Measuring Progress Towards SDG Goal 8

Queena, Yong, Shyanne, Roman, and Margaux

2025-12-10

```
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
```

Preparing all required packages for our analysis. If error occurs, install.packages function should be used to install missing packages

```
library(tidyverse)
library(sf)
library(rnaturalearth)
library(rnaturalearthdata)
library(viridis)
library(dplyr)
library(ggplot2)
library(esquisse)
library(here)
library(scales)
```

Opening the data sets. In order to open them, make sure that they are in the data sets folder.

```
continents <- read.csv(here("data sets", "continents-according-to-our-world-in-data.csv"))
gdp_per_capita <- read.csv(here("data sets", "gdp-per-capita-worldbank.csv"))
youth_neet <- read.csv(here("data sets", "youth-not-in-education-employment-training.csv"))
K2 <- read.csv(here("data sets", "K2.csv"))
```

Cleaning the data from na values and non-countries in Entity column

```
gdp_per_capita <- gdp_per_capita %>%
  filter(Code != "", !is.na(Code))

gdp_per_capita <- gdp_per_capita %>%
    filter(!Entity %in% c("World",
                          "European Union (27)",
                          "East Asia and Pacific (WB)",
                          "Europe and Central Asia (WB)",
                          "Latin America and Caribbean (WB)",
                          "Middle East and North Africa (WB)",
                          "North America (WB)",
                          "South Asia (WB)",
                          "Sub-Saharan Africa (WB)",
                          "High-income countries",
                          "Low-income countries",
```

```
                          "Lower-middle-income countries",
                          "Middle-income countries",
                          "Upper-middle-income countries"))
```

Joining gdp per capita with population and continents data set

```
gdp_pop <- gdp_per_capita %>%
  left_join(
    K2 %>% select(Entity, Code, Year, Population),
    by = c("Entity", "Code", "Year")
  )

gdp_pop <- left_join(gdp_pop, continents, by = "Code", "Entity" )
```

Renaming columns for clearness

```
gdp_per_capita <- gdp_per_capita %>%
  rename(GDP_pc  = `GDP.per.capita..PPP..constant.2017.international...`)

gdp_pop <- gdp_pop %>%
  rename(GDP_pc  = `GDP.per.capita..PPP..constant.2017.international...`)

gdp_pop <- gdp_pop %>%
    select(-Year.y) %>%
  rename(Year = Year.x) %>%
  select(-Entity.y) %>%
  rename(Entity = Entity.x)
```

Scaling the data using population

```
gdp_pop_scaled <- gdp_pop %>%
  group_by(Continent, Year) %>%
  mutate(
    continent_pop = sum(Population, na.rm = TRUE)
  )
```

Calculate each country's population share within its continent year group

```
gdp_pop_scaled <- gdp_pop_scaled %>%
  mutate(
    pop_share = Population / continent_pop
  )
```

Keep the data from 2000 to 2021 only, as it makes analysis easier and more reliable

```
gdp_pop_scaled <- gdp_pop_scaled %>%
  filter(Year >= 2000, Year <= 2021)
```

