GDP and Income groupings of nations

Shravan Kuchkula

3/14/2017

Introduction

In the *World Development Indicators* database, all 189 World Bank member countries, plus 28 other economies with populations of more than 30,000, are classified based on income groups, so that data users can aggregate, group, and compare statistical data of interest, and for the presentation of key statistics. From this database, two data sets: `EdStats` and `GDP rank table` were obtained for the year of 2012 to analyse GDP based rankings and income group classifications.

The `EdStats` data set (*csv format*) for the year 2012 contains 31 variables of which two key variables are:

- **CountryCode** - A unique three letter code to identify a Country/Economy.
- **Income Group** - One of five income groups: low, lower-middle, upper-middle, high OECD and high non-OECD

The `GDP rank table` data set (*csv format*) contains a ranking table with no headers. The header to the columns are added as listed:

- **CountryCode** - A unique three letter code to identify an Country/Economy.
- **Rank** - Ranking based on GDP.
- **Country** - Country name.
- **GDP** - Gross Domestic Product in millions of US dollars.

Both these data sets are merged based on the matching country code to facilitate with the analysis.

Cleaning the data sets

The `GDP rank table` data set has the following problems:

1. Columns names are not mapped correctly to the columns.
2. Actual data starts at row 6.
3. Third column is empty.
4. Sixth column is sparse and contains reference to footnotes.
5. Bottom part of the data set contains additional information.
6. The GDP column contains ".." for some missing values.
7. The CountryCode column contains missing values.

Problems 1-to-6 are fixed while importing the data into R. The script that is used to do this is `GatherData1.R`. As there are missing values in CountryCode which we will use later while merging with EdStats dataset, the rows containing missing CountryCode are removed from the dataset in this script `CleanData1.R`

The `EdStats` data set does not have any problems, it is imported into R using this script: `GatherData2.R`

File and Directory Organization

- `Makefile.txt` - Downloads the two data sets, cleans and merges them.
- `CleanData1.R` - Cleans the GDP Rank table data set.
- `GatherData1.R` - Downloads the GDP Rank table data set.
- `GatherData2.R` - Downloads the EducStats data set.
- `MergeData.R` - Merges GDP and EducStats based on CountryCode.

- `libraries.R` - Downloads and loads the packages required.
- `Analysis.R` - Contains functions used in the analysis.
- `Main.R` - Main script that ties everything together.
- `Report.Rmd` - RMarkdown file that ties data gathering and analysis.
- `Report.md` - Markdown file that renders on Github as a webpage.

The project structure is below:

```
GDPEduc
|_
  Analysis
  |_
    Data
    |_
      Makefile.txt
      CleanData1.R
      GatherData1.R
      GatherData2.R
      MergeData.R
      EDUC.csv
      GDP.csv
      MergedData.csv
  |_
    Analysis.R
    libraries.R
    Main.R
|_
  Paper
    |_
      Report.html
      Report.pdf
      Report.Rmd
      Report.md
```

Instructions to run the code

When you download this project from Github, you will be in project's root directory, which in this case is: GDPEduc. You have 2 methods to reproduce the analysis done in this project.

Method 1: Running `Main.R` script. If you are running from RStudio, then you just need to click 'Run' on the `Main.R` script. If you are running from the R command prompt, then make sure you are in the project root directory and then source the `Analysis/Main.R` script. The `Main.R` script sources the `Makefile.txt` to download, clean and merge the datasets. It then runs the analysis and displays the output.

```
# Running from command prompt
source("Analysis/Main.R")
```

Method 2: Running `Report.Rmd` to knit the RMarkdown document. In RStudio, open the Report.Rmd file from the GDPEduc/Paper directory, knit the `Report.Rmd` file to the desired output. `Report.Rmd` file sources the `Makefile.txt` and runs the analysis as illustrated in this document.

