Shannon Assignment 1

Aman kumar

07/02/2021

Goal: Using R explore the following data set

children_per_woman_total_fertility : It includes the data of average children per woman for 194 different countries per year from the year 1800 to 2100.

child_mortality_0_5_year_olds_dying_per_1000_born : It includes the data of child mortality(children under 5 year's age dying per 1000) for 183 different countries per year from the year 1800 to 2100.

income_per_person_gdppercapita_ppp_inflation_adjusted : It includes the data of GDP per capita for 192 different countries per year from the year 1800 to 2100.

life_expectancy_years : It includes the data of Life expectancy for 186 different countries per year from the year 1800 to 2100.

population_total : It includes the data of Total Population for 194 different countries per year from the year 1800 to 2100.

Hypothesis

we want to compare **INDIA** with mean for top 10 developed countries , 10 developing counties and 10 under developed counties

we explore each dataset individually. We consider ten countries each for three different sets of Countries (Developed, Developing and Under Developed), for each set of countries we take the mean of sample values for each year.We have counsidered mean of 10 countries because it will help us make a more accurate estimate of what trend a country from a set is likely to follow. Later we plot these means for Developed, Developing and Under Developed countries and compare their trends. We also make certain Hypothesis regarding the data we observe and try to prove or reject those Hypothesis with help of various Hypothesis testing methods learnt in this course so far. Countries that are considered are :

Developed Countries : Norway , Ireland , Switzerland , Hong Kong , Iceland , Germany , Sweden , Australia, Netherlands , Denmark

Developing Countries : Algeria , Lebanon , Fiji , Moldova , Maldives , Tunisia , Saint Vincent and the Grenadines , Suriname , Mongolia , Botswana

Underdeveloped Countries : Eritrea , Mozambique , Burkina Faso , Sierra Leone , Mali , Burundi , South Sudan , Chad , Central African Republic , Niger

```
Note: All the list of countries has been taken from **HDR website** http://www.hdr.undp.org/
```

Importing all datasets

```
read.csv("child_mortality_0_5_year_olds_dying_per_1000_born.csv", header = T, check.names = F) -
> ChildMortality

read.csv("children_per_woman_total_fertility.csv", header = T, check.names = F) -> ChildrenPerWo
men

read.csv("income_per_person_gdppercapita_ppp_inflation_adjusted.csv", header = T, check.names =
F)-> IncomePerPerson

read.csv("life_expectancy_years.csv", header = T, check.names = F)-> LifeExpectancy

read.csv("population_total.csv", header = T, check.names = F)-> Population
```

Loading the libraries

```
library(tidyverse)
library(dplyr)
library(tidyr)
library(Hmisc)
```

Creating generalised functions

we are creating function to extract counties data information

suppose a dummy dataset is following

```
Dataset <- IncomePerPerson
a <- "country 1"
b <- "country 2"
c <- "country 3"
d <- "country 4"
e <- "country 5"
f <- "country 6"
g <- "country 7"
h <- "country 8"
i <- "country 9"
j <- "country 10"
```

Listcollecter extracts our country of interest from the specified dataset and stores in a table.

```
Listcollecter <- function(Dataset, a, b, c, d, e, f, g, h, i, j) {Dataset[Dataset$country %in% c
(a, b, c, d, e, f, g, h, i, j), ]}
```

Extracting Specific countries from the list

```
countrydata<- function(countryname, Dataset)
{as.vector(na.omit(as.numeric(unlist(Dataset[Dataset$country==countryname,2:227]))))}

aicountrydata<- function(countryname, Dataset)
{as.vector(na.omit(as.numeric(unlist(Dataset[Dataset$country==countryname,149:227]))))}

bicountrydata<- function(countryname, Dataset)
{as.vector(na.omit(as.numeric(unlist(Dataset[Dataset$country==countryname,2:149]))))}
```

Extracting top ten countries

```
 a <- "Norway"
 b <- "Ireland"
 c <- "Switzerland"
 d <- "Hong Kong"
 e <- "Iceland"
 f <- "Germany"
 g <- "Sweden"
 h <- "Australia"
 i <- "Netherlands"
 j <- "Denmark"
```

Top ten countries life expectancies

```
Dataset <- LifeExpectancy
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Toptenlife
```

Top 10 countries child mortality rate

```
Dataset <- ChildMortality
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Toptenchildmortality
```

Top 10 countries Child per women

```
Dataset <- ChildrenPerWomen
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Toptenchildrenperwomen
```

Top 10 countries income per person

```
Dataset <- IncomePerPerson
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Toptenincomeperperson
```

Top 10 countries Population

```
Dataset <- Population
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Toptenpopulation
```

Extracting Middle Ten Countries

```
 a <- "Algeria"
 b <- "Lebanon"
 c <- "Fiji"
 d <- "Moldova"
 e <- "Maldives"
 f <- "Tunisia"
 g <- "Saint Vincent and the Grenadines"
 h <- "Suriname"
 i <- "Mongolia"
 j <- "Botswana"
```