Calculating population weighted continental GDP per capita for each year as well as annual GDP per capita
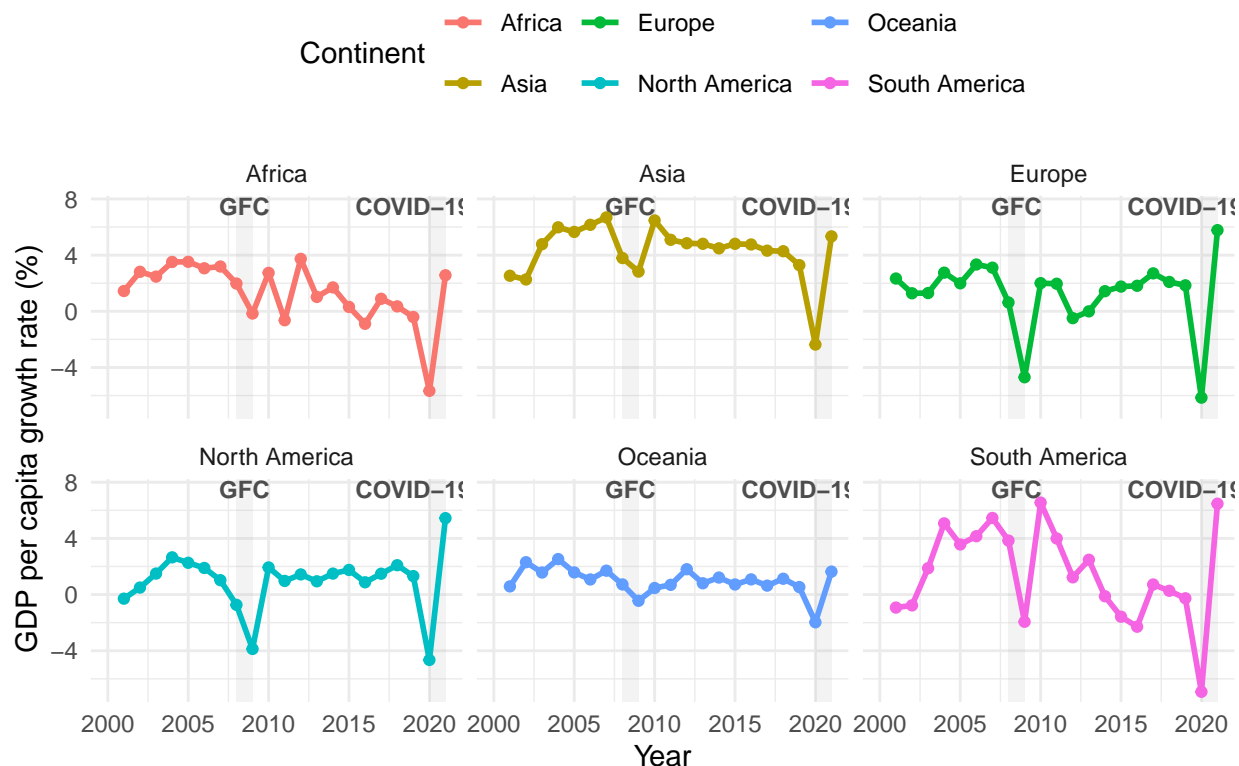growth for each continent

```r
continent_gdp <- gdp_pop_scaled %>%
  group_by(Continent, Year) %>%
  summarise(
    GDP_pc_weighted = sum(GDP_pc * pop_share, na.rm = TRUE),
    .groups = "drop"
  )

continent_growth <- continent_gdp %>%
  arrange(Continent, Year) %>%
  group_by(Continent) %>%
  mutate(
    GDP_pc_growth = (GDP_pc_weighted - lag(GDP_pc_weighted)) /
      lag(GDP_pc_weighted) * 100
  ) %>%
  ungroup()
```

Plotting the line plot for GDP per capita growth by continents from 2000 to 2021

```r
ggplot(continent_growth,
       aes(x = Year,
           y = GDP_pc_growth,
           colour = Continent,
           group = Continent)) +
  annotate("rect",
           xmin = 2008, xmax = 2009,
           ymin = -Inf, ymax = Inf,
           alpha = 0.15, fill = "grey70") +
  annotate("rect",
           xmin = 2020, xmax = 2021,
           ymin = -Inf, ymax = Inf,
           alpha = 0.15, fill = "grey70") +
  annotate("text",
           x = 2008.5, y = 7.5, label = "GFC",
           size = 3, fontface = "bold", color = "grey30") +
  annotate("text",
           x = 2019, y = 7.5, label = "COVID-19",
           size = 3, fontface = "bold", color = "grey30") +
  geom_line(linewidth = 1, na.rm = TRUE) +
  geom_point(size = 1.5, na.rm = TRUE) +
  facet_wrap(~ Continent, ncol = 3)+
  labs(
    title = "GDP per capita growth rate (2000-2021)",
    x = "Year",
    y = "GDP per capita growth rate (%)",
    colour = "Continent"
  ) +
  theme_minimal() +
  theme(
    plot.title      = element_text(hjust = 0.5, face = "bold"),
    legend.position = "top"
  )
```

GDP per capita growth rate (2000–2021)