Analysis

Data is gathered in the csv format from the two websites mentioned in the introduction. `Makefile.txt` executes a series of scripts to download the data, import it into R, clean the dataset by removing blank rows/columns and finally merges the two data sets based on *CountryCode*.

```
setwd('../Analysis/Data')
source('Makefile.txt')
```

```
## 3  observations with NA's in CountryCode are removed
```

```
setwd('../../Paper')
```

3 observations in the *CountryCode* column of the `GDP rank table` dataset were blanks. These rows are removed before merging with `EdStats` dataset.

Next, load all the libraries and analysis R scripts that are needed to conduct the analysis. Details of individual scripts can be found in the `File and Directory Structure` section of this document.

```
setwd('../Analysis')
source('libraries.R')
source('Analysis.R')
setwd('../Paper')
```

Question 1

> Merge the data based on the country shortcode. How many of the IDs match?

The data sets `GDP rank table` and `EdStats` are merged based on CountryCode. Invoke the function `idMatches` in `Analysis.R`. **Note:** While merging these 2 data sets, we only removed the NA's from CountryCode column. NA's in other columns like *Ranking* and *Gdp* are not removed while merging these data sets. If we removed the observations where Ranking or GDP were NA's, we would loose information regarding grouped Economies like "World", "AsiaPasific" etc. Thus, the number of IDs that matched when both these datasets are merged based on country shortcode is: **224**

```
num_id_matches <- idMatches()
```

```
paste("The number of IDs matched by merging GDP and EdStats datasets are ",
        num_id_matches)
```

```
## [1] "The number of IDs matched by merging GDP and EdStats datasets are  224"
```

Question 2

> Sort the data frame in ascending order by GDP (so United States is last). What is the 13th country in the resulting data frame?

Before we sort the data frame, we need to fix the following problems with the merged data set: * Remove NA's from GDP and Ranking columns. * Format the GDP data by removing commas "," * Convert GDP data to numeric.

The `gdpRank()` function in the `Analysis.R` script displays the n-th smallest economy. The dataframe is sorted based on the GDP column in ascending order.

```
paste("The 13th smallest GDP country is: ", gdpRank(13))
```

```
## [1] "The 13th smallest GDP country is:  St. Kitts and Nevis"
```

The 13th smallest GDP country is: **"St Kitts and Nevis"**

Question 3

> What are the average GDP rankings for the "High income: OECD" and "High income: nonOECD" groups?

The `groupRankAverages()` function in the `Analysis.R` script displays the average GDP Ranking by income group. The resulting dataframe is then grep'ed for income groups ending in OECD.

```
grpAvgs <- groupRankAverages()
grpAvgs[grepl(".*OECD$", grpAvgs$`Income Group`),]
```

```
## # A tibble: 2 × 2
##        `Income Group`      avg
##                 <chr>    <dbl>
## 1 High income: nonOECD 91.91304
## 2    High income: OECD 32.96667
```

