# EEEB UN3005/GR5005
# Homework - Week 03 - Due 19 Feb 2019

*Xun Zhao, xz2827*

**Homework Instructions:** Complete this assignment by writing code in the code chunks provided. If required, provide written explanations below the relevant code chunks. Replace "USE YOUR NAME HERE" with your name in the document header. When complete, knit this document within RStudio to generate a pdf. Please review the resulting pdf to ensure that all content relevant for grading (i.e., code, code output, and written explanations) appears in the document. Rename your pdf document according to the following format: hw_week_03_firstname_lastname.pdf. Upload this final homework document to CourseWorks by 5 pm on the due date.

## Problem 1 (5 points)

Find on the class CourseWorks or GitHub site a dataset called `mammals.csv` that contains data on body (kg) and brain (g) masses across 62 mammal species.

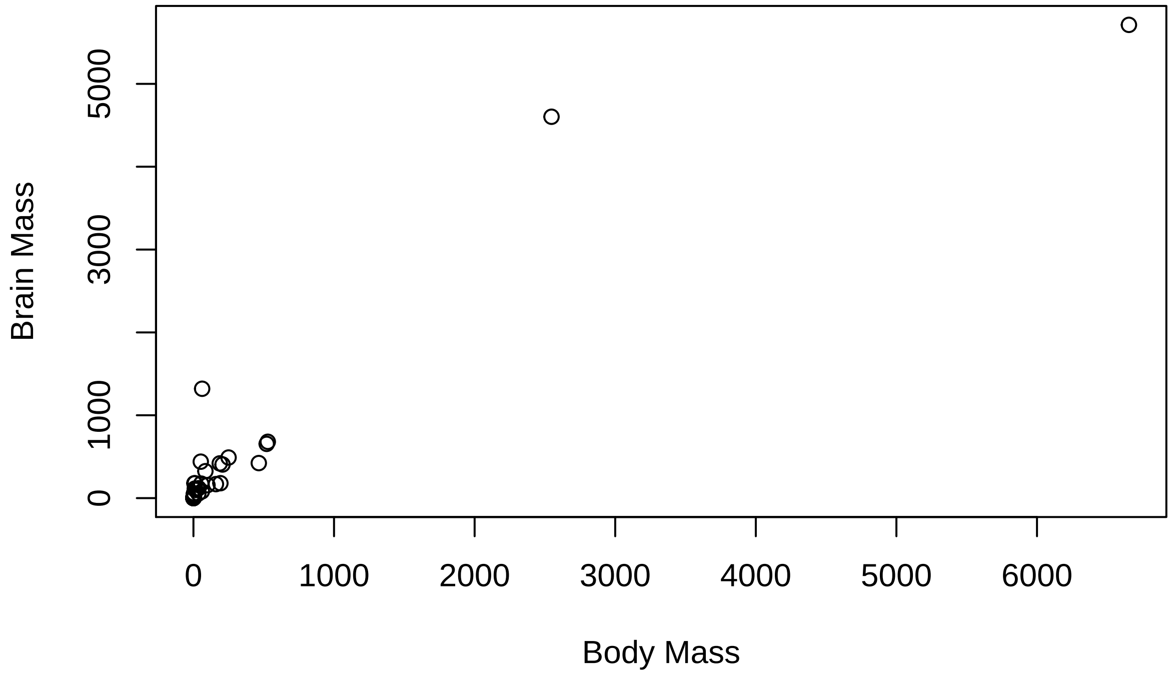
a) Import the `mammals.csv` dataset into R, and assign it to an object called `mammals`. Run the `head()` function on `mammals` to get a glimpse of the raw data.

```
mammals = read.csv('mammals.csv')
head(mammals)
```

```
##           species    body brain
## 1      Arctic fox   3.385  44.5
## 2      Owl monkey   0.480  15.5
## 3 Mountain beaver   1.350   8.1
## 4             Cow 465.000 423.0
## 5       Grey wolf  36.330 119.5
## 6            Goat  27.660 115.0
```

b) Use `plot()` (the base R plotting function) to create a scatter plot of the `mammals` data, with body mass on the x-axis and brain mass on the y-axis. Do you notice anything unusual about the resulting plot? Why might this be the case?

```
plot(mammals$body, mammals$brain, xlab = 'Body Mass', ylab = 'Brain Mass')
```



