Draft Documents

Team 1 (Hedgehog, Callista, Imtiyaaz, Issac)

2022-10-30

```r
# required library
library(knitr)
library(readr)
library(tidyr)
library(dplyr)
library(ggplot2)
```

2. Data Characteristic

2.1. Nature of Data

The data set is collection The World Bank Data, the variables of interest are extracted from the raw data files and combined into a single data frame for analysis. The final data set includes:

1. **country.code**: Country code
2. **country.name**: Country name
3. **year**: Year
4. **income**: Income class
    - Low income (L)
    - Lower middle income (LM)
    - Upper middle income (UM)
    - High income (H)
5. **reg**: Region
6. **pov**: Poverty headcount ratio
7. **mpi**: Multidimensional Poverty Index
8. **edu.total**: Total expenditure on education (% of GDP)
9. **edu.pri**: Total expenditure on primary education (% of total education expenditure)

10. **edu.sec**: Total expenditure on secondary education (% of total education expenditure)
11. **edu.ter**: Total expenditure on tertiary education (% of total education expenditure)
12. **hlth**: Total expenditure on health (% of GDP)
13. **mil**: Total expenditure on military (% of GDP)
14. **fdi**: Foreign Direct Investment
15. **lbr.part**: Labour force participation (% of population ages 15+)
16. **unemp**: Unemployment rate
17. **pop.gwth.total**: Total population growth rate
18. **pop.gwth.rural**: Total rural population growth rate
19. **pop.gwth.urban**: Total urban population growth rate
20. **gdp.dflt**: GDP deflator
21. **gdr.eql**: Gender equality rating
22. **gcf**: Gross Capital Formation
23. **trade**: Trade = import + export (% of GDP)

Data imports and combining:

```
# helper functions
importWDI <- function(filepath, value_name) {
    df <- read_csv(filepath, skip = 4)

    colnames(df) <- tolower(gsub(" ", ".", colnames(df)))

    df <- df %>%
        pivot_longer(5:ncol(.), names_to = "year", values_to = "value") %>%
        filter(!is.null(value) & !is.na(value)) %>%
        mutate(country.code = factor(country.code), country.name = factor(country.name),
            year = as.numeric(year)) %>%
        select(country.code, country.name, year, value)

    colnames(df)[4] <- value_name

    df
}

importRegionClass <- function(filepath) {
    df <- read_csv(filepath, skip = 4)

    colnames(df) <- tolower(gsub(" ", ".", colnames(df)))

    df %>%
```

```r
        mutate(country.name = factor(country.name), region = factor(region)) %>%
        select(country.name, reg = region)
}

importIncomeClass <- function(filepath) {
    df <- read_csv(filepath, skip = 4)

    colnames(df) <- tolower(gsub(" ", ".", colnames(df)))

    df %>%
        pivot_longer(3:ncol(.), names_to = "year", values_to = "income") %>%
        filter(!is.null(income) & !is.na(income)) %>%
        mutate(country.code = factor(country.code), country.name = factor(country.name),
            year = as.numeric(year), income = factor(income)) %>%
        select(country.code, country.name, year, income)
}
```

```r
# import data
setwd("../data")

poverty.headcount <- importWDI("poverty.headcount.215dollar.csv",
    "pov")
mpi <- importWDI("mpi.csv", "mpi")
education.expenditure.total <- importWDI("total.education.expenditure.csv",
    "edu.total")
education.expenditure.primary <- importWDI("primary.education.expenditure.csv",
    "edu.pri")
education.expenditure.secondary <- importWDI("secondary.education.expenditure.csv",
    "edu.sec")
education.expenditure.tertiary <- importWDI("tertiary.education.expenditure.csv",
    "edu.ter")
health.expenditure <- importWDI("health.expenditure.csv", "hlth")
military.expenditure <- importWDI("military.expenditure.csv",
    "mil")
fdi <- importWDI("fdi.csv", "fdi")
labour.force.participation <- importWDI("labour.force.participation.csv",
    "lbr.part")
unemployment.rate <- importWDI("unemployment.csv", "unemp")
population.growth <- importWDI("population.growth.csv", "pop.gwth.total")
rural.population.growth <- importWDI("rural.population.growth.csv",
    "pop.gwth.rural")
urban.population.growth <- importWDI("urban.population.growth.csv",
    "pop.gwth.urban")
gdp.deflator <- importWDI("gdp.deflator.csv", "gdp.dflt")
gender.equality <- importWDI("gender.equality.csv", "gdr.eql")
gross.capital.formation <- importWDI("gross.capital.formation.csv",
    "gcf")
trade <- importWDI("trade.csv", "trade")
region.class <- importRegionClass("region.class.csv")
income.class <- importIncomeClass("income.class.csv")

setwd("../src")
```