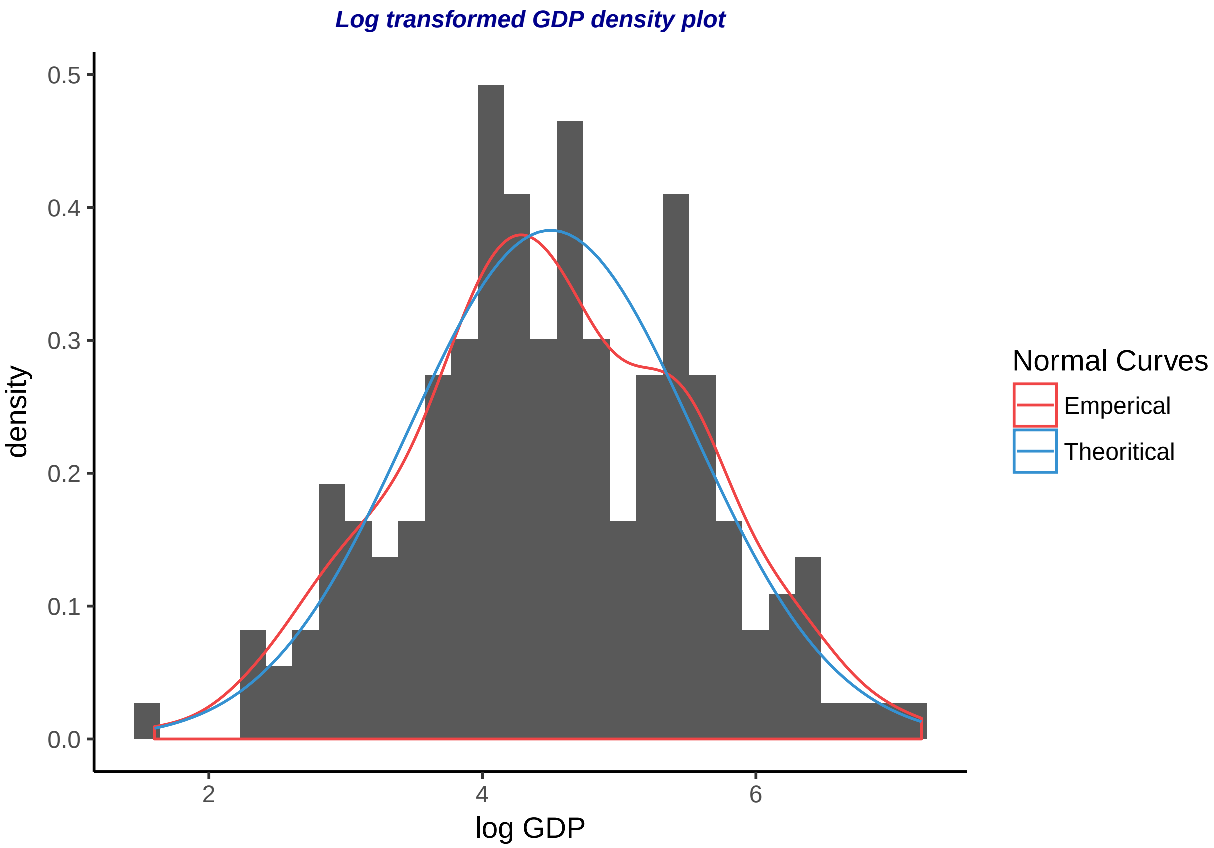
- The average GDP Ranking for *High income: nonOECD* is: **91.91304**
- The average GDP Ranking for *High income: OECD* is: **32.96667**

Question 4

> Show the distribution of GDP value for all the countries and color plots by income group. Use ggplot2 to create your plot.