Middle 10 countries Life expectancy

```
Dataset <- LifeExpectancy
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Middletenlife
```

Middle 10 countries child mortality rate

```
Dataset <- ChildMortality
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Middletenchildmortality
```

Middle 10 countries Child per women

```
Dataset <- ChildrenPerWomen
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Middletenchildrenperwomen
```

Middle 10 countries income per person

```
Dataset <- IncomePerPerson
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Middletenincomeperperson
```

Middle 10 countries Population

```
Dataset <- Population
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Middletenpopulation
```

Bottom 10 countries

```
a <- "Eritrea"
b <- "Mozambique"
c <- "Burkina Faso"
d <- "Sierra Leone"
e <- "Mali"
f <- "Burundi"
g <- "South Sudan"
h <- "Chad"
i <- "Central African Republic"
j <- "Niger"
```

Bottom10 countries Life expectancy

```
Dataset <- LifeExpectancy
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Bottomtenlife
```

Bottom 10 countries child mortality rate

```
Dataset <- ChildMortality
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Bottomtenchildmortality
```

Bottom 10 countries Child per women

```
Dataset <- ChildrenPerWomen
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Bottomtenchildrenperwomen
```

Bottom 10 countries income per person

```
Dataset <- IncomePerPerson
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Bottomtenincomeperperson
```

Bottom 10 countries Population

```
Dataset <- Population
Listcollecter(Dataset, a, b, c, d, e, f, g, h, i, j) -> Bottomtenpopulation
```

Merging files

Calculating means

Means of Life expectancies

```
apply(Toptenlife[,2:227], 2, mean) -> topmeanlife
apply(Middletenlife[,2:227], 2, mean) -> middlemeanlife
apply(Bottomtenlife[,2:227], 2, mean) -> bottommeanlife
apply(Toptenlife[,149:227], 2, mean) -> aitopmeanlife
apply(Middletenlife[,149:227], 2, mean) -> aimiddlemeanlife
apply(Bottomtenlife[,149:227], 2, mean) -> aibottommeanlife
apply(Toptenlife[,2:149], 2, mean) -> bitopmeanlife
apply(Middletenlife[,2:149], 2, mean) -> bimiddlemeanlife
apply(Bottomtenlife[,2:149], 2, mean) -> bibottommeanlife
```

Means of Child Mortality

```
apply(Toptenchildmortality[,2:227], 2, mean) -> topmeanchildmortality
apply(Middletenchildmortality[,2:227], 2, mean) -> middlemeanchildmortality
apply(Bottomtenchildmortality[,2:227], 2, mean) -> bottommeanchildmortality
apply(Toptenchildmortality[,149:227], 2, mean) -> aitopmeanchildmortality
apply(Middletenchildmortality[,149:227], 2, mean) -> aimiddlemeanchildmortality
apply(Bottomtenchildmortality[,149:227], 2, mean) -> aibottommeanchildmortality
apply(Toptenchildmortality[,2:149], 2, mean) -> bitopmeanchildmortality
apply(Middletenchildmortality[,2:149], 2, mean) -> bimiddlemeanchildmortality
apply(Bottomtenchildmortality[,2:149], 2, mean) -> bibottommeanchildmortality
```

Means of Children per women

```
apply(Toptenchildrenperwomen[,2:227], 2, mean) -> topmeanchildrenperwomen
apply(Middletenchildrenperwomen[,2:227], 2, mean) -> middlemeanchildrenperwomen
apply(Bottomtenchildrenperwomen[,2:227], 2, mean) -> bottommeanchildrenperwomen
apply(Toptenchildrenperwomen[,149:227], 2, mean) -> aitopmeanchildrenperwomen
apply(Middletenchildrenperwomen[,149:227], 2, mean) -> aimiddlemeanchildrenperwomen
apply(Bottomtenchildrenperwomen[,149:227], 2, mean) -> aibottommeanchildrenperwomen
apply(Toptenchildrenperwomen[,2:149], 2, mean) -> bitopmeanchildrenperwomen
apply(Middletenchildrenperwomen[,2:149], 2, mean) -> bimiddlemeanchildrenperwomen
apply(Bottomtenchildrenperwomen[,2:149], 2, mean) -> bibottommeanchildrenperwomen
```

Means of Income Per person

```
apply(Toptenincomeperperson[,2:227], 2, mean) -> topmeangdp
apply(Middletenincomeperperson[,2:227], 2, mean) -> middlemeangdp
apply(Bottomtenincomeperperson[,2:227], 2, mean) -> bottommeangdp
apply(Toptenincomeperperson[,149:227], 2, mean) -> aitopmeangdp
apply(Middletenincomeperperson[,149:227], 2, mean) -> aimiddlemeangdp
apply(Bottomtenincomeperperson[,149:227], 2, mean) -> aibottommeangdp
apply(Toptenincomeperperson[,2:149], 2, mean) -> bitopmeangdp
apply(Middletenincomeperperson[,2:149], 2, mean) -> bimiddlemeangdp
apply(Bottomtenincomeperperson[,2:149], 2, mean) -> bibottommeangdp
```