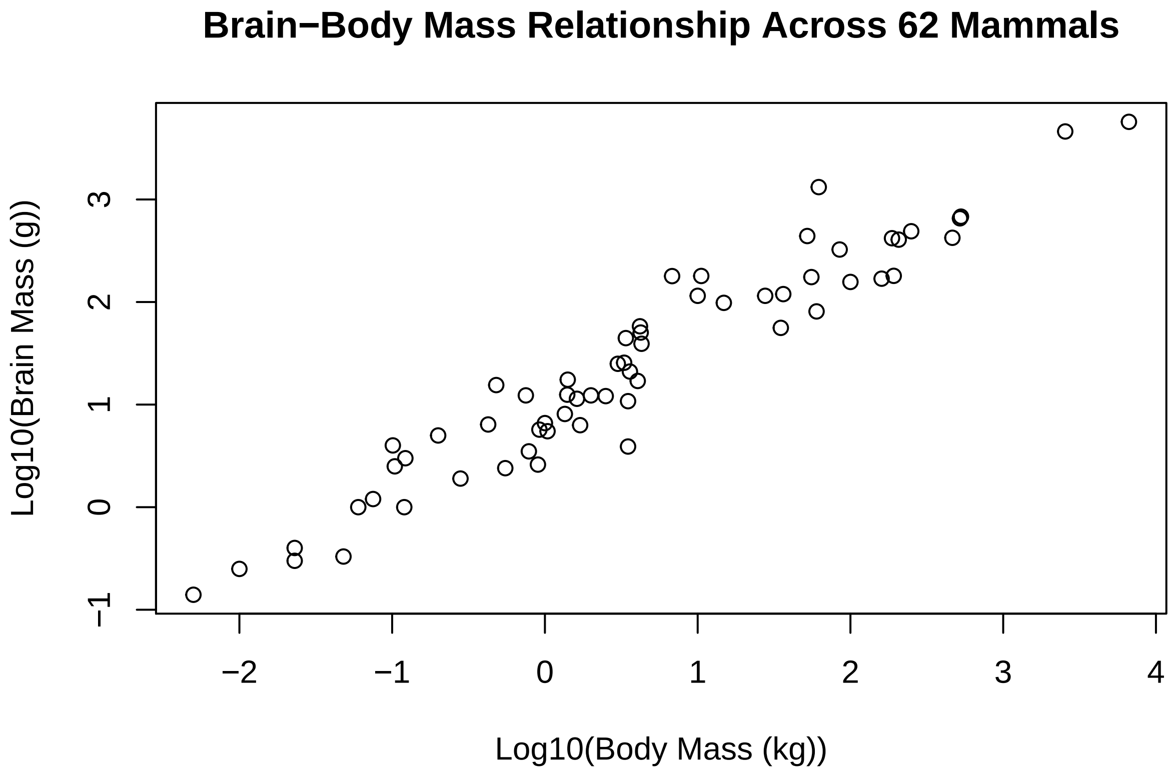
**Answer:**

Because there are two data points that have values much larger than others, R plot function has to adjust the x-axis and y-axis intervals to show all the data points.

c) Now, create a plot analogous to Problem 1b but rather than plotting the raw body and brain mass values, use the `log10()` function to plot log-transformed body and brain mass values. In this plot, have the x-axis label read "Log10(Body Mass (kg))", have the y-axis label read "Log10(Brain Mass (g))", and have the main title read "Brain-Body Mass Relationship Across 62 Mammals".

