

Question 1 & 2

2025-11-17

Data

Just a brief description of the data used in this analysis. There are 4 files which I am going to import to this file later, two for Kenya, two for Bangladesh.

For Kenya, there are “Kenya_Wealth”, “Kenya_Income”, and in each of them, it contains the following indicators:

- Gini coefficient
- Share of income/wealth held by highest 1%
- Share of income/wealth held by highest 10%
- Share of income/wealth held by bottom 50%
- Share of income/wealth held by middle 40%

The same applies for Bangladesh, with files “Bangladesh_Wealth” and “Bangladesh_Income”.

Note that the data for Bangladesh are imported in form of .xlsx files, as I found that it would be easier to import than the .csv file generated by the website as csv files generated are not in regular format. And I have changed the name for the files for convenience and clarity of the files.

```
# !!! DON'T FORGET TO CHANGE THE WORKING DIRECTORY TO YOUR OWN DIRECTORY !!!
setwd("~/R/BI_Group_Proj/Data")

# Importing Kenya Data
Ken_W_Ineq <- read_delim("Kenya_Wealth_Inequality.csv",
  delim = ";", escape_double = FALSE, col_names = FALSE,
  trim_ws = TRUE, skip = 1
) # This is why I would rather use xls file

Ken_I_Ineq <- read_delim("Kenya_Income_Inequality.csv",
  delim = ";", escape_double = FALSE, col_names = FALSE,
  trim_ws = TRUE, skip = 1
)

# Importing Bangladesh Data
Bang_W_Ineq <- read_excel("Bang_Wealth_Inequality.xlsx", col_names = FALSE)
Bang_I_Ineq <- read_excel("Bang_Income_Inequality.xlsx", col_names = FALSE)

# Make a function for cleaning data sets
clean_data_inequality <- function(x) {
  colnames(x) <- c("Country", "Indicator", "Percentile", "Year", "Value")
  # Columns of Inequality data sets are all in this order, check when use for others
  x <- x %>%
    pivot_wider(
      names_from = Percentile,
      values_from = Value
    ) %>%
}
```

```

filter(!if_all(c(pall, p0p50, p50p90, p90p100, p99p100), is.na)) %>%
# filter out those rows where all the values are NA
select(Country, Year, pall, p0p50, p50p90, p90p100, p99p100) %>%
# To ensure the columns are in correct order and delete indicator column
group_by(Country, Year) %>%
summarise(
  across(
    c(pall, p0p50, p50p90, p90p100, p99p100),
    ~ first(na.omit(.))
  ),
  .groups = "drop"
)
# Note that previously we have 5 lines for a single year, and each
# line only shows a single indicator. By doing this, we combine the data together.
colnames(x) <- c(
  "Country", "Year", "Gini_Coeff", "Share_Bottom50",
  "Share_Middle40", "Share_Top10", "Share_Top1"
)
return(x)
}

Bang_I_Ineq_wider <- clean_data_inequality(Bang_I_Ineq)
Bang_W_Ineq_wider <- clean_data_inequality(Bang_W_Ineq)
Ken_I_Ineq_wider <- clean_data_inequality(Ken_I_Ineq)
Ken_W_Ineq_wider <- clean_data_inequality(Ken_W_Ineq)

longer_format <- function(x) {
  x %>% pivot_longer(
    cols = c(
      Gini_Coeff, Share_Bottom50, Share_Middle40,
      Share_Top10, Share_Top1
    ),
    names_to = "Indicator",
    values_to = "Value"
  )
}

Bang_I_Ineq_longer <- longer_format(Bang_I_Ineq_wider)
Bang_W_Ineq_longer <- longer_format(Bang_W_Ineq_wider)
Ken_I_Ineq_longer <- longer_format(Ken_I_Ineq_wider)
Ken_W_Ineq_longer <- longer_format(Ken_W_Ineq_wider)

```

Question 1

In this sector I will draw some line chart comparing inequality of income & wealth between Kenya and Bangladesh by means of Gini coefficient and share of income/wealth occupied by different social class.

```

# -----Income Inequality-----
I_Inequality_K_and_B <- bind_rows(Ken_I_Ineq_longer, Bang_I_Ineq_longer)

indicator_labels <- c(
  "Gini_Coeff" = "Gini Coefficient",
  "Share_Bottom50" = "Income Share: Bottom 50%",