Means of Population

```
apply(Toptenpopulation[,2:227], 2, mean) -> topmeanpopulation
apply(Middletenpopulation[,2:227], 2, mean) -> middlemeanpopulation
apply(Bottomtenpopulation[,2:227], 2, mean) -> bottommeanpopulation
apply(Toptenpopulation[,149:227], 2, mean) -> aitopmeanpopulation
apply(Middletenpopulation[,149:227], 2, mean) -> aimiddlemeanpopulation
apply(Bottomtenpopulation[,149:227], 2, mean) -> aibottommeanpopulation
apply(Toptenpopulation[,2:149], 2, mean) -> bitopmeanpopulation
apply(Middletenpopulation[,2:149], 2, mean) -> bimiddlemeanpopulation
apply(Bottomtenpopulation[,2:149], 2, mean) -> bibottommeanpopulation
```

Country datasets

Overall dataset for India

```
as.vector(countrydata("India", LifeExpectancy))-> indialife
as.vector(countrydata("India", IncomePerPerson))-> indiagdp
as.vector(countrydata("India", ChildMortality))-> indiachildmortality
as.vector(countrydata("India", ChildrenPerWomen))-> indiachildperwomen
as.vector(countrydata("India", Population))-> indiapopulation
```

India dataset after independence

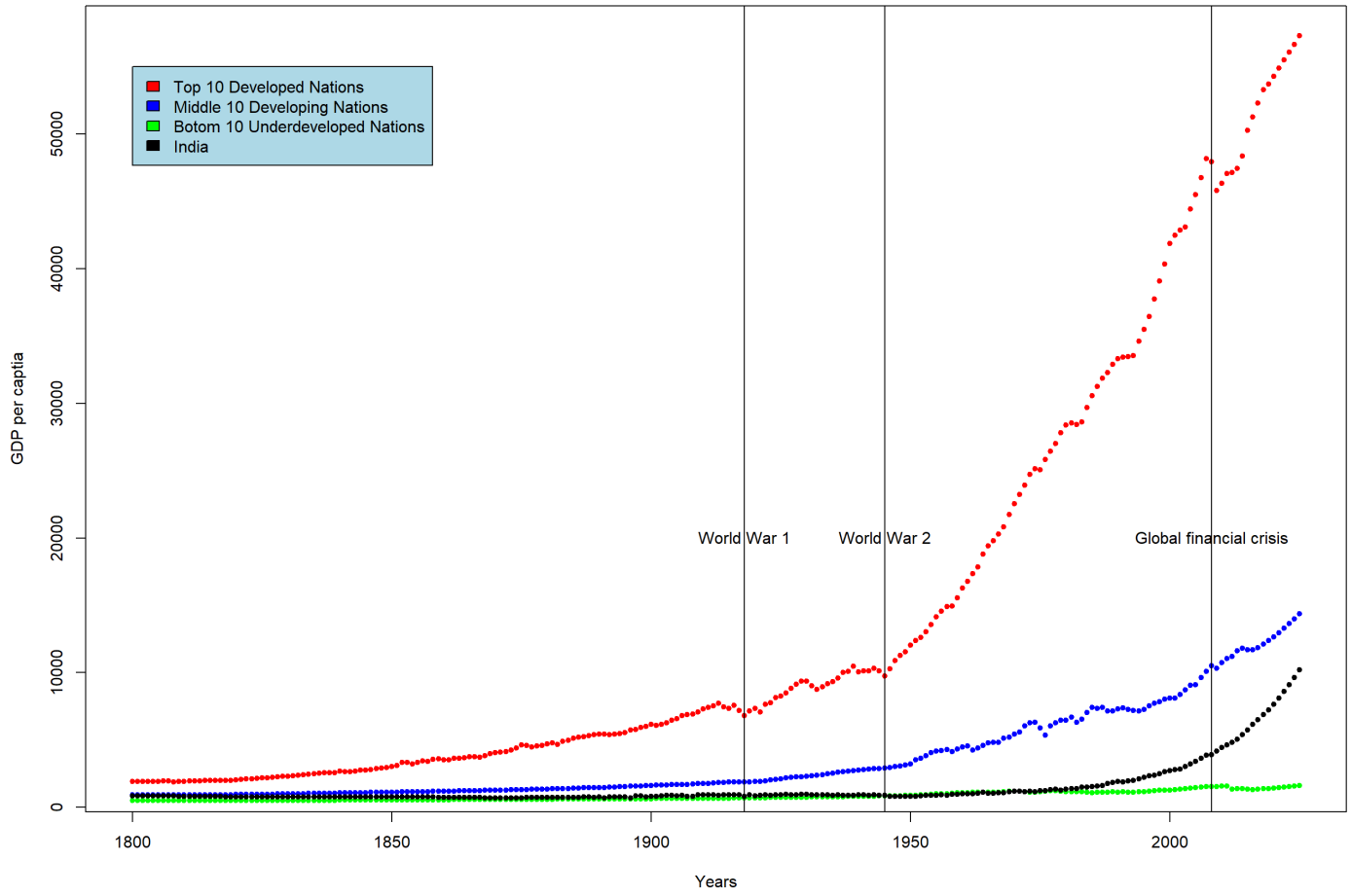
```
as.vector(aicountrydata("India", LifeExpectancy))-> aiindialife
as.vector(aicountrydata("India", IncomePerPerson))-> aiindiagdp
as.vector(aicountrydata("India", ChildMortality))-> aiindiachildmortality
as.vector(aicountrydata("India", ChildrenPerWomen))-> aiindiachildperwomen
as.vector(aicountrydata("India", Population))-> aiindiapopulation
```