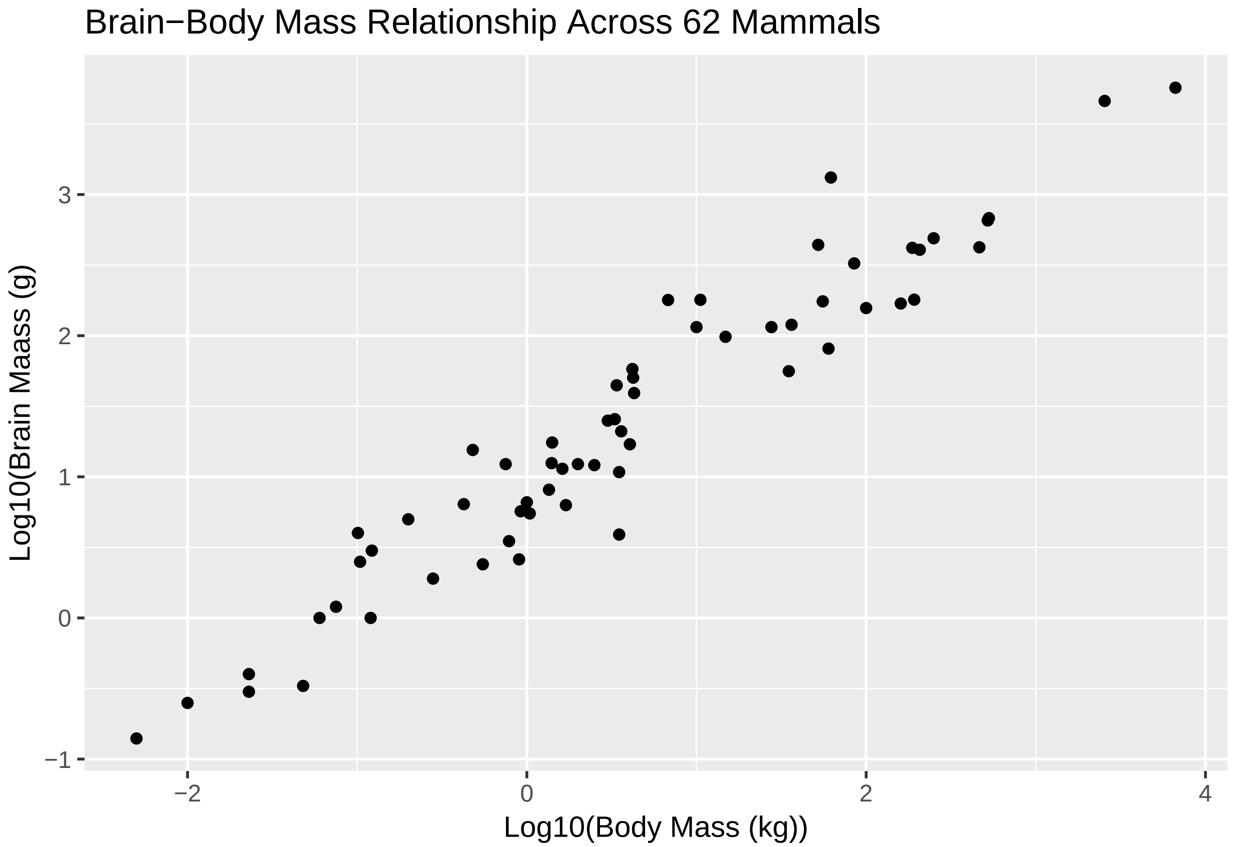
   Hint: There are multiple ways to approach this problem. You may want to create new variables in your `mammals` data frame that are log-transformed versions of the raw variables or you can insert the `log10()` function calls directly within your `plot()` call to do the transformation there. It's up to you!

```
plot(log10(mammals$body), log10(mammals$brain), xlab = 'Log10(Body Mass (kg))', ylab = '
```



d) Replicate the plot in Problem 1c as completely as possible using `ggplot()`. Include the log transformation of both the body and brain values. Modify the x-axis, y-axis, and main title labels as previously described.

```r
ggplot(mammals, aes(x = log10(body), y = log10(brain))) +
    geom_point() +
    xlab('Log10(Body Mass (kg))') +
    ylab('Log10(Brain Maass (g)') +
    ggtitle('Brain-Body Mass Relationship Across 62 Mammals')
```



## Problem 2 (5 points)

Find on the class CourseWorks or GitHub site a demographic dataset called `gapminder.csv`. While this is not *technically* ecological data, one could reasonably argue that human population size and resulting resource demands are critical drivers of ecological and evolutionary processes. Plus, this is just a good dataset to work with when learning plotting.