```

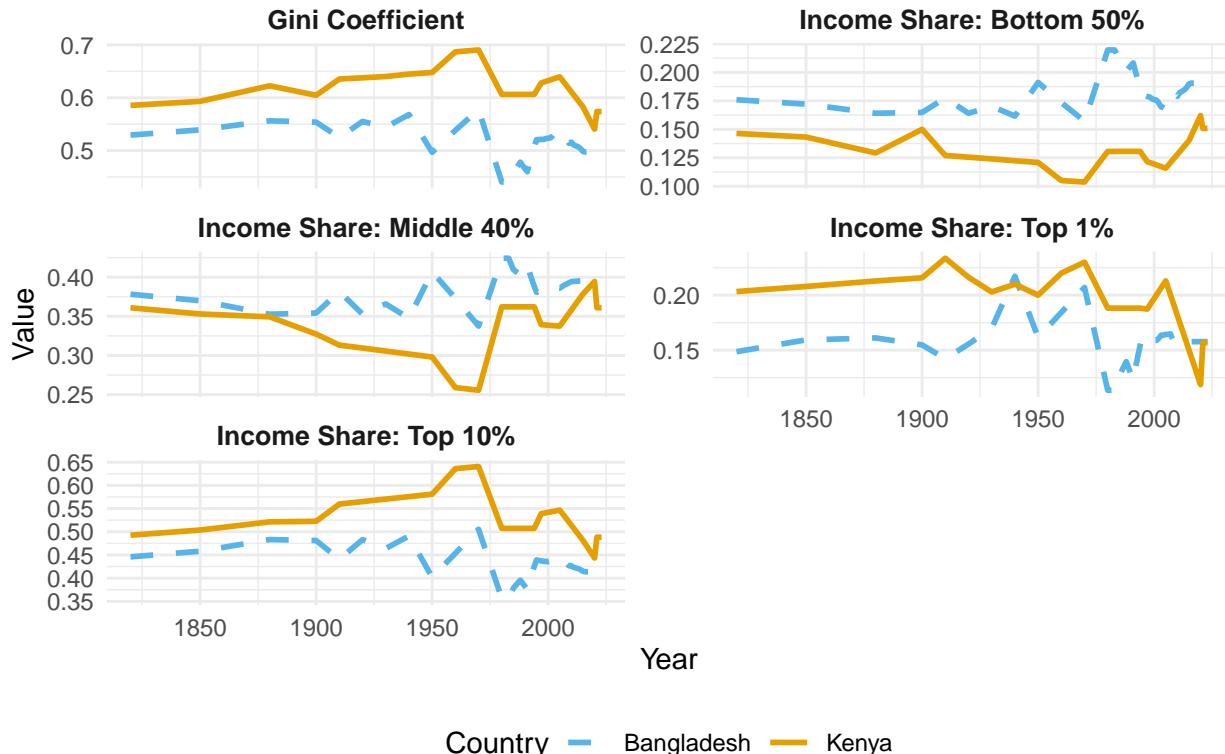
```

    "Share_Middle40" = "Income Share: Middle 40%",
    "Share_Top10" = "Income Share: Top 10%",
    "Share_Top1" = "Income Share: Top 1%"
)
)

I_Inequality_K_and_B %>% ggplot(aes(x = Year, y = Value, color = Country, linetype = Country)) +
  geom_line(linewidth = 1) +
  facet_wrap(~Indicator,
             scales = "free_y", ncol = 2,
             labeller = labeller(Indicator = indicator_labels))
) +
  scale_color_manual(values = c("Kenya" = "#E69F00", "Bangladesh" = "#56B4E9")) +
  scale_linetype_manual(values = c("Kenya" = "solid", "Bangladesh" = "dashed")) +
  labs(
    title = "Income Inequality Indicators: Kenya vs Bangladesh",
    x = "Year",
    y = "Value"
) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    plot.title = element_text(face = "bold", size = 14),
    strip.text = element_text(face = "bold", size = 10)
)