Creating list of LDC countries

```r
ldc_list <- c(
  # Africa (32)
  "Angola", "Benin", "Burkina Faso", "Burundi", "Central African Republic",
  "Chad", "Comoros", "Democratic Republic of the Congo", "Djibouti", "Eritrea",
  "Ethiopia", "Gambia", "Guinea", "Guinea-Bissau", "Lesotho", "Liberia",
  "Madagascar", "Malawi", "Mali", "Mauritania", "Mozambique", "Niger",
  "Rwanda", "Senegal", "Sierra Leone", "Somalia", "South Sudan", "Sudan",
  "Togo", "Uganda", "United Republic of Tanzania", "Zambia",

  # Asia (8)
  "Afghanistan", "Bangladesh", "Cambodia", "Lao People's Democratic Republic",
  "Myanmar", "Nepal", "Timor-Leste", "Yemen",

  # Caribbean (1)
  "Haiti",

  # Pacific (3)
  "Kiribati", "Solomon Islands", "Tuvalu"
)

ldc_list[ldc_list == "Democratic Republic of the Congo"] <- "Democratic Republic of Congo"
ldc_list[ldc_list == "Lao People's Democratic Republic"] <- "Laos"
ldc_list[ldc_list == "United Republic of Tanzania"] <- "Tanzania"
```

Create a lookup table marking which countries are LDCs

```
ldc_df <- data.frame(
  Entity = ldc_list,
  is_ldc = TRUE
)
```

Testing reliability of GDP per capita data by finding the percentage of missing values by continent and by contry

```
continents_map <- continents %>%
  group_by(Entity, Code) %>%
  summarise(
    Continent = first(na.omit(Continent)),
    .groups = "drop"
  )
```

Joining gdp per capita with continent data set

```
gdp <- gdp_per_capita %>%
  left_join(continents_map, by = c("Entity", "Code"))
```

Clean the data

```
gdp <- gdp %>%
  filter(Code != "", !is.na(Code))

gdp <- gdp %>%
  filter(!Entity %in% c("World",
                        "European Union (27)",
                        "East Asia and Pacific (WB)",
                        "Europe and Central Asia (WB)",
                        "Latin America and Caribbean (WB)",
                        "Middle East and North Africa (WB)",
                        "North America (WB)",
                        "South Asia (WB)",
                        "Sub-Saharan Africa (WB)",
                        "High-income countries",
                        "Low-income countries",
                        "Lower-middle-income countries",
                        "Middle-income countries",
                        "Upper-middle-income countries"))

gdp <- gdp |>
  filter(Year >= 2000, Year <= 2021)
```

Creating a complete panel, so there is no missing year for any country

```
full_panel <- gdp |>
    distinct(Entity, Continent) |>
    tidyr::crossing(Year = 2000:2021)
```

```
gdp_completed <-
    full_panel |>
    left_join(gdp, by = c("Entity", "Continent", "Year"))
```

Calculate the percentage of NA values per country and per continent

```
country_missing_summary <-
  gdp_completed |>
  summarise(
    missing_count = sum(is.na(GDP_pc)),
    total_count   = n(),
    missing_pct   = (missing_count / total_count) * 100,
    .by = c(Entity, Continent)
  )

continent_missing_summary <-
  country_missing_summary |>
  summarise(
    continent_missing_pct = mean(missing_pct),
    .by = Continent
  )

print(continent_missing_summary)
```

```
## # A tibble: 6 x 2
##   Continent      continent_missing_pct
##   <chr>                          <dbl>
## 1 Asia                           0.494
## 2 Europe                         0.974
## 3 Africa                         2.36
## 4 North America                  4.55
## 5 South America                  0
## 6 Oceania                        1.30
```

Testing the reliability of youth NEET data by finding the percentage of missing values by continent and by
contry

Joining youth NEET with continent data set

```
neet <- left_join(youth_neet, continents,
                  by = c("Entity", "Code", "Year"))
```

Renaming the columns and adding a continent to each row

```
neet <- neet %>%
  rename(
    Country   = Entity,
    neet_share = `Share.of.youth.not.in.education..employment.or.training..total....of.youth.population
  )

continent_lookup <- continents %>%
    filter(Year == 2015) %>%
    select(Code, Continent) %>%
    distinct()
```

Cleaning the data

```r
neet <- neet %>%
   select(-Continent) %>%
   left_join(continent_lookup, by = "Code")

countries_neet <-
  neet |>
  distinct(Country, Continent)

years_neet <- 2015:2020
```

Creating a complete panel, so there is no missing year for any country

```r
neet_full_panel <-
  countries_neet |>
  tidyr::crossing(Year = years_neet)

neet_completed <-
  neet_full_panel |>
  left_join(neet, by = c("Country", "Continent", "Year"))
```