India dataset before independence

```
as.vector(bicountrydata("India", LifeExpectancy))-> biindialife
as.vector(bicountrydata("India", IncomePerPerson))-> biindiagdp
as.vector(bicountrydata("India", ChildMortality))-> biindiachildmortality
as.vector(bicountrydata("India", ChildrenPerWomen))-> biindiachildperwomen
as.vector(bicountrydata("India", Population))-> biindiapopulation
```

```
as.vector(unlist(colnames(LifeExpectancy[1,2:227]))) -> Years
as.vector(unlist(colnames(LifeExpectancy[1,149:227]))) -> a47Years
as.vector(unlist(colnames(LifeExpectancy[1,2:149]))) -> b47Years
```

```
plot(Years, topmeangdp, col = "red", pch = 20, ylab = "GDP per captia")
points(Years, middlemeangdp, col= " blue", pch = 20)
points(Years, bottommeangdp, col ="green", pch= 20)
points(Years, indiagdp, col = "black", pch = 20)
legend(1800, 55000, legend=c("Top 10 Developed Nations", "Middle 10 Developing Nations", "Botom
 10 Underdeveloped Nations",  "India"), fill = c("red", "blue", "green", "black"), bg= "lightblu
e")
abline(v= 1918)
abline(v= 1945)
abline(v= 2008)
text(1918, 20075, "World War 1")
text(1945, 20075, "World War 2")
text(2008, 20075, "Global financial crisis")
```



Observations

here we use abline to represent the important information retrive from plot

World war 1 (~1918)

we can see dip in gdp during world war in developed counties more

World war 2 (~1945)