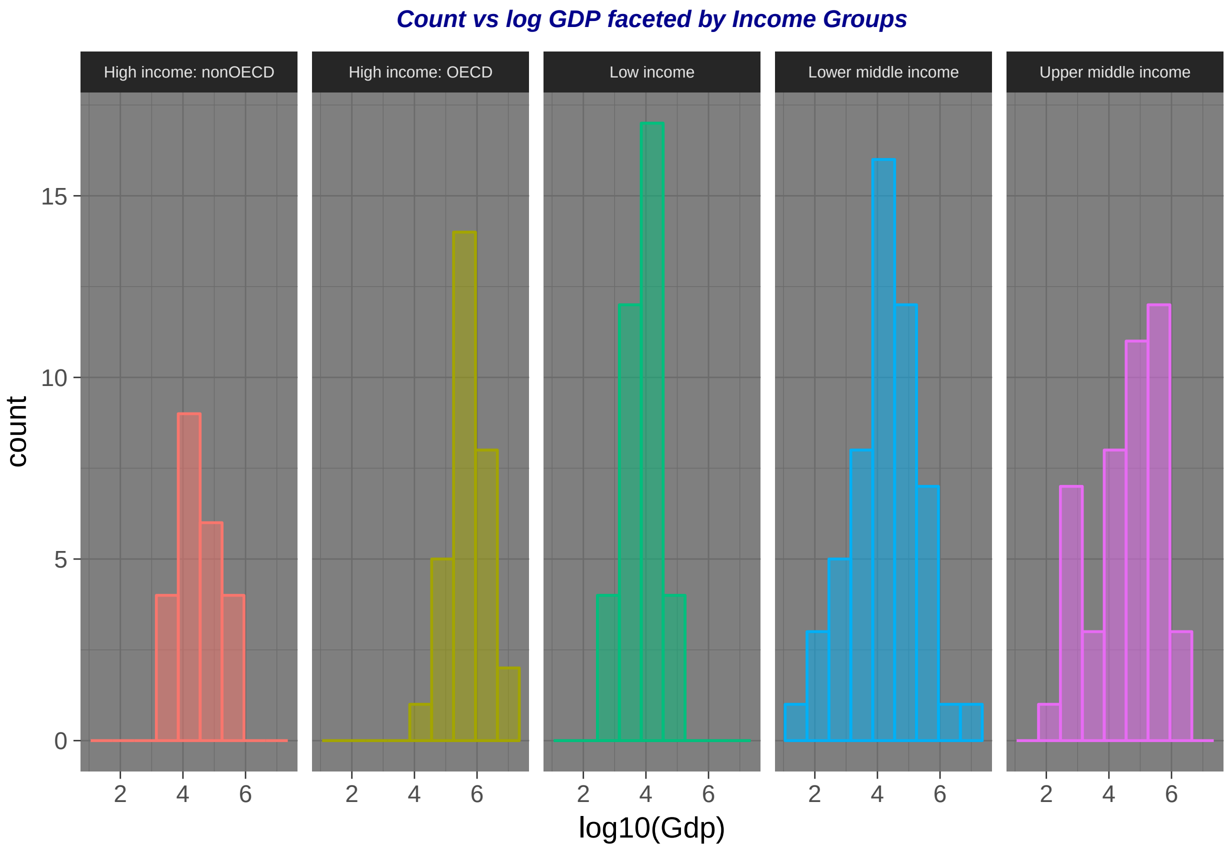
The distribution of GDP value is right skewed, a log transformation is required to visualize the distribution of GDP value. The below plot shows how the log transformed GDP density looks like. The `Emperical` line represents the distribution of GDP value based on the current data set values. The `Theoritical` line represents the theoritical normal curve.

```
# Draw a density curve
cols <- c("Emperical"="#f04546","Theoritical"="#3591d1")
  fun_args <- list(mean = mean(log10(gdpEduc$Gdp)),
                   sd = sd(log10(gdpEduc$Gdp)))
  ggplot(gdpEduc, aes(x = log10(gdpEduc$Gdp))) +
    geom_histogram(aes(y = ..density..)) +
    geom_density(aes(col = "Emperical")) +
    stat_function(fun = dnorm, args = fun_args, aes(col = "Theoritical")) +
    theme_classic() +
    scale_color_manual(name = "Normal Curves", values = cols) +
    labs(x = "log GDP") +
    ggtitle("Log transformed GDP density plot") +
    theme(plot.title = element_text(hjust = 0.5, face="bold.italic",
                                    size = rel(0.8), color = "darkblue"))
```