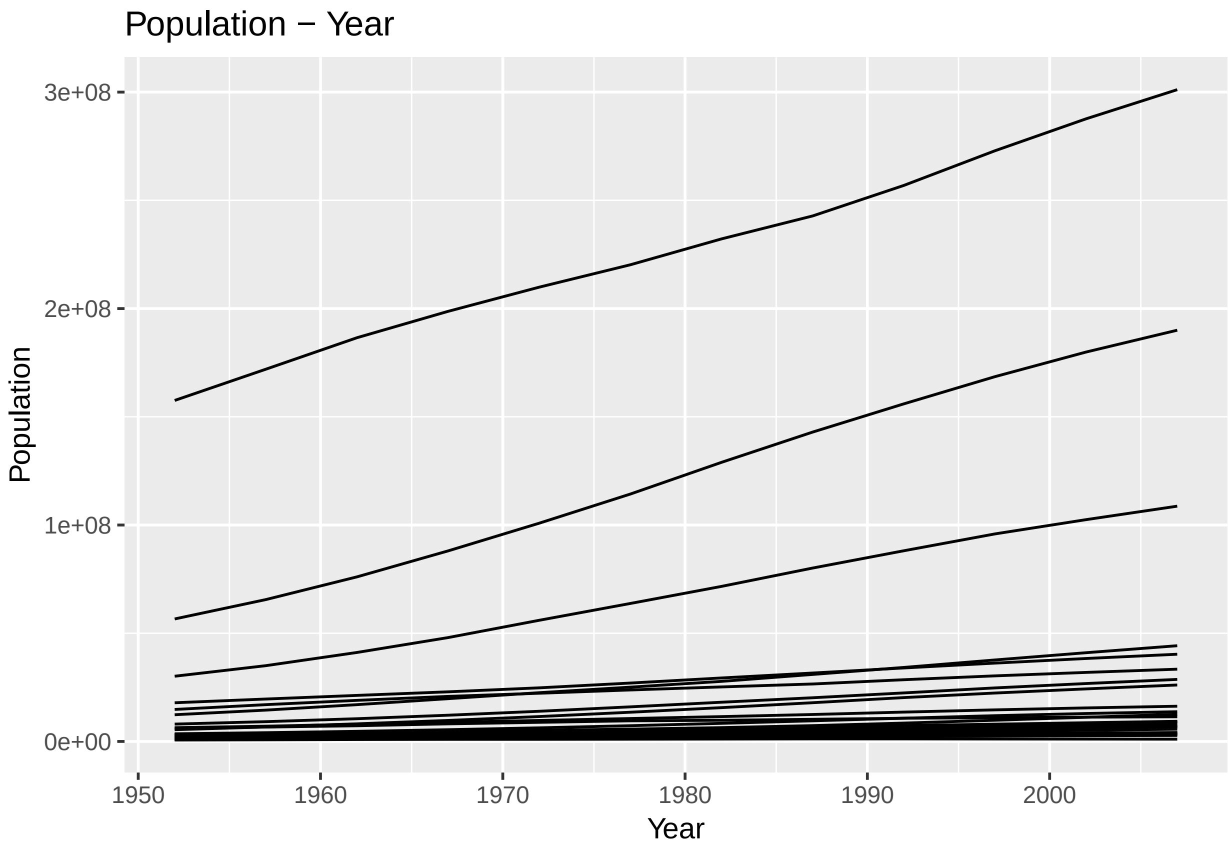
a) Import the `gapminder` dataset into R. Within the data, the `continent` variable has 5 different potential values, one of which is "Americas". Create a dataset called `g.americas` that only contains `gapminder` data from the Americas. Use `summary()` to examine your `g.americas` dataset and verify you only have data from the Americas.

```r
gapminder = read.csv('gapminder.csv')
g.americas = filter(gapminder, continent == 'Americas')
summary(g.americas)
```

```
##        country        continent        year         lifeExp
##  Argentina: 12   Africa  :  0   Min.   :1952   Min.   :37.58
##  Bolivia  : 12   Americas:300   1st Qu.:1966   1st Qu.:58.41
##  Brazil   : 12   Asia    :  0   Median :1980   Median :67.05
##  Canada   : 12   Europe  :  0   Mean   :1980   Mean   :64.66
##  Chile     : 12   Oceania :  0   3rd Qu.:1993   3rd Qu.:71.70
##  Colombia : 12                    Max.   :2007   Max.   :80.65
##  (Other)  :228
##       pop              gdpPercap
##  Min.   :   662850   Min.   : 1202
##  1st Qu.:  2962359   1st Qu.: 3428
##  Median :  6227510   Median : 5466
##  Mean   : 24504795   Mean   : 7136
##  3rd Qu.: 18340309   3rd Qu.: 7830
##  Max.   :301139947   Max.   :42952
##
```