we can see dip in gdp during world war in developed counties more

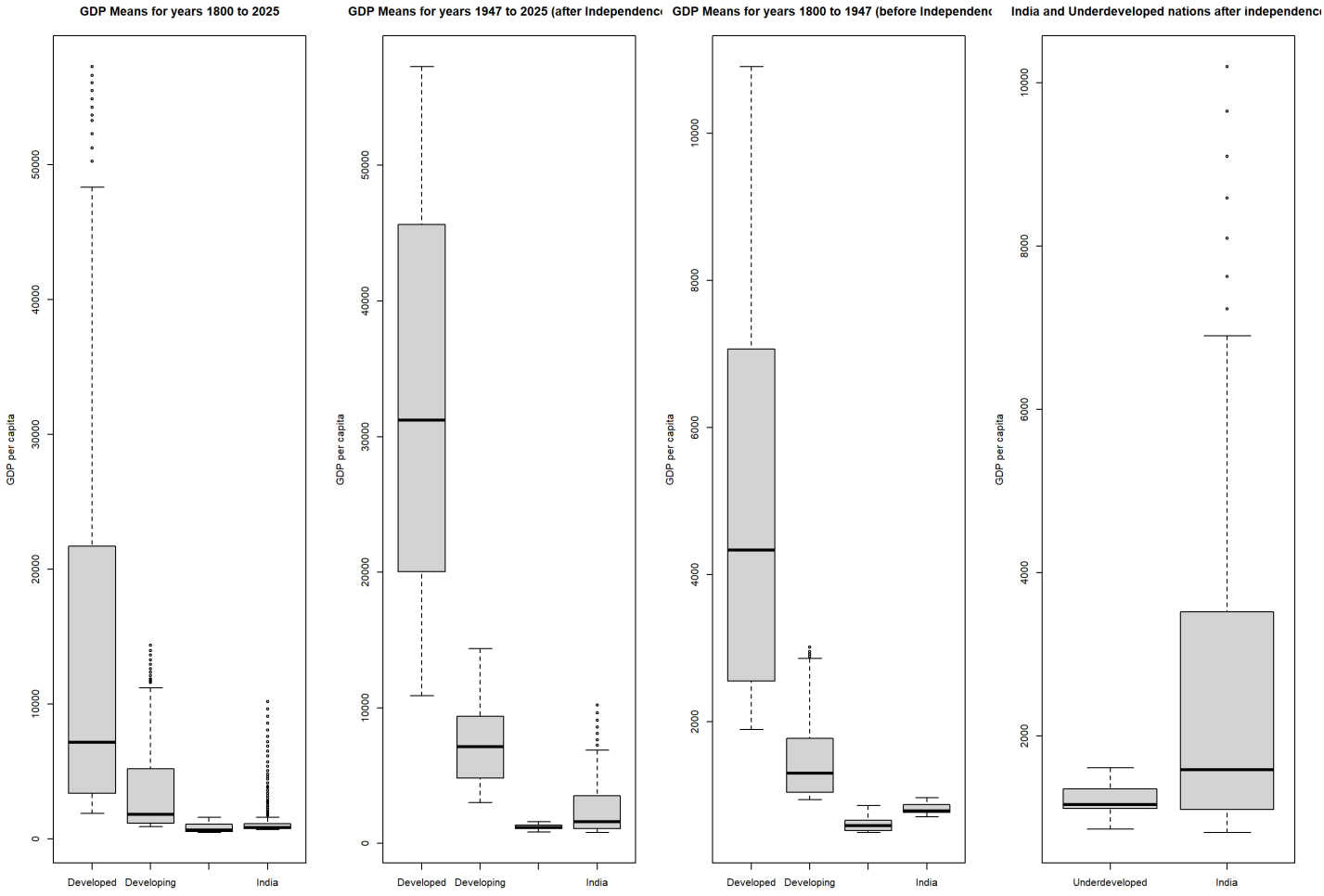
Global financial crisis

india got very minimal impact , by comparison with other developed

Testing Above observations

As India has shown a sharp change after independence it will be injustice to compare overall mean so we'll also check before and after independence data too.

```
par(mfrow=c(1,4))
boxplot(topmeangdp, middlemeangdp, bottommeangdp, indiagdp, names = c("Developed", "Developing",
"Underdeveloped", "India"), ylab= "GDP per capita", main= "GDP Means for years 1800 to 2025")
boxplot(aitopmeangdp, aimiddlemeangdp, aibottommeangdp, aiindiagdp, names = c("Developed", "Deve
loping", "Underdeveloped", "India"), ylab= "GDP per capita", main= "GDP Means for years 1947 to
 2025 (after Independence)")
boxplot(bitopmeangdp, bimiddlemeangdp, bibottommeangdp, biindiagdp, names = c("Developed", "Deve
loping", "Underdeveloped", "India"), ylab= "GDP per capita", main= "GDP Means for years 1800 to
 1947 (before Independence)")
boxplot(aibottommeangdp, aiindiagdp, names = c("Underdeveloped", "India"), ylab= "GDP per capit
a", main= "India and Underdeveloped nations after independence")
```



By the above plots we can say India GDP has increased a lot after independence to check this hypothesis we will do some tests.

Saphiro test to check normality

NULL Hypothesis: The data is normal **Alternate Hypothesis**: The data is not normal

If P> 0.05 our NULL Hypothesis will be true

```
shapiro.test(topmeangdp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  topmeangdp
## W = 0.76516, p-value < 2.2e-16
```

```
shapiro.test(middlemeangdp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  middlemeangdp
## W = 0.76398, p-value < 2.2e-16
```

```
shapiro.test(bottommeangdp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bottommeangdp
## W = 0.85596, p-value = 1.003e-13
```