```

Income Inequality Indicators: Kenya vs Bangladesh



```

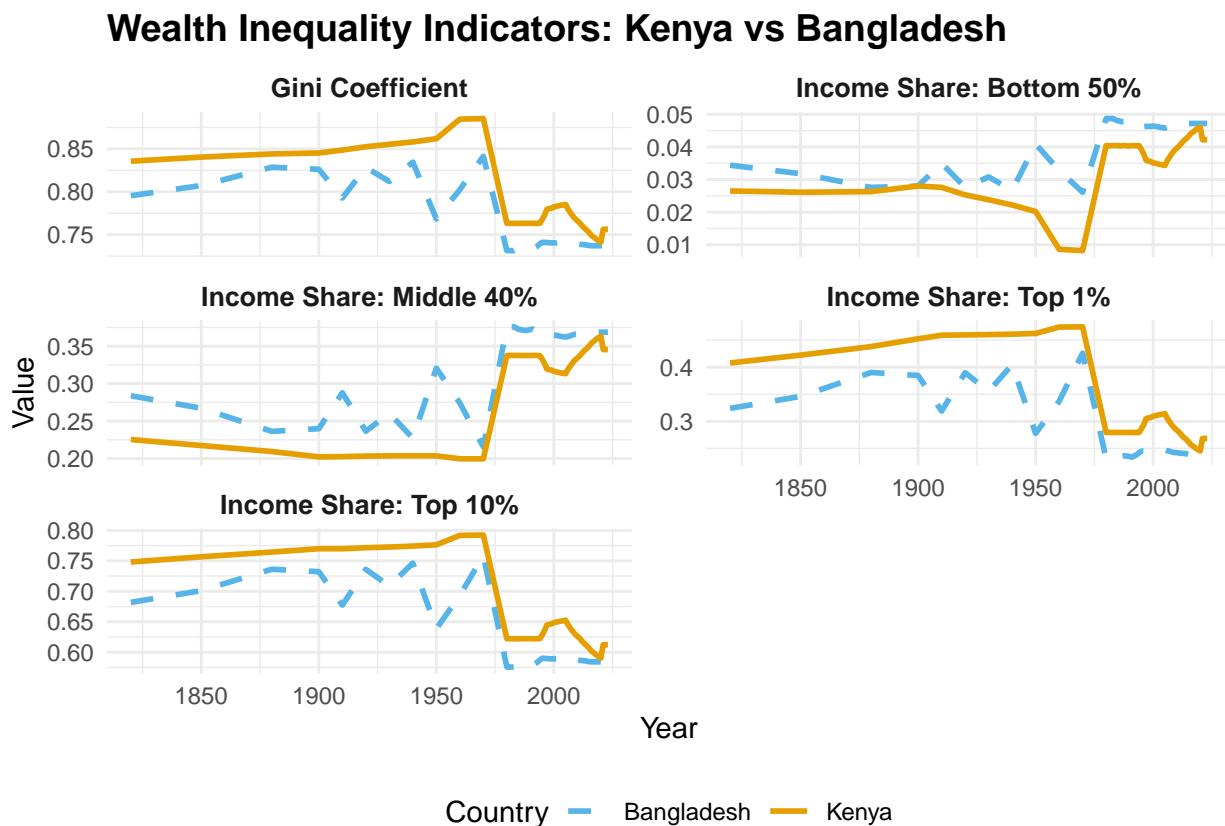
# -----Wealth Inequality-----
W_Inequality_K_and_B <- bind_rows(Ken_W_Ineq_longer, Bang_W_Ineq_longer)

```

```

W_Inequality_K_and_B %>% ggplot(aes(x = Year, y = Value, color = Country, linetype = Country)) +
  geom_line(linewidth = 1) +
  facet_wrap(~Indicator,
             scales = "free_y", ncol = 2,
             labeller = labeller(Indicator = indicator_labels))
) +
  scale_color_manual(values = c("Kenya" = "#E69F00", "Bangladesh" = "#56B4E9")) +
  scale_linetype_manual(values = c("Kenya" = "solid", "Bangladesh" = "dashed")) +
  labs(
    title = "Wealth Inequality Indicators: Kenya vs Bangladesh",
    x = "Year",
    y = "Value"
) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    plot.title = element_text(face = "bold", size = 14),
    strip.text = element_text(face = "bold", size = 10)
)