We found that the data sets collected from World Bank's Data helpdesk and The World Bank's Data have different naming convention for certain countries (e.g. "Czechia" vs. "Czechnia Republic"). So we need to rename these countries to avoid some error when joining.

Furthermore, WDI's data sets rate also account for non-country (e.g. country.name = "Low income" or "South Asia"). These special groups are not in our scope of interest, which is national, so we eliminate them.

```
# using poverty.headcount as a naming standard (as other
# data from WDI also use this convention) join a subset of
# data to process the names
d <- poverty.headcount %>%
    select(country.name) %>%
    mutate(inPov = T) %>%
    full_join(income.class %>%
        select(country.name) %>%
        mutate(inIncome = T), by = "country.name") %>%
    full_join(region.class %>%
        select(country.name) %>%
        mutate(inReg = T), by = "country.name") %>%
    mutate(inPov = !is.na(inPov), inIncome = !is.na(inIncome),
        inReg = !is.na(inReg))

d
```

```
## # A tibble: 62,759 x 4
##    country.name inPov inIncome inReg
##    <fct>        <lgl> <lgl>    <lgl>
##  1 Angola       TRUE  TRUE     TRUE
##  2 Angola       TRUE  TRUE     TRUE
##  3 Angola       TRUE  TRUE     TRUE
##  4 Angola       TRUE  TRUE     TRUE
##  5 Angola       TRUE  TRUE     TRUE
##  6 Angola       TRUE  TRUE     TRUE
##  7 Angola       TRUE  TRUE     TRUE
##  8 Angola       TRUE  TRUE     TRUE
##  9 Angola       TRUE  TRUE     TRUE
## 10 Angola       TRUE  TRUE     TRUE
## # ... with 62,749 more rows
## # i Use `print(n = ...)` to see more rows
```

First, remove special economic groups from `poverty.headcount`. We figured these regions will not appear in `income.class` or `region.class`, so we might find something from looking at the countries **only** appear in `poverty.headcount`.

```
d %>%
    filter(inPov & (!inIncome | !inReg)) %>%
    distinct(country.name)
```

```
## # A tibble: 18 x 1
##    country.name
##    <fct>
##  1 Cote d'Ivoire
##  2 Czechia
##  3 East Asia & Pacific
```

```
##   4 Europe & Central Asia
##   5 Fragile and conflict affected situations
##   6 High income
##   7 IDA total
##   8 Latin America & Caribbean
##   9 Low income
##  10 Lower middle income
##  11 Low & middle income
##  12 Middle East & North Africa
##  13 South Asia
##  14 Sub-Saharan Africa
##  15 Sao Tome and Principe
##  16 Turkiye
##  17 Upper middle income
##  18 World
```

Lucky! We can look through these 18 results and compose a list of special regions.

```
spec.reg <- c("Fragile and conflict affected situations", "IDA total",
    "World", "East Asia & Pacific", "Europe & Central Asia",
    "Latin America & Caribbean", "Middle East & North Africa",
    "South Asia", "Sub-Saharan Africa", "Low income", "Low & middle income",
    "Lower middle income", "Upper middle income", "High income")
```

Then, we rename those countries with inconsistent naming convention. Since we should only care about countries whose poverty headcount is available, reusing the list generated above, we can identify:

1. Cote d'Ivoire (also Côte d'Ivoire)
2. Czechia (also Czechoslovakia or Czech Republic)
3. Curacao (also Curaçao)
4. Turkiye (formerly known as Turkey, also Türkiye)
5. Sao Tome and Principe (also São Tomé and Príncipe)

```
# mapping standard name and variation
nameMap <- tibble(standard = c("Cote d'Ivoire", "Czechia", "Czechia",
    "Curacao", "Turkiye", "Turkiye", "Sao Tome and Principe"),
    variation = c("Côte d'Ivoire", "Czechoslovakia", "Czech Republic",
        "Curaçao", "Turkey", "Türkiye", "São Tomé and Príncipe"))

correctName <- function(name) {
    tibble(name = name) %>%
        left_join(nameMap, by = c(name = "variation")) %>%
        mutate(standard = ifelse(is.na(standard), name, standard)) %>%
        select(standard) %>%
        pull()
}

orig.name <- c("Vietnam", "China", "Turkey", "Czechia Republic")
correctName(orig.name)
```

```
## [1] "Vietnam"          "China"            "Turkiye"          "Czechia Republic"
```

Let's test this out!

```r
d <- poverty.headcount %>%
  # correct name here
  mutate(country.name = correctName(country.name)) %>%
  select(country.name) %>%
  mutate(inPov = T) %>%
  full_join(income.class %>%
              # correct name here
              mutate(country.name = correctName(country.name)) %>%
              select(country.name) %>%
              mutate(inIncome = T), by = "country.name") %>%
  full_join(region.class %>%
              # correct name here
              mutate(country.name = correctName(country.name)) %>%
              select(country.name) %>%
              mutate(inReg = T), by = "country.name") %>%
  filter(!(country.name %in% spec.reg)) %>%
  mutate(inPov = !is.na(inPov), inIncome = !is.na(inIncome), inReg = !is.na(inReg))

# countries not in region list, but is in Pov list
d %>%
  filter(!inReg & inPov) %>%
  distinct(country.name) %>%
  nrow()
```

```
## [1] 0
```

```r
# countries not in income list, but is in Pov list
d %>%
  filter(!inIncome & inPov) %>%
  distinct(country.name) %>%
  nrow()
```

```
## [1] 0
```

We are *pretty* confident that there's no inconsistent naming left unprocessed in the data sets.

```r
# Rename countries in all data sets.
poverty.headcount <- poverty.headcount %>%
    mutate(country.name = correctName(country.name))
mpi <- mpi %>%
    mutate(country.name = correctName(country.name))
education.expenditure.total <- education.expenditure.total %>%
    mutate(country.name = correctName(country.name))
education.expenditure.primary <- education.expenditure.primary %>%
    mutate(country.name = correctName(country.name))
education.expenditure.secondary <- education.expenditure.secondary %>%
    mutate(country.name = correctName(country.name))
education.expenditure.tertiary <- education.expenditure.tertiary %>%
    mutate(country.name = correctName(country.name))
health.expenditure <- health.expenditure %>%
    mutate(country.name = correctName(country.name))
military.expenditure <- military.expenditure %>%
```

```
    mutate(country.name = correctName(country.name))
fdi <- fdi %>%
    mutate(country.name = correctName(country.name))
labour.force.participation <- labour.force.participation %>%
    mutate(country.name = correctName(country.name))
unemployment.rate <- unemployment.rate %>%
    mutate(country.name = correctName(country.name))
population.growth <- population.growth %>%
    mutate(country.name = correctName(country.name))
rural.population.growth <- rural.population.growth %>%
    mutate(country.name = correctName(country.name))
urban.population.growth <- urban.population.growth %>%
    mutate(country.name = correctName(country.name))
gdp.deflator <- gdp.deflator %>%
    mutate(country.name = correctName(country.name))
gender.equality <- gender.equality %>%
    mutate(country.name = correctName(country.name))
gross.capital.formation <- gross.capital.formation %>%
    mutate(country.name = correctName(country.name))
trade <- trade %>%
    mutate(country.name = correctName(country.name))
region.class <- region.class %>%
    mutate(country.name = correctName(country.name))
income.class <- income.class %>%
    mutate(country.name = correctName(country.name))
```

