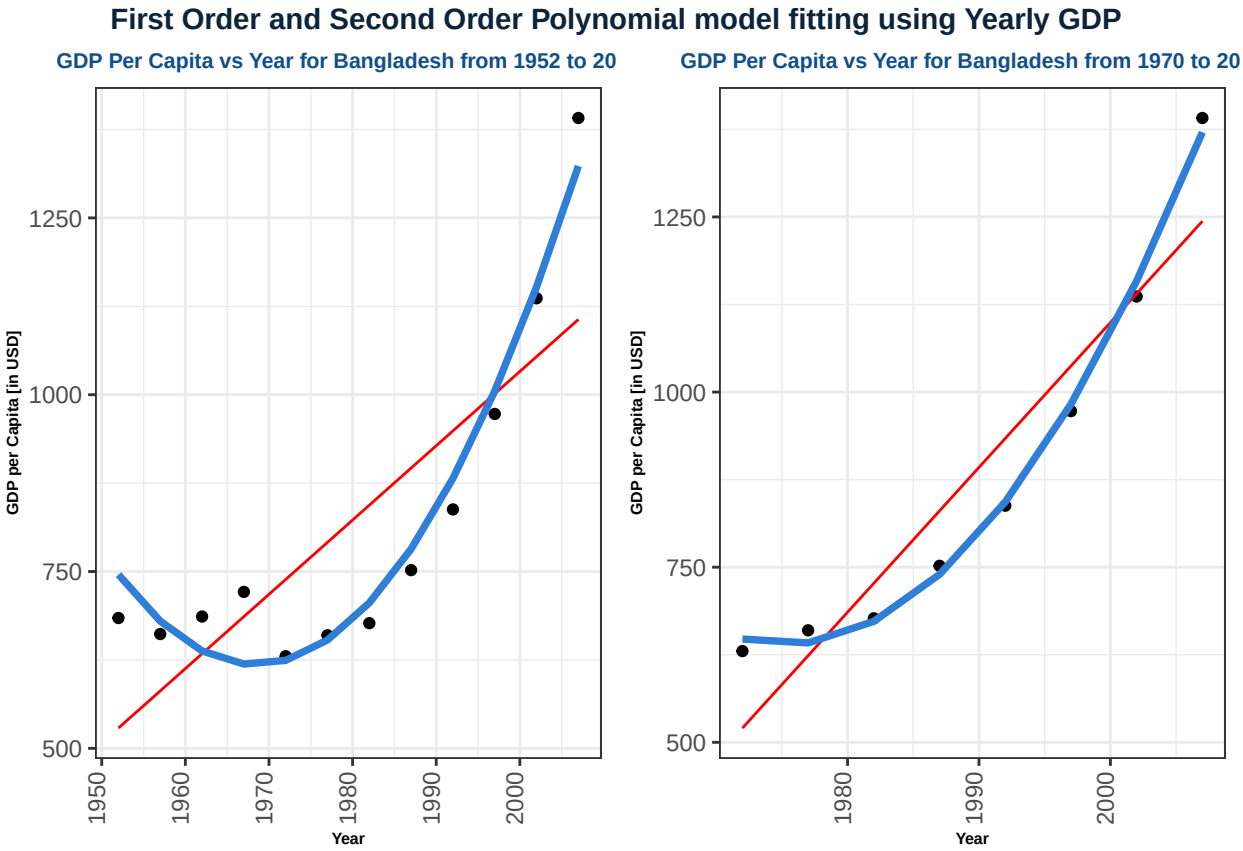
```
## after 1970, gdp modelling
bd_gdp_model_70 <- lm(gdpPercap~year, data = gpAfter70)
bd_gdp_model_squared_70 <- lm(gdpPercap~year + I(year^2), data = gpAfter70)
# summary(bd_gdp_model_70)
summary(bd_gdp_model_squared_70)
```

```
##
## Call:
## lm(formula = gdpPercap ~ year + I(year^2), data = gpAfter70)
##
## Residuals:
##       1       2       3       4       5       6       7       8
## -17.161  17.989   4.309  12.230  -5.307 -10.007 -22.337  20.284
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.833e+06  2.334e+05   12.14 6.71e-05 ***
## year        -2.867e+03  2.346e+02  -12.22 6.48e-05 ***
## I(year^2)    7.258e-01  5.896e-02   12.31 6.26e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.1 on 5 degrees of freedom
## Multiple R-squared:  0.9964, Adjusted R-squared:  0.9949
## F-statistic: 690.6 on 2 and 5 DF,  p-value: 7.815e-07
```

```
predict(bd_gdp_model_squared_70, newdata = data.frame(year = c(2011, 2021)))
```

```
##        1        2
## 1566.893 2158.317
```

We fit linear and polynomial for both data sets(one is 1952-2007 and other is 1970-2007). Polynomial model with second data sets(from year 1970) explains best and predicts $1567 GDP per capita in 2011 and $2158 in 2021. Both models ar plotted below with linear and polynomial regression line



In this report, we've introduced the idea of doing data manipulation using *tidyverse*, and talked about exploratory data analysis. We also try to predict life expectancy and GDP per Capita using linear and polynomial regression

Reference List

1. gapminder: Data from Gapminder
2. R for Data Science
3. ggplot2: Elegant Graphics for Data Analysis