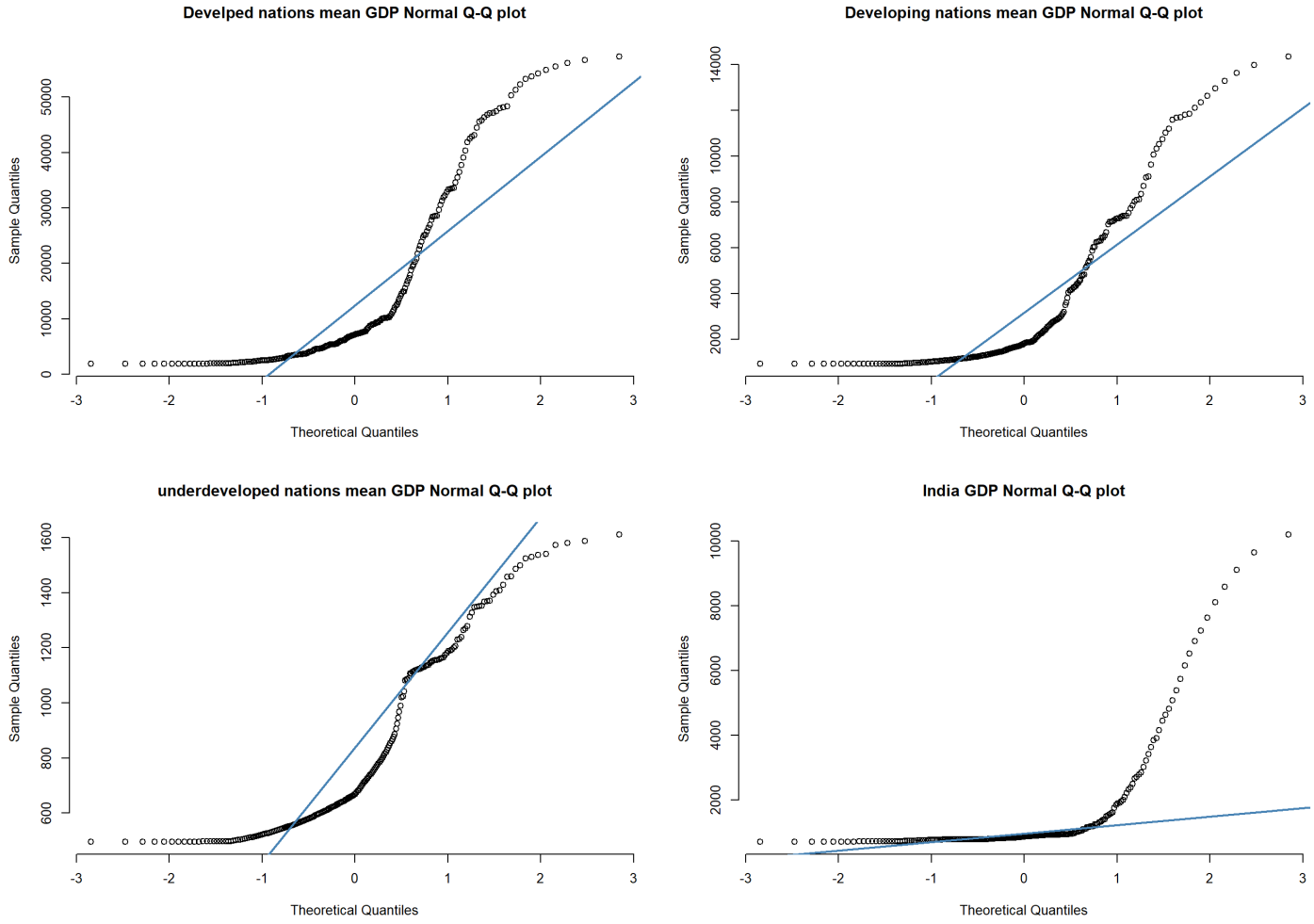
```
shapiro.test(indiagdp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  indiagdp
## W = 0.48676, p-value < 2.2e-16
```

As the P values are less then 0.05 so our data is not normal

To further confirm we can do Q-Q plot

```
par(mfrow=c(2,2) )
qqnorm(topmeangdp, pch = 1, frame = FALSE, main = "Develped nations mean GDP Normal Q-Q plot")
qqline(topmeangdp, col = "steelblue", lwd = 2)
qqnorm(middlemeangdp, pch = 1, frame = FALSE, main = "Developing nations mean GDP Normal Q-Q plo
t")
qqline(middlemeangdp, col = "steelblue", lwd = 2)
qqnorm(bottommeangdp, pch = 1, frame = FALSE, main = "underdeveloped nations mean GDP Normal Q-Q
plot")
qqline(bottommeangdp, col = "steelblue", lwd = 2)
qqnorm(indiagdp, pch = 1, frame = FALSE, main = "India GDP Normal Q-Q plot")
qqline(indiagdp, col = "steelblue", lwd = 2)
```



From the above tests it is clear that the data sets are not normal so we can't use parametric tests like **T test** or **ANOVA** to check weather means are same or not, so we will be using non parametric alternatives of those like **Wilcox** and **Kruskal** tests.

Kruskal test

creating data frame

```
gdpdataframe<- data.frame(values=c(topmeangdp, middlemeangdp, bottommeangdp, indiagdp), variable
= c(rep("topmeangdp", length(topmeangdp)), rep("middlemeangdp", length(middlemeangdp)), rep("bot
tommeangdp", length(bottommeangdp)), rep("indiagdp", length(indiagdp))))
```

```
kruskal.test(gdpdataframe$values~gdpdataframe$variable)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  gdpdataframe$values by gdpdataframe$variable
## Kruskal-Wallis chi-squared = 573.59, df = 3, p-value < 2.2e-16
```

P values is significant so we can say the datas are differfent, so now we will do wilcoxon test for pairwise analysis

Wilcoxon test

NULL Hypotheis: Mean of bottom ten countries is less than India before independence **Alternate Hypotheis**: Mean of bottom ten countries is greater than India before independence

```
wilcox.test(bibottommeangdp, biindiagdp, paired = T, alternative = "greater")
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  bibottommeangdp and biindiagdp
## V = 3, p-value = 1
## alternative hypothesis: true location shift is greater than 0
```

As P value is significant we can say mean life expectancy of india was greater than bottom ten countries before independence

NULL Hypotheis: Mean of bottom ten countries is less than India after independence **Alternate Hypotheis**: Mean of bottom ten countries is greater than India after independence