Join the data

```
countries <- poverty.headcount %>%
    # We used a full join here so we can conduct a separate
    # analysis on mpi later
full_join(mpi, by = c("country.name", "country.code", "year")) %>%
    left_join(income.class, c("country.name", "country.code",
        "year")) %>%
    left_join(region.class, by = "country.name") %>%
    left_join(education.expenditure.total, by = c("country.name",
        "country.code", "year")) %>%
    left_join(education.expenditure.primary, by = c("country.name",
        "country.code", "year")) %>%
    left_join(education.expenditure.secondary, by = c("country.name",
        "country.code", "year")) %>%
    left_join(education.expenditure.tertiary, by = c("country.name",
        "country.code", "year")) %>%
    left_join(health.expenditure, by = c("country.name", "country.code",
        "year")) %>%
    left_join(military.expenditure, by = c("country.name", "country.code",
        "year")) %>%
    left_join(fdi, by = c("country.name", "country.code", "year")) %>%
    left_join(labour.force.participation, by = c("country.name",
        "country.code", "year")) %>%
    left_join(unemployment.rate, by = c("country.name", "country.code",
        "year")) %>%
    left_join(population.growth, by = c("country.name", "country.code",
```

```r
        "year")) %>%
    left_join(rural.population.growth, by = c("country.name",
        "country.code", "year")) %>%
    left_join(urban.population.growth, by = c("country.name",
        "country.code", "year")) %>%
    left_join(gdp.deflator, by = c("country.name", "country.code",
        "year")) %>%
    left_join(gender.equality, by = c("country.name", "country.code",
        "year")) %>%
    left_join(gross.capital.formation, by = c("country.name",
        "country.code", "year")) %>%
    left_join(trade, by = c("country.name", "country.code", "year")) %>%
    # filter special groups
filter(!(country.name %in% spec.reg))
```