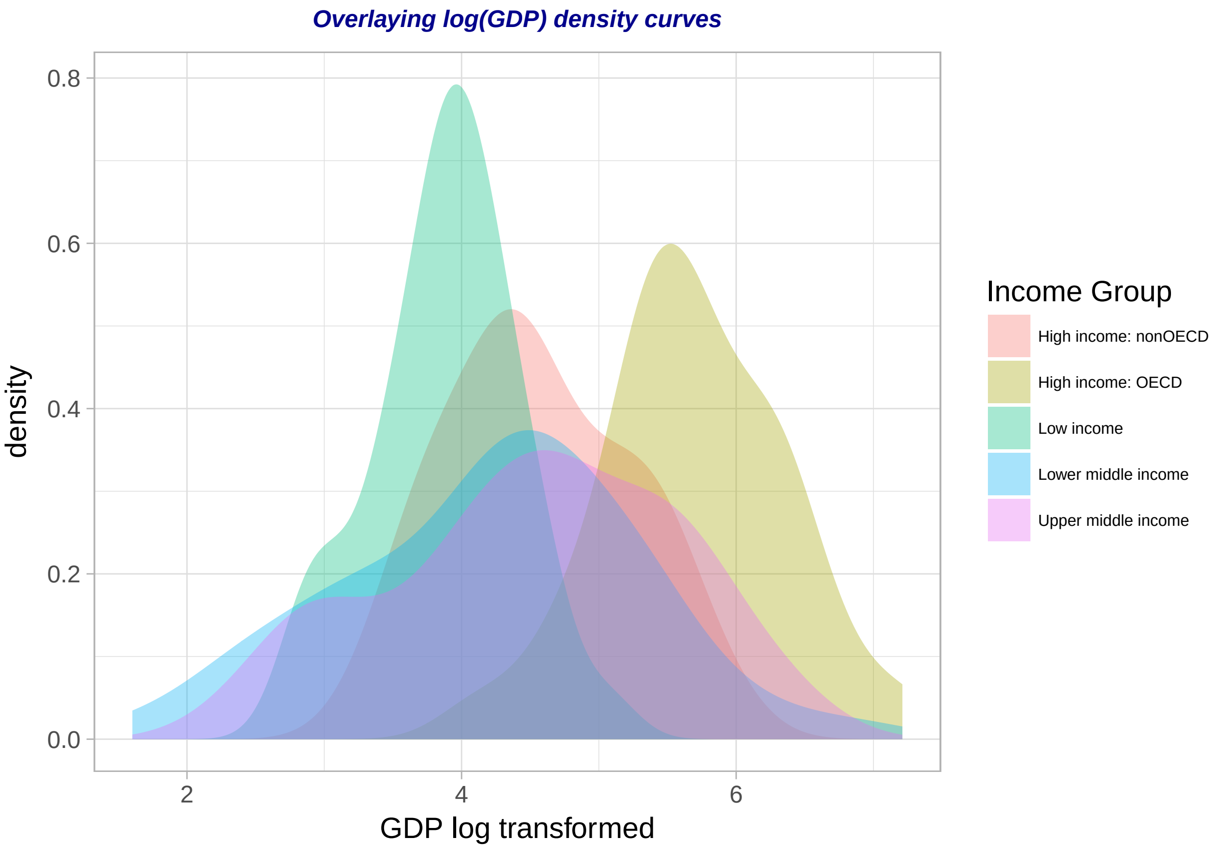
A histogram of count vs log(GDP) grouped by the Income groups reveals at a glance that there are more countries which fall into lower middle income than any other group.

```
# Draw the histogram of count vs log10(GDP) faceted by Income Groups
ggplot(gdpEduc, aes(x = log10(Gdp), col = factor(`Income Group`),
                fill = factor(`Income Group`))) +
  geom_histogram(binwidth = 0.7, alpha = 0.5) +
  facet_grid(.~ factor(`Income Group`)) +
  theme_dark() +
  theme(legend.position = "none", strip.text = element_text(size = rel(0.5))) +
  labs(x = "log10(Gdp)") +
  ggtitle("Count vs log GDP faceted by Income Groups") +
  theme(plot.title = element_text(hjust = 0.5, face="bold.italic",
                                  size = rel(0.8), color = "darkblue"))
```