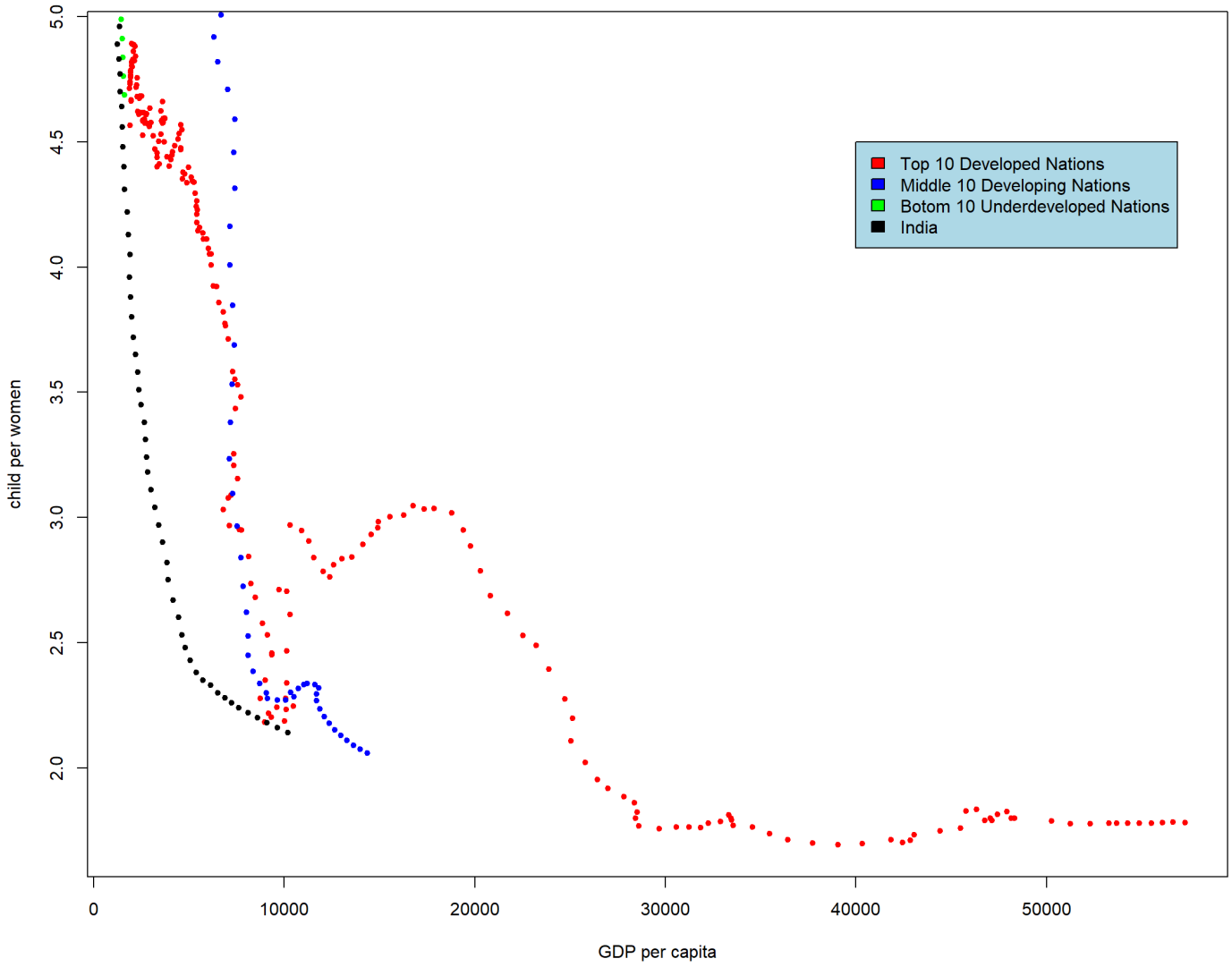
```
wilcox.test(aibottommeangdp, aiindiagdp, paired = T, alternative = "greater")
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  aibottommeangdp and aiindiagdp
## V = 438, p-value = 1
## alternative hypothesis: true location shift is greater than 0
```

As P value is significant we can say mean gdp of India became greater than bottom ten countries before independence

How GDP per capita changed along with child per women

```
plot(topmeangdp, topmeanchildrenperwomen, col = "red", pch = 20, xlab = "GDP per capita", ylab =
"child per women")
points(middlemeangdp, middlemeanchildrenperwomen, col= " blue", pch = 20)
points(bottommeangdp, bottommeanchildrenperwomen, col ="green", pch= 20)
points(indiagdp, indiachildperwomen, col = "black", pch = 20)
legend(40000, 4.5, legend=c("Top 10 Developed Nations", "Middle 10 Developing Nations", "Botom 1
0 Underdeveloped Nations",  "India"), fill = c("red", "blue", "green", "black"), bg= "lightblue"
)
```



Hypthesis With increase in GDP children per women decreases , to check this correlation we can do correlation tests.