Data preview

```r
head(countries)
```

```
## # A tibble: 6 x 23
##   count~1 count~2  year   pov   mpi income reg   edu.t~3 edu.pri edu.sec edu.ter
##   <fct>   <chr>   <dbl> <dbl> <dbl> <fct>  <fct>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 AGO     Angola   2000  21.4    NA L      Sub-~    2.61      NA      NA      NA
## 2 AGO     Angola   2008  14.6    NA LM     Sub-~    2.69      NA      NA      NA
## 3 AGO     Angola   2018  31.1    NA LM     Sub-~    2.04      NA      NA      NA
## 4 ALB     Albania  1996   0.5    NA LM     Euro~    3.08      NA      NA      NA
## 5 ALB     Albania  2002   1.1    NA LM     Euro~    3.12      NA      NA      NA
## 6 ALB     Albania  2005   0.6    NA LM     Euro~    3.28      NA      NA      NA
## # ... with 12 more variables: hlth <dbl>, mil <dbl>, fdi <dbl>, lbr.part <dbl>,
## #   unemp <dbl>, pop.gwth.total <dbl>, pop.gwth.rural <dbl>,
## #   pop.gwth.urban <dbl>, gdp.dflt <dbl>, gdr.eql <dbl>, gcf <dbl>,
## #   trade <dbl>, and abbreviated variable names 1: country.code,
## #   2: country.name, 3: edu.total
## # i Use `colnames()` to see all variable names
```

```r
str(countries)
```

```
## tibble [1,901 x 23] (S3: tbl_df/tbl/data.frame)
##  $ country.code  : Factor w/ 272 levels "AGO","ALB","ARE",..: 1 1 1 2 2 2 2 2 2 2 ...
##  $ country.name  : chr [1:1901] "Angola" "Angola" "Angola" "Albania" ...
##  $ year          : num [1:1901] 2000 2008 2018 1996 2002 ...
##  $ pov           : num [1:1901] 21.4 14.6 31.1 0.5 1.1 0.6 0.2 0.6 1 0.1 ...
##  $ mpi           : num [1:1901] NA NA NA NA NA NA NA NA NA NA ...
##  $ income        : Factor w/ 4 levels "H","L","LM","UM": 2 3 3 3 3 3 3 4 4 4 ...
##  $ reg           : Factor w/ 7 levels "East Asia & Pacific",..: 7 7 7 2 2 2 2 2 2 2 ...
##  $ edu.total     : num [1:1901] 2.61 2.69 2.04 3.08 3.12 ...
##  $ edu.pri       : num [1:1901] NA NA NA NA NA ...
##  $ edu.sec       : num [1:1901] NA NA NA NA NA ...
##  $ edu.ter       : num [1:1901] NA NA NA NA NA ...
##  $ hlth          : num [1:1901] 1.91 3.32 2.54 NA 6.91 ...
##  $ mil           : num [1:1901] 6.39 3.57 1.87 1.38 1.32 ...
##  $ fdi           : num [1:1901] 8.79e+08 1.68e+09 -6.46e+09 9.01e+07 1.35e+08 ...
```

```
##  $ lbr.part      : num [1:1901] NA NA NA 38.8 59.6 ...
##  $ unemp         : num [1:1901] NA NA NA 12.3 15.8 ...
##  $ pop.gwth.total: num [1:1901] 3.277 3.711 3.276 -0.622 -0.3 ...
##  $ pop.gwth.rural: num [1:1901] 0.921 1.91 1.338 -1.546 -2.169 ...
##  $ pop.gwth.urban: num [1:1901] 5.682 5.02 4.312 0.812 2.181 ...
##  $ gdp.dflt      : num [1:1901] 418.02 19.37 28.17 38.17 3.65 ...
##  $ gdr.eql       : num [1:1901] NA 3 NA NA NA 4 NA NA NA NA ...
##  $ gcf           : num [1:1901] 30.5 30.8 17.9 18.1 35.3 ...
##  $ trade         : num [1:1901] 152.5 121.4 66.4 44.9 68.5 ...
```

```
summary(countries)
```

```
##   country.code  country.name            year           pov
##  BRA    :  36   Length:1901        Min.   :1967   Min.   : 0.00
##  CRI    :  34   Class :character   1st Qu.:2002   1st Qu.: 0.20
##  ARG    :  32   Mode  :character   Median :2009   Median : 1.50
##  USA    :  32                      Mean   :2007   Mean   :10.04
##  DEU    :  30                      3rd Qu.:2014   3rd Qu.:11.60
##  HND    :  30                      Max.   :2021   Max.   :91.50
##  (Other):1707                                     NA's   :58
##       mpi         income                        reg        edu.total
##  Min.   : 2.37   H   :644   East Asia & Pacific       :167   Min.   : 1.033
##  1st Qu.:18.30   L   :253   Europe & Central Asia     :883   1st Qu.: 3.522
##  Median :24.80   LM  :501   Latin America & Caribbean :416   Median : 4.519
##  Mean   :27.06   UM  :438   Middle East & North Africa:107   Mean   : 4.582
##  3rd Qu.:33.30   NA's: 65   North America             : 50   3rd Qu.: 5.457
##  Max.   :74.20              South Asia                : 61   Max.   :15.750
##  NA's   :1446               Sub-Saharan Africa        :217   NA's   :596
##     edu.pri          edu.sec          edu.ter          hlth
##  Min.   : 0.6578   Min.   : 2.724   Min.   : 0.00   Min.   : 1.718
##  1st Qu.:24.0269   1st Qu.:30.138   1st Qu.:16.61   1st Qu.: 5.151
##  Median :30.4730   Median :35.713   Median :20.59   Median : 6.914
##  Mean   :31.6633   Mean   :35.630   Mean   :20.96   Mean   : 6.975
##  3rd Qu.:38.3324   3rd Qu.:41.380   3rd Qu.:25.14   3rd Qu.: 8.565
##  Max.   :70.0950   Max.   :71.587   Max.   :50.44   Max.   :17.733
##  NA's   :1090      NA's   :1094     NA's   :963     NA's   :464
##       mil              fdi                lbr.part         unemp
##  Min.   : 0.000   Min.   :-3.444e+11   Min.   :30.50   Min.   : 0.250
##  1st Qu.: 1.042   1st Qu.: 2.979e+08   1st Qu.:56.17   1st Qu.: 4.513
##  Median : 1.468   Median : 1.709e+09   Median :61.18   Median : 6.880
##  Mean   : 1.787   Mean   : 1.598e+10   Mean   :60.78   Mean   : 8.145
##  3rd Qu.: 2.103   3rd Qu.: 9.821e+09   3rd Qu.:65.49   3rd Qu.:10.078
##  Max.   :19.385   Max.   : 7.338e+11   Max.   :93.00   Max.   :49.700
##  NA's   :105      NA's   :17           NA's   :320     NA's   :267
##  pop.gwth.total    pop.gwth.rural     pop.gwth.urban     gdp.dflt
##  Min.   :-3.6295   Min.   :-8.56066   Min.   :-4.078   Min.   : -26.300
##  1st Qu.: 0.2656   1st Qu.:-0.85664   1st Qu.: 0.510   1st Qu.:   1.696
##  Median : 1.0318   Median :-0.02461   Median : 1.484   Median :   3.865
##  Mean   : 1.0565   Mean   : 0.00083   Mean   : 1.691   Mean   :  15.831
##  3rd Qu.: 1.7761   3rd Qu.: 0.96362   3rd Qu.: 2.657   3rd Qu.:   8.537
##  Max.   : 5.6145   Max.   : 4.59686   Max.   :13.805   Max.   :3333.585
##                    NA's   :14         NA's   :13       NA's   :18
##     gdr.eql          gcf            trade
##  Min.   :1.500   Min.   : 0.00   Min.   :  1.378
```

```
## 1st Qu.:3.000   1st Qu.:19.65   1st Qu.: 51.063
## Median :3.500   Median :22.73   Median : 73.496
## Mean   :3.592   Mean   :23.88   Mean   : 84.429
## 3rd Qu.:4.000   3rd Qu.:26.76   3rd Qu.:105.462
## Max.   :5.000   Max.   :69.48   Max.   :380.104
## NA's   :1635    NA's   :72      NA's   :57
```