Calculating the percentage of NA values by country and by continent

```r
country_neet_missing <-
  neet_completed |>
  summarise(
    missing_count = sum(is.na(neet_share)),
    total_count   = n(),
    missing_pct   = (missing_count / total_count) * 100,
    .by = c(Country, Continent)
  )

continent_neet_missing <-
  country_neet_missing |>
  summarise(
    continent_missing_pct = mean(missing_pct),
    .by = Continent
  )

print(continent_neet_missing[-nrow(continent_neet_missing), ])
```

```
## # A tibble: 6 x 2
##   Continent       continent_missing_pct
##   <chr>                           <dbl>
## 1 Asia                             39.3
## 2 Europe                            4.07
## 3 Africa                           69.6
## 4 South America                    27.8
## 5 North America                    38.6
## 6 Oceania                          72.2
```

Joining gdp per capita with continent data set

```r
gdp <- gdp_per_capita %>%
  left_join(continents_map, by = c("Entity", "Code"))
```

Cleaning the data set

```r
gdp <- gdp %>%
  filter(Code != "", !is.na(Code))

gdp <- gdp %>%
  filter(!Entity %in% c("World",
                        "European Union (27)",
                        "East Asia and Pacific (WB)",
                        "Europe and Central Asia (WB)",
                        "Latin America and Caribbean (WB)",
                        "Middle East and North Africa (WB)",
                        "North America (WB)",
                        "South Asia (WB)",
                        "Sub-Saharan Africa (WB)",
                        "High-income countries",
                        "Low-income countries",
                        "Lower-middle-income countries",
                        "Middle-income countries",
                        "Upper-middle-income countries"))


gdp_class_00_2020 <- gdp %>%
  filter(Year >= 2000, Year <= 2021)
```

Calculating GDP per capita growth

```r
gdp_00_20 <- gdp_class_00_2020 %>%
  filter(Year >= 2000, Year <= 2021)

country_growth <- gdp_00_20 %>%
  arrange(Code, Year) %>%
  group_by(Code, Entity, Continent) %>%
  mutate(
    growth = (GDP_pc - lag(GDP_pc)) / lag(GDP_pc) * 100
  ) %>%
  ungroup()
```

Calculating the average GDP growth over the period for each country

```r
country_mean <- country_growth %>%
  group_by(Code, Entity, Continent) %>%
  summarise(
    mean_growth = mean(growth, na.rm = TRUE),
    .groups = "drop"
  )
```

Applying benchmark by LDC status and flagging countries meeting the benchmarks (4% - non LDCs, 7% - LDCs)

```r
clean_country_mean <- country_mean %>%
  left_join(ldc_df %>% select(Entity, is_ldc), by = "Entity") %>%
  mutate(
    is_ldc      = replace_na(is_ldc, FALSE),    # non-LDCs become FALSE
    threshold   = if_else(is_ldc, 7, 4),
    meets_target = mean_growth >= threshold
  )
```

The proportion of countries meeting target by continent

```r
continent_prop <- clean_country_mean %>%
  group_by(Continent) %>%
  summarise(
    prop_countries_meeting = mean(meets_target),       # TRUE = 1, FALSE = 0
    countries_meeting      = sum(meets_target),
    total_countries        = n(),
    .groups = "drop"
  )
```

Creating a bar chart for proportion of continent that achieved sustained growth

```r
ggplot(continent_prop,
       aes(x = Continent,
           y = prop_countries_meeting,
           fill = Continent)) +
  geom_col() +
  scale_y_continuous(
    labels = percent_format(accuracy = 1),
    limits = c(0, 0.8)              # adjust max if you want more/less headroom
  ) +
  labs(
    title = "Sustained Economic Growth by Continent",
    x = "Continent",
    y = "Percentage (%)",
    fill = "Continent"
  ) +
  theme_minimal() +
  theme(
    plot.title      = element_text(hjust = 0.5, face = "bold"),
    panel.grid.minor = element_blank()
  )
```