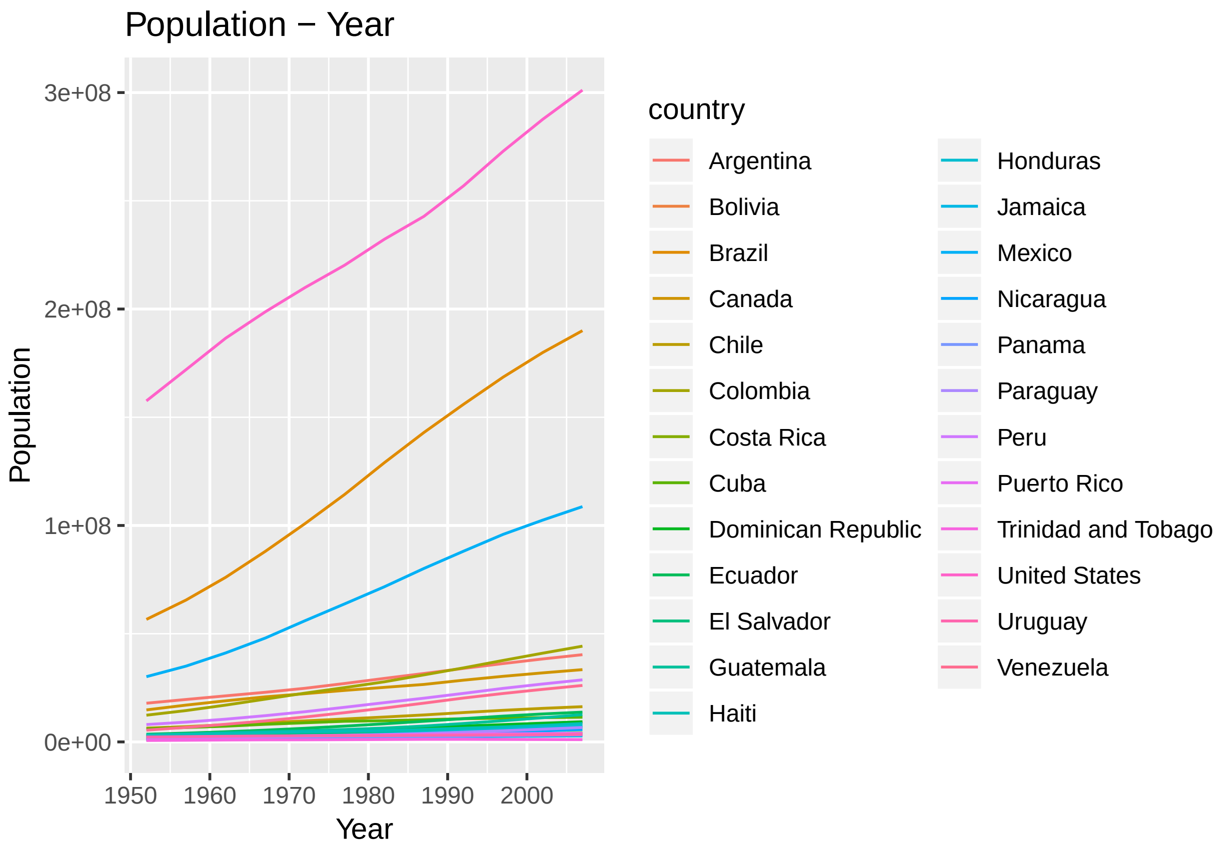
b) Using `ggplot()` and `geom_line()`, make a line plot with year along the x-axis and population size (the `pop` variable) along the y-axis. Within your `aes()` call, you'll need to specify that `group = country` so that the lines appearing in your plot represent population data over time from one country (the plot won't make much sense otherwise).

```
ggplot(g.americas, aes(x = year, y = pop, group = country)) +
    geom_line() +
    xlab('Year') +
    ylab('Population') +
    ggtitle('Population - Year')
```