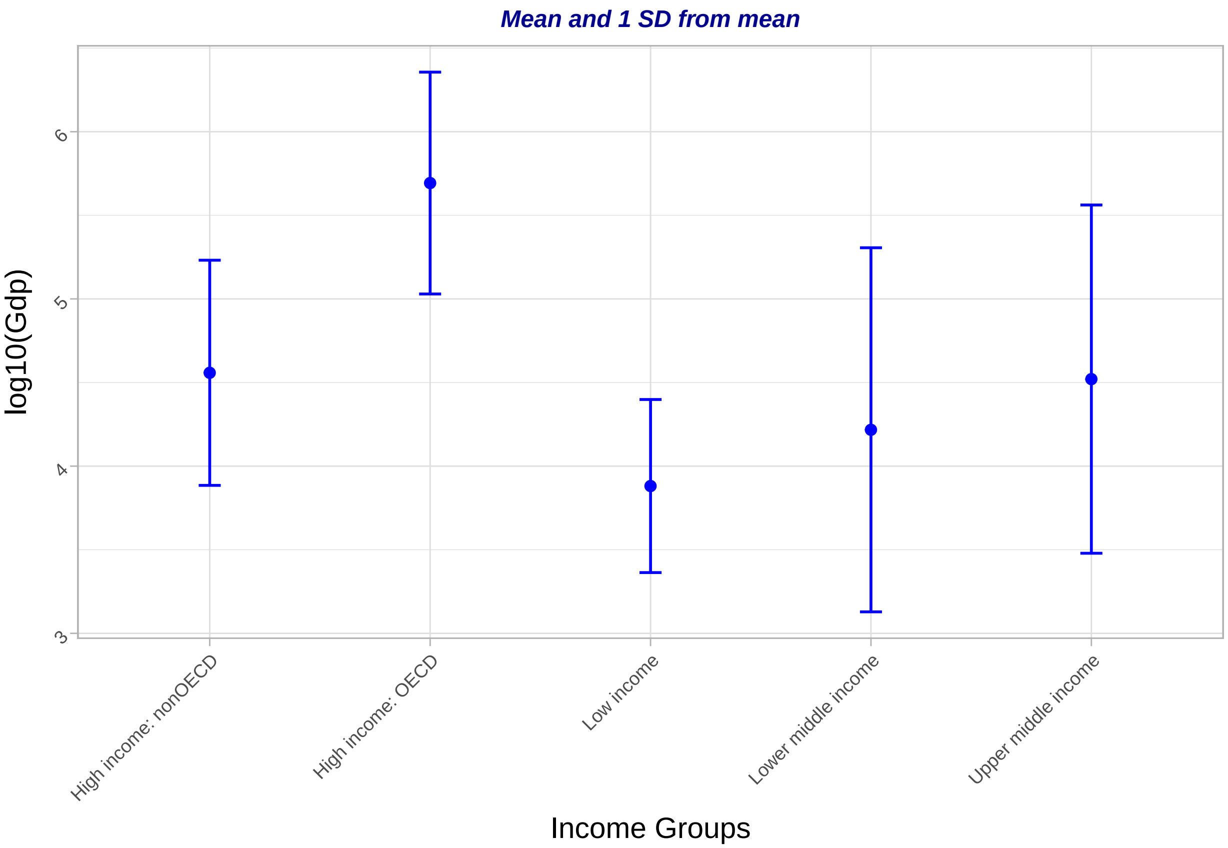
Overlaying the density plots of log GDP values of Income Groups gives us a clear picture of where each group is centered. Interestingly, the *lower middle income* and *upper middle income* are centered very closely and have similar distributions.

```
# Overlay multiple density plots
ggplot(gdpEduc, aes(x = log10(Gdp), fill = factor(`Income Group`))) +
  geom_density(col = NA, alpha = 0.35) +
  theme_light() +
  theme(legend.text = element_text(size=rel(0.5))) +
  labs(x = "GDP log transformed") +
  scale_fill_discrete(name = "Income Group") +
  ggtitle("Overlaying log(GDP) density curves") +
  theme(plot.title = element_text(hjust = 0.5, face="bold.italic",
                                  size = rel(0.8), color = "darkblue"))
```