```



Question 2

```

# Import Data of wider region
SS_Africa_I_Ineq <- read_excel("Data/Sub-Saharan_Africa_Income_Inequality.xlsx",
  col_names = FALSE
)

```

```

SS_Africa_W_Ineq <- read_excel("Data/Sub_Sahara_Africa_Wealth_Inequality.xlsx",
  col_names = FALSE
)

SS_Asia_I_Ineq <- read_excel("Data/SAndSE_Asia_Income_Inequality.xls",
  col_names = FALSE
)
SS_Asia_W_Ineq <- read_excel("Data/SAndSE_Asia_Wealth_Inequality.xls",
  col_names = FALSE
)

# Clean data

SS_Africa_I_Ineq_wider <- clean_data_inequality(SS_Africa_I_Ineq)
SS_Africa_W_Ineq_wider <- clean_data_inequality(SS_Africa_W_Ineq)

SS_Asia_I_Ineq_wider <- clean_data_inequality(SS_Asia_I_Ineq)
SS_Asia_W_Ineq_wider <- clean_data_inequality(SS_Asia_W_Ineq)

# Pivot longer for plot
SS_Africa_I_Ineq_longer <- longer_format(SS_Africa_I_Ineq_wider)
SS_Africa_W_Ineq_longer <- longer_format(SS_Africa_W_Ineq_wider)

SS_Asia_I_Ineq_longer <- longer_format(SS_Asia_I_Ineq_wider)
SS_Asia_W_Ineq_longer <- longer_format(SS_Asia_W_Ineq_wider)

# Then we draw a line chart to compare the income and wealth distribution.
# For simplicity, we compare the Gini coefficient only for now.
# So we write a function first to combine the df's into one.

# NOTE: write in order of regional income, regional wealth, country income, country wealth
W_I_Gini <- function(df1, df2, df3, df4) {
  Keep_Gini_Only <- function(x, y) {
    x %>%
      select(Country, Year, Gini_Coeff) %>%
      mutate(Category = ifelse(y == "I", "Income", "Wealth"))
  }

  df1 <- Keep_Gini_Only(df1, "I")
  df2 <- Keep_Gini_Only(df2, "W")
  df3 <- Keep_Gini_Only(df3, "I")
  df4 <- Keep_Gini_Only(df4, "W")

  df <- bind_rows(df1, df2, df3, df4)
  return(df)
}

Africa_Inequality <- W_I_Gini(
  SS_Africa_I_Ineq_wider, SS_Africa_W_Ineq_wider,
  Ken_I_Ineq_wider, Ken_W_Ineq_wider
)