There's no NA in `reg`, which is a sign that all naming in the data is remedied. There's some expected NAs in `income` and `pov`, as these data are collected by year. There's a substantial amount of missing data in `mpi`, as this is a relative new concept. We will address the nature, and processing of missing data in the next sections.

2.2. Missing values

As observed from the summary above, the data set contains a lot of missing values in some of the variables.

```
mean(is.na(countries))
```

```
## [1] 0.1896942
```

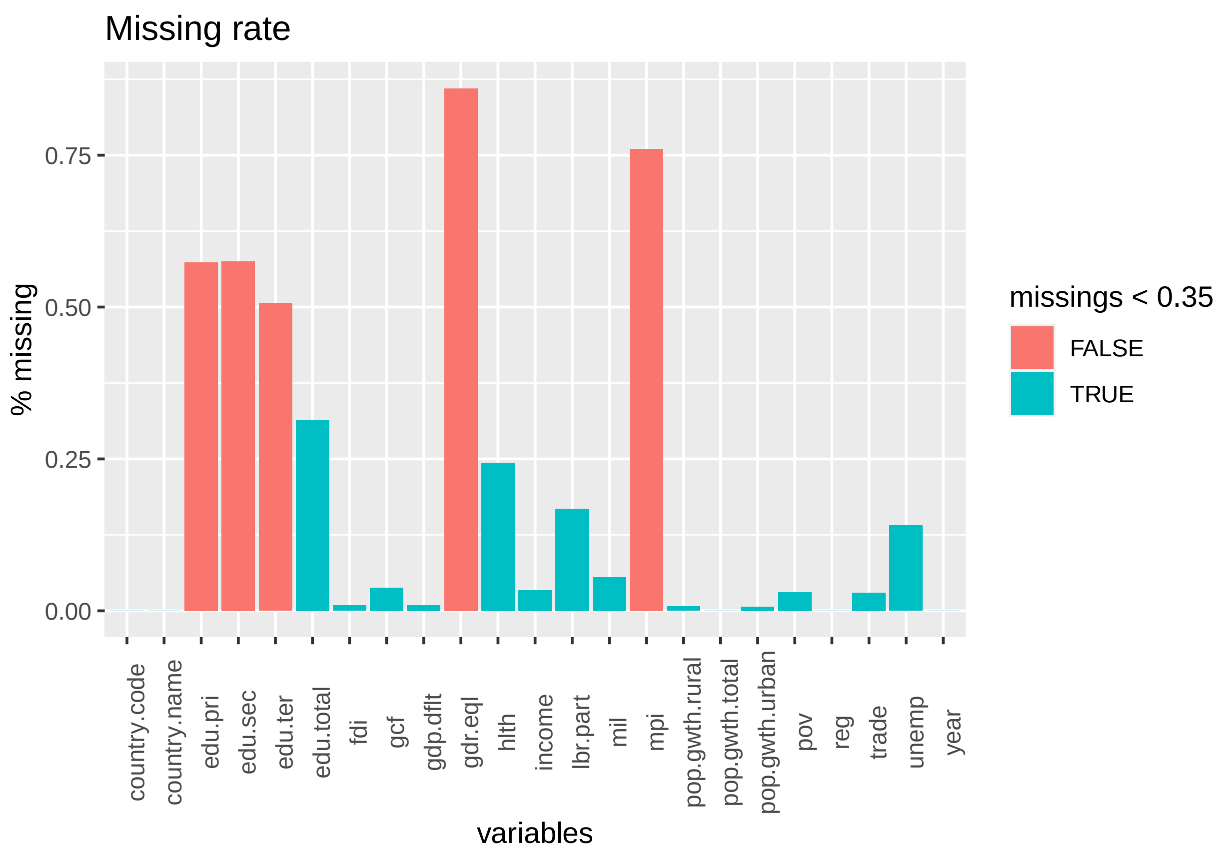
About 19% of the data set is missing.

```
nCompleteObs <- sum(complete.cases(countries))
print(paste("No. of complete cases:", nCompleteObs))
```

```
## [1] "No. of complete cases: 3"
```