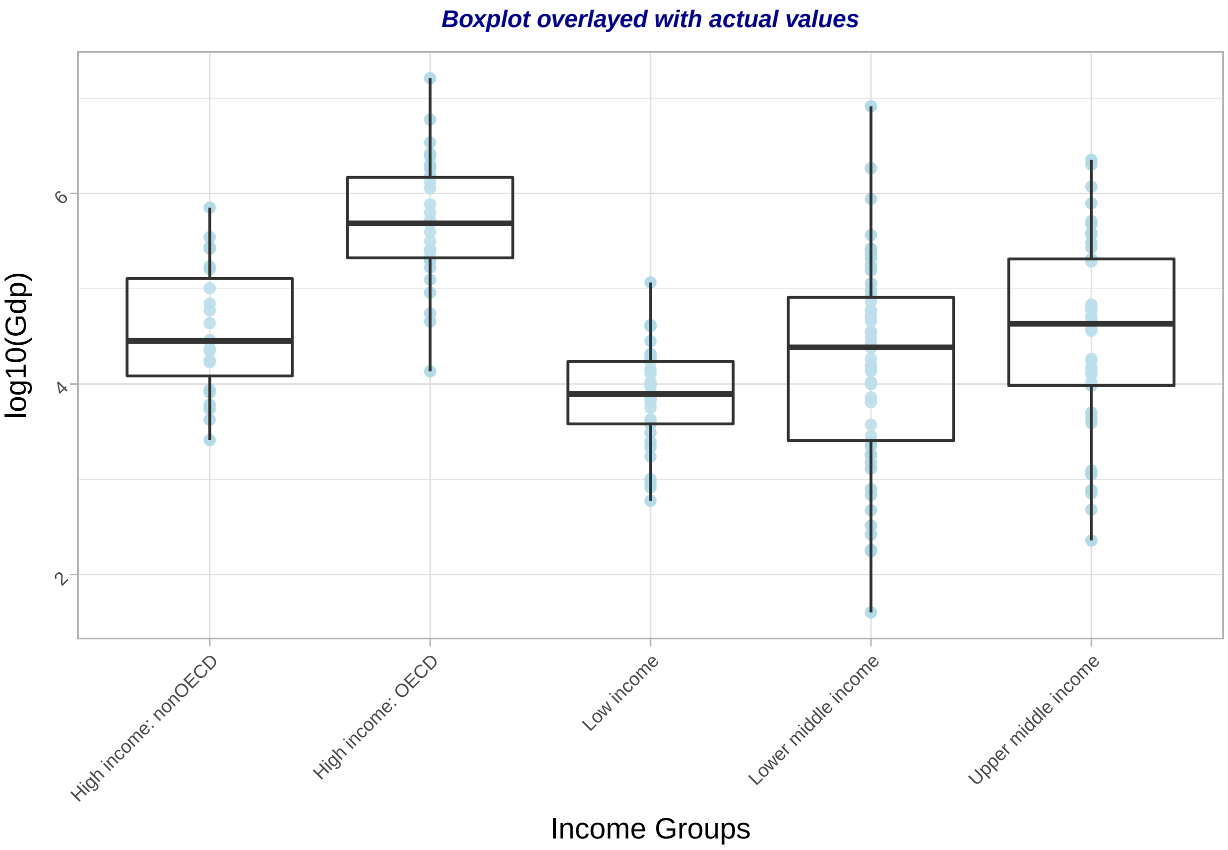
Displaying the mean and 1 standard deviation of the distribution of log(GDP) values within each group shows which group has the highest and lowest spread of values. As we can see here, the *lower middle income* group has the widest spread, whereas the *low income* group has the narrowest spread.

```
# Display mean and 1 sd
ggplot(gdpEduc, aes(x = factor(`Income Group`), y = log10(Gdp))) +
  stat_summary(geom = "point", fun.y = mean, col = "blue") +
  stat_summary(geom = "errorbar", fun.data = mean_sdl,
               fun.args = list(mult = 1), col = "blue", width = 0.1) +
  theme_light() +
  theme(axis.text = element_text(angle = 45, hjust = c(1), size = rel(0.6))) +
  labs(x = "Income Groups") +
  ggtitle("Mean and 1 SD from mean") +
  theme(plot.title = element_text(hjust = 0.5, face="bold.italic",
                                  size = rel(0.8), color = "darkblue"))
```



Five number summaries using Boxplots reveal the spread and distribution of the log(GDP) values within and between income groups. The *blue* dots here represent the actual values within each group.

```
# Box plots with data overlayed to get an idea of the distribution of GDP
ggplot(gdpEduc, aes(x = factor(`Income Group`), y = log10(Gdp))) +
geom_point(colour="lightblue", alpha=0.9, position="identity") +
  geom_boxplot(outlier.size=0, alpha=0.2) +
  theme_light() +
  theme(axis.text = element_text(angle = 45, hjust = c(1), size = rel(0.6))) +
  labs(x = "Income Groups") +
  ggtitle("Boxplot overlayed with actual values") +
  theme(plot.title = element_text(hjust = 0.5, face="bold.italic",
                                  size = rel(0.8), color = "darkblue"))
```



All the above plots are different ways in which we can visualize the distribution of GDP values within and between the income groups.