```

```

SS_Asia_Inequality <- W_I_Gini(
  SS_Asia_I_Ineq_wider, SS_Asia_W_Ineq_wider,
  Bang_I_Ineq_wider, Bang_W_Ineq_wider
)

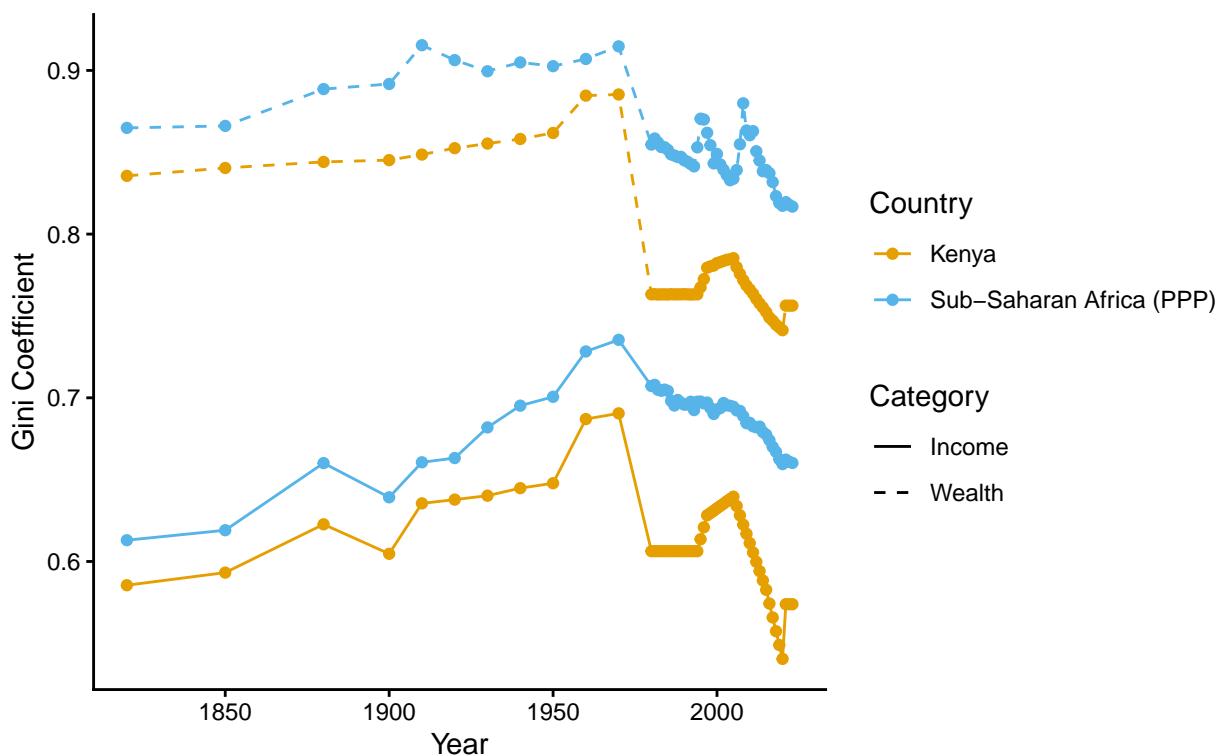
# Compare Sub-Saharan Africa with Kenya

Africa_Inequality %>% ggplot(aes(
  x = Year, y = Gini_Coeff,
  colour = Country, linetype = Category
)) +
  geom_line() +
  geom_point() +
  scale_colour_manual(
    values =
    c(
      "Kenya" = "#E69F00", "Sub-Saharan Africa (PPP)" = "#56B4E9",
      labels = c(
        "Sub-Saharan Africa (PPP)" = "Sub-Saharan Africa",
        "Kenya" = "Kenya"
      )
    )
  ) +
  scale_linetype_manual(
    values =
    c("Wealth" = "dashed", "Income" = "solid")
) +
  labs(
    title = "Gini Coefficient Comparison",
    subtitle = "Kenya v.s. Sub-Saharan Africa",
    x = "Year",
    y = "Gini Coefficient"
) +
  theme_classic()