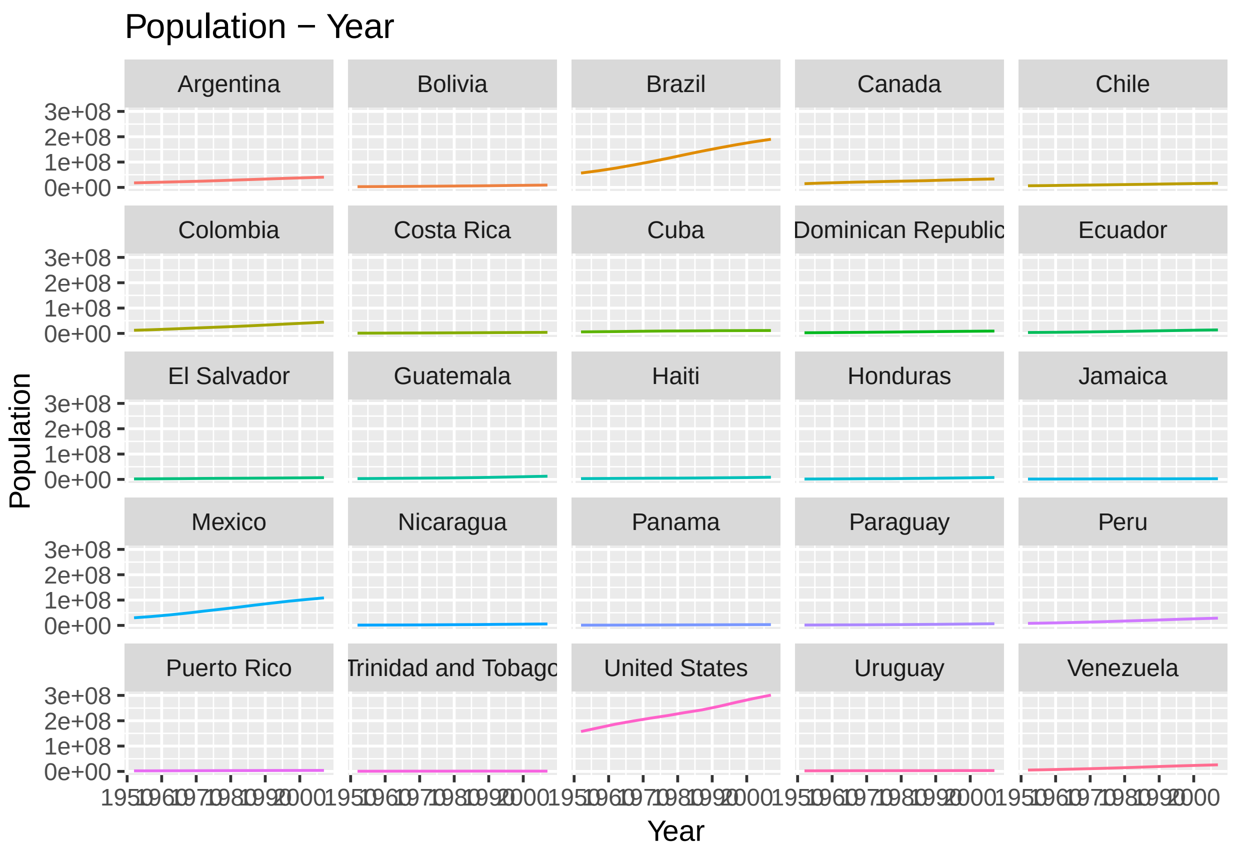
c) Modify the plot you created above to show the lines for each country in different colors. `ggplot()` should automatically produce a legend for you.

```
ggplot(g.americas, aes(x = year, y = pop, group = country, color = country)) +
    geom_line() +
    xlab('Year') +
    ylab('Population') +
    ggtitle('Population - Year')
```



d) Modify the plot from Problem 2c to facet the plot by country. You'll probably want to add the layer `theme(legend.position = "none")` to your plot in order to get rid of all legends. Otherwise, the legend will take up a lot of your plotting space.

```
ggplot(g.americas, aes(x = year, y = pop, group = country, color = country)) +
    geom_line() +
    xlab('Year') +
    ylab('Population') +
    ggtitle('Population - Year') +
    theme(legend.position = "none") +
    facet_wrap(~country)
```



e) The plot you produced in Problem 2d is often called a small multiple plot. Which do you prefer, the plot from Problem 2c or 2d? What do you think are the benefits and drawbacks of each plot's aesthetics?

**Answer:**

For figure 2c, the advantage is that there are enough place to show the main change of data, because it uses a bigger graph. The disadvantage is that it cannot show every data type clearly. At the bottom of the graph, all the data that has small value gather together.

For figure 2d, the advantage is that it show different countries separately. However, when using the same y-axis scale, the small changes of small values are not obvious.