Question 5

Provide summary statistics of GDP by income groups.

A quick and easy way to obtain group-wise summary statistics is by using the `psych` package. `describeBy` function in this package, takes a continuous variable and a categorical variable and provides descriptive statistics by group. The argument `mat=TRUE` displays the results in a matrix format.

```
# Descriptive statistics using psych package.
  descStatsGDP <- describeBy(gdpEduc$Gdp, gdpEduc$`Income Group`, mat=TRUE)
  descStatsGDP %>%
    select(-item, -vars, -mad) %>%
    print(row.names = FALSE)
```

```
##                 group1  n       mean         sd   median   trimmed   min
##  High income: nonOECD 23  104349.83  165334.45  28373.0  70189.05  2584
##     High income: OECD 30 1483917.13 3070463.52 486528.5 782126.21 13579
##            Low income 37   14410.78   20473.09   7843.0  10715.90   596
##   Lower middle income 54  256663.48 1139619.92  24272.0  51890.64    40
##   Upper middle income 45  231847.84  476872.04  42945.0 113409.27   228
##       max    range     skew  kurtosis         se
##    711050   708466 2.284047  5.230623  34474.616
##  16244600 16231021 3.776308 14.976956 560587.378
##    116355   115759 3.446657 13.898099   3365.755
##   8227103  8227063 6.381095 41.381778 155082.628
##   2252664  2252436 3.013289  8.925480  71087.887
```

Question 6

> Cut the GDP ranking into 5 separate quantile groups. Make a table versus Income Group. How many countries are Lower middle income but among the 38 nations with highest GDP?

The `quantileCut` function from the `lsr` package works much the same way as the base R's `cut` function. However, it differs from the `cut` function in the manner in which it calculates the quantile groups. The `quantileCut` uses the `quantile` function to calculate the groups.
By default, `quantileCut` divides the Ranking column into the following quantile groups: `Levels: (0.811,38.6] (38.6,76.2] (76.2,114] (114,152] (152,190]`. Labels Q1-5 are assigned to these levels. We then use the `dplyr` package's `mutate`, `filter` and `select` functions to tabulate the data.

```
# QuantileCut to divide the GDP rankings into 5 quantile groups
  gdpEducTemp <- gdpEduc
  gdpEducTemp %>%
    mutate(quantiles = quantileCut(gdpEduc$Ranking,5,
                                   labels=c("Q1","Q2","Q3","Q4","Q5"))) %>%
    filter(as.character(quantiles) == "Q1", `Income Group` == "Lower middle income") %>%
    select(CountryCode, Economy, Ranking, `Income Group`, quantiles)
```

```
## # A tibble: 5 × 5
##   CountryCode          Economy Ranking      `Income Group` quantiles
##         <chr>            <chr>   <dbl>               <chr>    <fctr>
## 1         CHN            China       2 Lower middle income        Q1
## 2         IND            India      10 Lower middle income        Q1
## 3         IDN        Indonesia      16 Lower middle income        Q1
## 4         THA         Thailand      31 Lower middle income        Q1
## 5         EGY Egypt, Arab Rep.      38 Lower middle income        Q1
```

There are **5** countries which are lower middle income but fall amoung the 38 nations with highest GDP

Conclusion

GDP data being heavily right-skewed was log transformed to gain a better understanding of the distribution of logGDP for the five income groups. Interesting patters emerged by doing some exploratory data analysis. The distribution of logGDP for countries classified into High income and Low income groups seem to be following a normal distribution. Lower middle and Upper middle income groups have roughly the same mean-this could be attributed to the fact that Lower middle income economies have the largest spread of all the income groups. Five of the top 38 economies are part of the lower middle income group, which has caused this large spread.

References

- Rendering rmarkdown files on github
- qunatileCut
- Reproducible research