```

Gini Coefficient Comparison

Kenya v.s. Sub-Saharan Africa

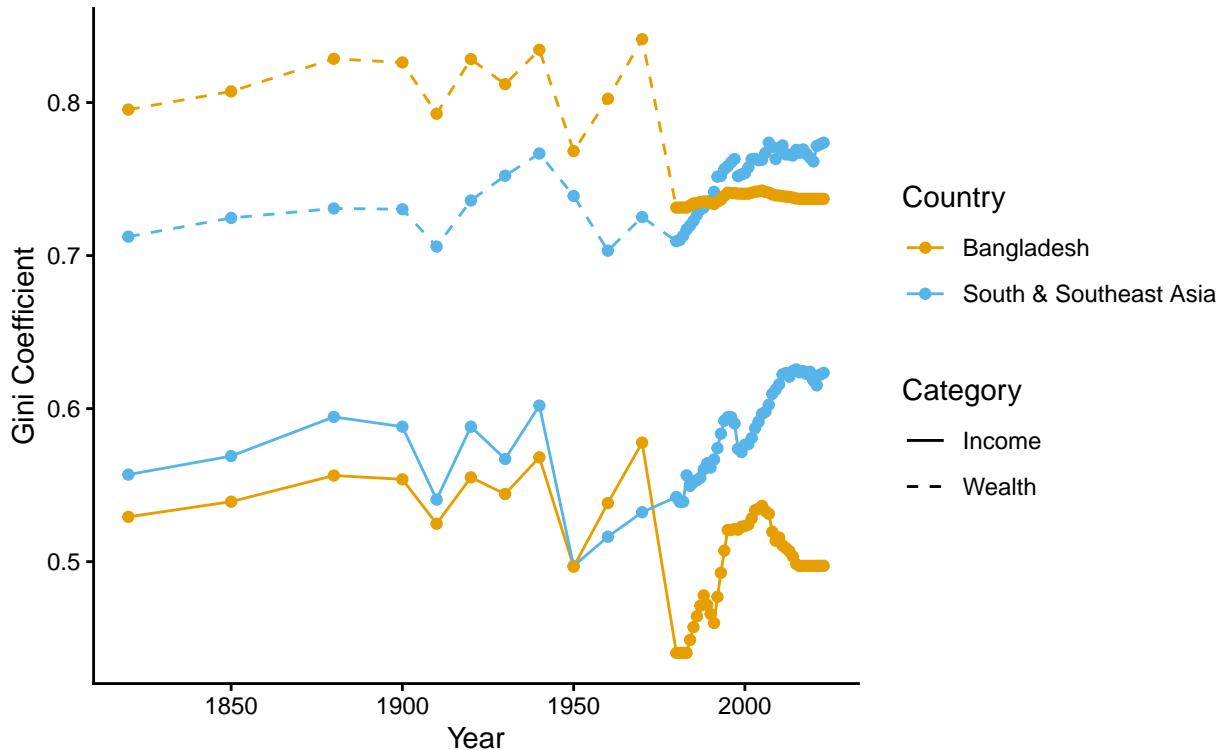


```
# Compare South and Southeast Asia with Bangladesh
```

```
SS_Asia_Inequality %>% ggplot(aes(
  x = Year, y = Gini_Coeff,
  colour = Country, linetype = Category
)) +
  geom_line() +
  geom_point() +
  scale_color_manual(
    values = c("Bangladesh" = "#E69F00", "South & Southeast Asia (PPP)" = "#56b4e9"),
    labels = c(
      "Bangladesh" = "Bangladesh",
      "South & Southeast Asia (PPP)" = "South & Southeast Asia"
    )
  ) +
  scale_linetype_manual(
    values = c("Wealth" = "dashed", "Income" = "solid")
  ) +
  labs(
    title = "Gini Coefficient Comparison",
    subtitle = "South & Southeast Asia v.s. Bangladesh",
    x = "Year",
    y = "Gini Coefficient"
  ) +
  theme_classic()
```

Gini Coefficient Comparison

South & Southeast Asia v.s. Bangladesh



AI Usage

AI language models, such as ChatGPT and Claude, were used during the development of this project, particularly in the coding component. I will provide a detailed explanation of how these tools were used.

1. Coding Support

AI tools are generally used to help clarify the usage of certain functions, and advise on improvement in plotting. For instance, in the plot of question 2, where a graph comparing the income and wealth inequality were drawn, one would like to add a legend to side of the plot. Hence, one wrote the following prompt in Claude:

I have drew a plot, year against Gini coefficient, and in that plot I compared Gini coefficient of two countries in terms of income and wealth respectively. I have used function scale_colour_manual() to assign each country a colour, how should I add a legend on the side and rename the country/region labels as they are not in the desired form. Also, give me a pair of contract colour in Unicode format.

2. Git and Version Control:

AI also played a significant role in supporting our group's use of Git. As we chose GitHub for code sharing and were initially unfamiliar with Git workflow, AI tools guided us through installing Git, connecting to a remote repository, resolving file-upload conflicts, etc. Claude was particularly helpful during a file conflict that nearly caused loss of data, offering step-by-step instruction to recover overwritten work. Example prompts include:

Walk me through the process of pushing a file onto our GitHub repository.

Remind me, how to push files to a new repository?

What is the difference between rebase true/false and fast forward only?