Corelation test

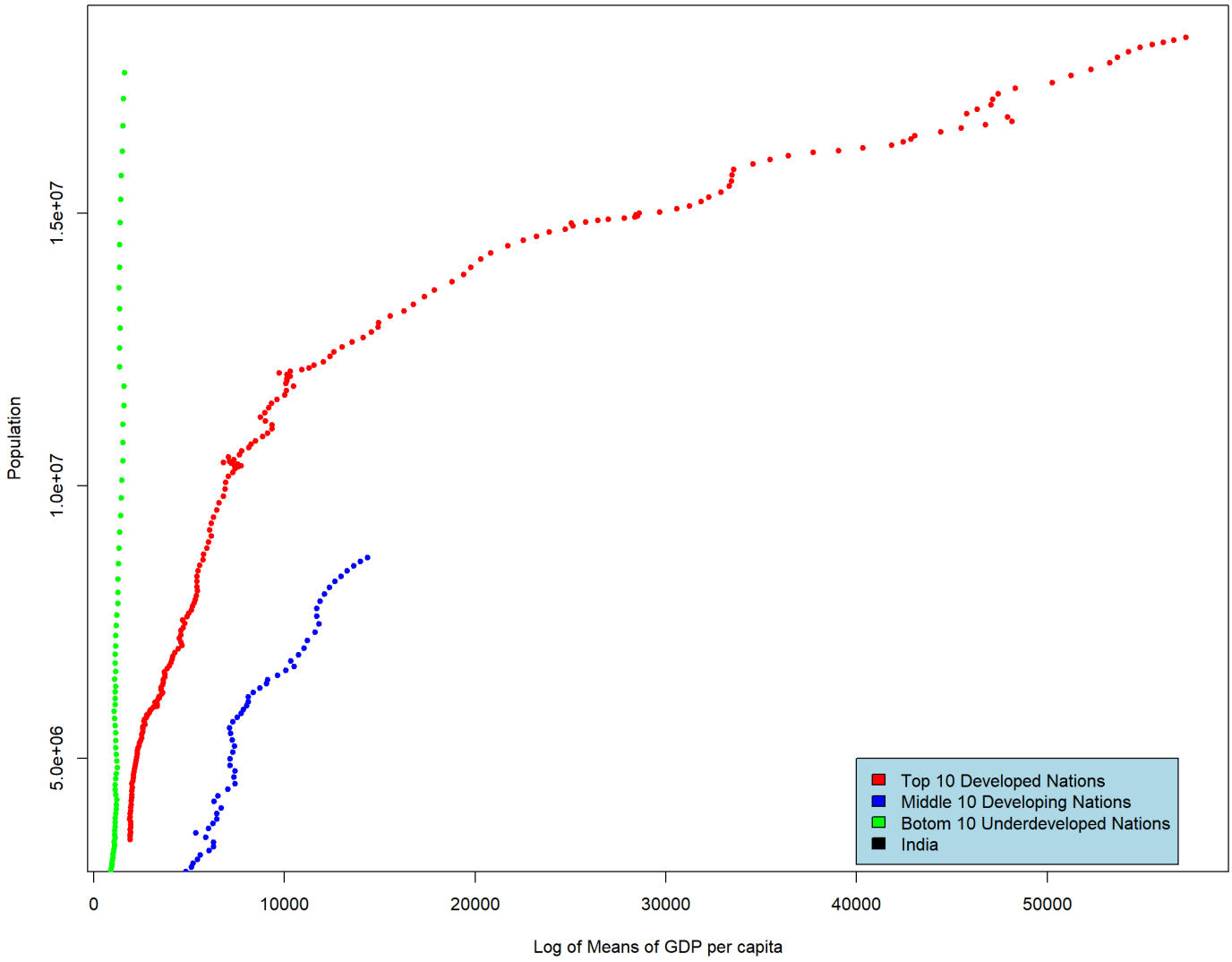
```
cor.test(topmeangdp, topmeanchildrenperwomen)
```

```
##
##  Pearson's product-moment correlation
##
## data:  topmeangdp and topmeanchildrenperwomen
## t = -23.085, df = 224, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8738958 -0.7957202
## sample estimates:
##       cor
## -0.8390882
```

As the cor value is in negative it means with increase of GDP children per women decreases

How population changed with gdp per capita

```
plot(topmeangdp, topmeanpopulation, col = "red", pch = 20, xlab = "Log of Means of GDP per capit
a", ylab = "Population")
points(middlemeangdp, middlemeanpopulation, col= " blue", pch = 20)
points(bottommeangdp, bottommeanpopulation, col ="green", pch= 20)
points(indiagdp, indiapopulation, col = "black", pch = 20)
legend(40000, 5000000, legend=c("Top 10 Developed Nations", "Middle 10 Developing Nations", "Bot
om 10 Underdeveloped Nations",  "India"), fill = c("red", "blue", "green", "black"), bg= "lightb
lue")
```



Hypthesis

With increase in GDP population als increasing, to check this correlation we can do correlation tests.

Corelation test

```
cor.test(topmeangdp, topmeanpopulation)
```

```
##
##  Pearson's product-moment correlation
##
## data:  topmeangdp and topmeanpopulation
## t = 30.203, df = 224, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8668906 0.9190577
## sample estimates:
##       cor
## 0.8960241
```

As the cor value is in postive it means with increase of GDP population increase