Sustained Economic Growth by Continent

Filtering out to keep LDCs only in order to see their average growth

```
ldc_plot <- clean_country_mean %>%
  filter(is_ldc) %>%                          # keep only LDCs
  group_by(Continent) %>%
  arrange(mean_growth, .by_group = TRUE) %>%
  mutate(
    Entity_f = factor(Entity, levels = unique(Entity))  # ordered within continent
  ) %>%
  ungroup()
```
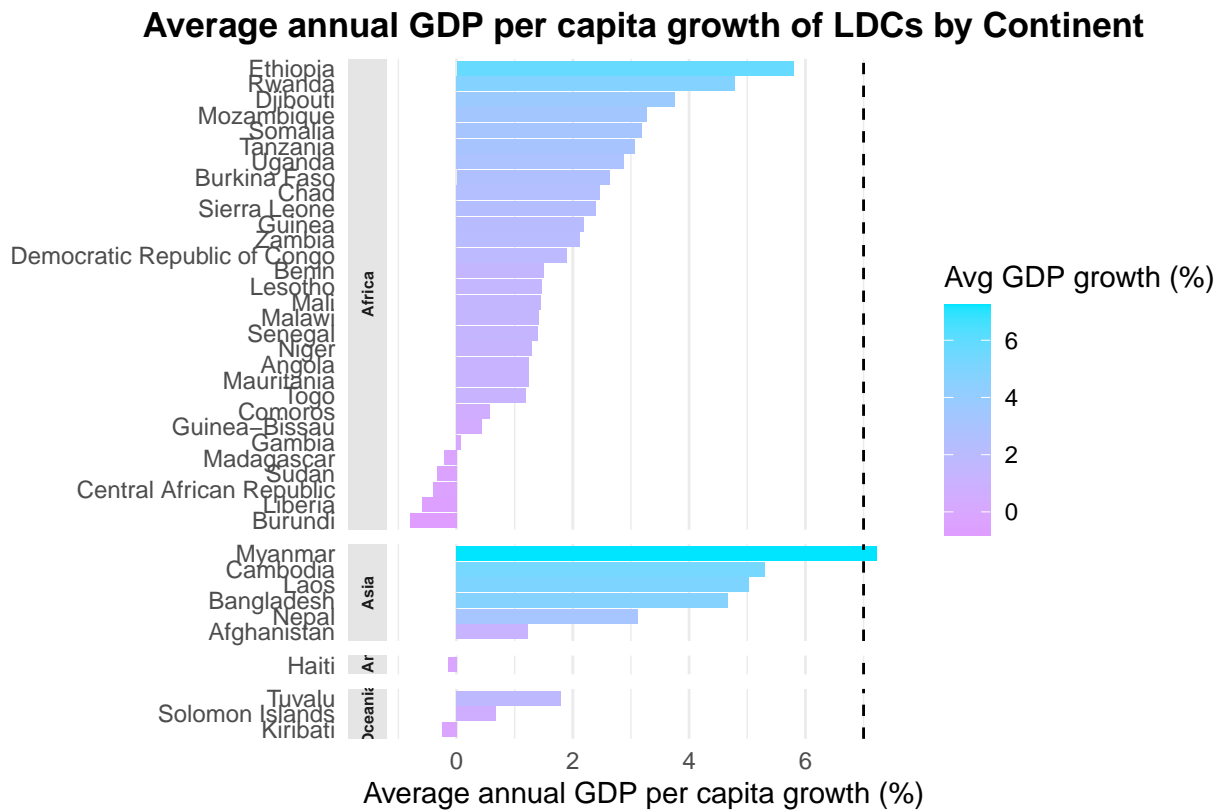
Creating a bar chart for the LDCs only

```
ggplot(ldc_plot,
       aes(x = mean_growth,
           y = Entity_f,
           fill = mean_growth)) +
  geom_col() +
  facet_grid(Continent ~ ., scales = "free_y", space = "free_y", switch = "y") +
  geom_vline(xintercept = 7, linetype = "dashed") +  # 7% SDG target line
  scale_fill_gradient2(
    name = "Avg GDP growth (%)",
    low = "#FF00FF", mid = "#D9A7FF", high = "#00E5FF",
    midpoint = 0
  ) +
  labs(
```

```
    title = "Average annual GDP per capita growth of LDCs by Continent",
    x = "Average annual GDP per capita growth (%)",
    y = NULL,
    caption = "Growth = mean year-on-year % change (2000 - 2020). Dotted line = 7% SDG target for LDCs.
  ) +
  theme_minimal() +
  theme(
    strip.background = element_rect(fill = "grey90", colour = NA),
    strip.text.y = element_text(face = "bold", size = 6),
    panel.grid.major.y = element_blank(),
    plot.title = element_text(hjust = 0.5, face = "bold")
  )
```


Average annual GDP per capita growth of LDCs by Continent

Average annual GDP per capita growth (%)

mean year–on–year % change (2000 – 2020). Dotted line = 7% SDG target for LDCs.