There are only 3 complete cases where all the variable is available. This is nowhere near acceptable to conduct any meaningful analysis. Therefore, we need to eliminate some variables for a more balance data set.

```
missings <- colMeans(is.na(countries))
ggplot(mapping = aes(x = names(missings), y = missings, fill = missings <
    0.35)) + geom_bar(stat = "identity") + ggtitle("Missing rate") +
    xlab("variables") + ylab("% missing") + theme(axis.text.x = element_text(size = 9,
    angle = 90))
```




```
missings[missings > 0.35]
```

```
##       mpi   edu.pri   edu.sec   edu.ter   gdr.eql
## 0.7606523 0.5733824 0.5754866 0.5065755 0.8600736
```

There are 5 variables with missing rate >35%.

These can be very useful and relevant information (Akbar et al. 2019). However, we would like to exclude these variables from some first analyses to make use of the richer set of data. We can conduct a separate analysis with these variable to gain more insight.

```
# variables with high missing rate
hMiss <- names(missings[missings > 0.35])
# exclude these variables in countries1
countries1 <- countries %>%
    select(!hMiss)
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(hMiss)` instead of `hMiss` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
str(countries1)
```

```
## tibble [1,901 x 18] (S3: tbl_df/tbl/data.frame)
##  $ country.code   : Factor w/ 272 levels "AGO","ALB","ARE",..: 1 1 1 2 2 2 2 2 2 2 ...
##  $ country.name   : chr [1:1901] "Angola" "Angola" "Angola" "Albania" ...
##  $ year           : num [1:1901] 2000 2008 2018 1996 2002 ...
##  $ pov            : num [1:1901] 21.4 14.6 31.1 0.5 1.1 0.6 0.2 0.6 1 0.1 ...
##  $ income         : Factor w/ 4 levels "H","L","LM","UM": 2 3 3 3 3 3 3 4 4 4 ...
##  $ reg            : Factor w/ 7 levels "East Asia & Pacific",..: 7 7 7 2 2 2 2 2 2 2 ...
##  $ edu.total      : num [1:1901] 2.61 2.69 2.04 3.08 3.12 ...
##  $ hlth           : num [1:1901] 1.91 3.32 2.54 NA 6.91 ...
##  $ mil            : num [1:1901] 6.39 3.57 1.87 1.38 1.32 ...
##  $ fdi            : num [1:1901] 8.79e+08 1.68e+09 -6.46e+09 9.01e+07 1.35e+08 ...
##  $ lbr.part       : num [1:1901] NA NA NA 38.8 59.6 ...
##  $ unemp          : num [1:1901] NA NA NA 12.3 15.8 ...
##  $ pop.gwth.total: num [1:1901] 3.277 3.711 3.276 -0.622 -0.3 ...
##  $ pop.gwth.rural: num [1:1901] 0.921 1.91 1.338 -1.546 -2.169 ...
##  $ pop.gwth.urban: num [1:1901] 5.682 5.02 4.312 0.812 2.181 ...
##  $ gdp.dflt       : num [1:1901] 418.02 19.37 28.17 38.17 3.65 ...
##  $ gcf            : num [1:1901] 30.5 30.8 17.9 18.1 35.3 ...
##  $ trade          : num [1:1901] 152.5 121.4 66.4 44.9 68.5 ...
```

Re-evaluate the `countries1` set.

```
mean(is.na(countries1))
```

```
## [1] 0.06037758
```

```
sum(complete.cases(countries1))
```

```
## [1] 937
```

```
mean(complete.cases(countries1))
```

```
## [1] 0.4928985
```

On average, each column has 6% missing rate, results in 937 complete data point (i.e. 49%). This can be a sufficient number for the analysis. However, the missing data can induce loss of power due to the reduced sample size, and some other biases depending on which variables is missing.

```
# complete rate of data by regions
countries1 %>%
    mutate(isComplete = complete.cases(.)) %>%
    group_by(reg) %>%
    summarise(complete.rate = mean(isComplete)) %>%
    arrange(desc(complete.rate))
```

```
## # A tibble: 7 x 2
##   reg                          complete.rate
##   <fct>                                <dbl>
## 1 Europe & Central Asia                0.618
## 2 Latin America & Caribbean            0.505
## 3 Middle East & North Africa           0.449
```

```
## 4 East Asia & Pacific                    0.437
## 5 South Asia                             0.213
## 6 Sub-Saharan Africa                     0.184
## 7 North America                          0.14
```

Countries from North America, Sub-Saharan Africa, and South Asia have the highest rate of missing data. We suspect that Sub-Saharan Africa, and South Asia are comparably less accessible regions. We also know that Americans don't like filling out forms, so their high rate of missing data is understandable as well.

Still, we need to find a way to address this issue. we propose several approaches:

1. **Use complete cases**: Only use the complete cases for the analysis. This is a straightforward approach, but doesn't resolve the bias resulted from the mass loss of data.
2. **Selectively remove variables with high missing rate**: The same as we did before, but this process should be carried out carefully as we run the chance of dropping an important variable.
3. **Update the data set as we select variables**: As we drop insignificant variables (in backward selection), the number of NAs are changed as well. We can utilize the extra complete cases to build the next model in the steps.
4. **Imputation**: The idea is to replace the missing observations on the response or the predictors with artificial values that try to preserve the data set structure. This is a quite complex topic of its own, but we think why not. You can read more at from Arel-Bundock and Pelc (2018).

2.3. Descriptive Analytics

(To be done) ## 2.4. Data Source

- poverty.headcount
- mpi
- education.expenditure.primary
- education.expenditure.secondary
- education.expenditure.tertiary
- education.expenditure.total
- health.expenditure
- military.expenditure
- fdi
- unemployment.rate
- labour.force.participation
- gender.equality
- population.growth
- urban.population.growth
- rural.population.growth
- gdp.deflator
- gross capital formation
- trade
- region.class
- income.class
- gross.capital.formation

3. Model Selection and Interpretation

3.1. Assumption Check (To be done)

3.2. Ordinary Multiple Linear Regression

We conduct a normal linear regression, following the approaches mentioned above to address missing values issues.

3.2.1. Use Complete Cases (To be done)

3.2.1.1. Model Fitting

3.2.1.2. Assessment

3.2.1.3. Interpretation

3.2.2. Selectively remove variables with high missing rate (To be done)

3.2.2.1. Model Fitting

3.2.2.2. Assessment

3.2.2.3. Interpretation

3.2.3. Update the data set as we select variables (To be done)

3.2.3.1. Model Fitting

3.2.3.2. Assessment

3.2.3.3. Interpretation

3.2.4. Imputation (To be done)

3.2.4.1. Model Fitting

3.2.4.2. Assessment

3.2.4.3. Interpretation

3.3. Panel Data Analysis (To be done)

4. Conclusion

5. Appendix

6. References

Akbar, Muhammad, Mukaram Khan, Haidar Farooqe, and Kaleemullah. 2019. "Public Spending, Education and Poverty: A Cross Country Analysis" 4 (April): 12–20.

Arel-Bundock, Vincent, and Krzysztof J. Pelc. 2018. "When Can Multiple Imputation Improve Regression Estimates?" *Political Analysis* 26 (2): 240–45. https://doi.org/10.1017/pan.2017.43.