Selecting the youth NEET data in 2015 only

```
neet_2015 <- youth_neet %>%
  filter(Year == 2015) %>%
  rename(NEET = 4) %>%
  select(Entity, Code, NEET)
```

Load World map data keeping only real ISO country codes, and koin the NEET data with the world map.

```
world <- ne_countries(scale = "medium", returnclass = "sf")

valid_iso <- world$iso_a3
```

```r
neet_2015_clean <- neet_2015 %>%
  filter(Code %in% valid_iso)

world_neet <- world %>%
  left_join(neet_2015_clean, by = c("iso_a3" = "Code"))
```
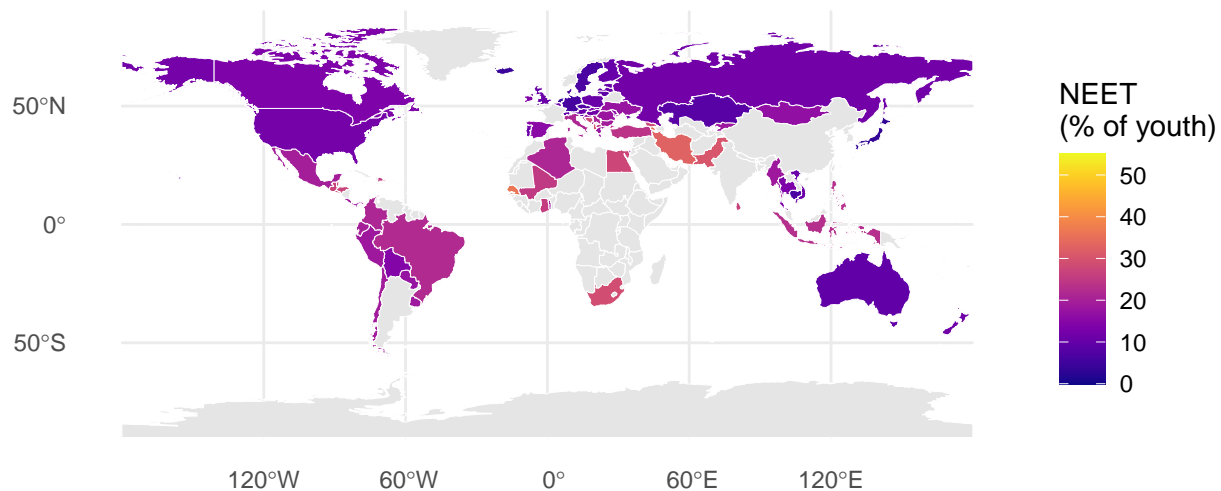
Creating a heatmap in 2015 for NEET

```r
p <- ggplot(world_neet) +
  geom_sf(aes(fill = NEET), color = "white", size = 0.1) +
  scale_fill_viridis(option = "C", na.value = "grey90",
                     name = "NEET\n(% of youth)",
                     limits = c(0, 55)) +
  labs(
    title = "Youth Not in Employment, Education or Training (NEET) - 2015",
    subtitle = "Proportion of youth aged 15-24",
    caption = "Source: OWID / World Bank"
  ) +
  theme_minimal() +
  theme(
    legend.position = "right",
    plot.title = element_text(size = 16, face = "bold"),
    plot.subtitle = element_text(size = 12)
  )

print(p)
```

Youth Not in Employment, Education or Training (NEET) –

Proportion of youth aged 15–24

Source: OWID / World Bank

Now, we need to create the same map, but for 2020 data

Selecting the youth NEET data in 2020 only

```
neet_2020 <- youth_neet %>%
  filter(Year == 2020) %>%
  rename(NEET = 4) %>%
  select(Entity, Code, NEET)
```

Clean the data and join the world map with NEET data

```
valid_iso <- world$iso_a3
neet_2020_clean <- neet_2020 %>%
  filter(Code %in% valid_iso)

world_neet1 <- world %>%
  left_join(neet_2020_clean, by = c("iso_a3" = "Code"))
```

Creating a heatmap in 2020 for NEET

```
p <- ggplot(world_neet) +
  geom_sf(aes(fill = NEET), color = "white", size = 0.1) +
  scale_fill_viridis(option = "C", na.value = "grey90",
                     name = "NEET\n(% of youth)",
                     limits = c(0, 55)) +
  labs(
```
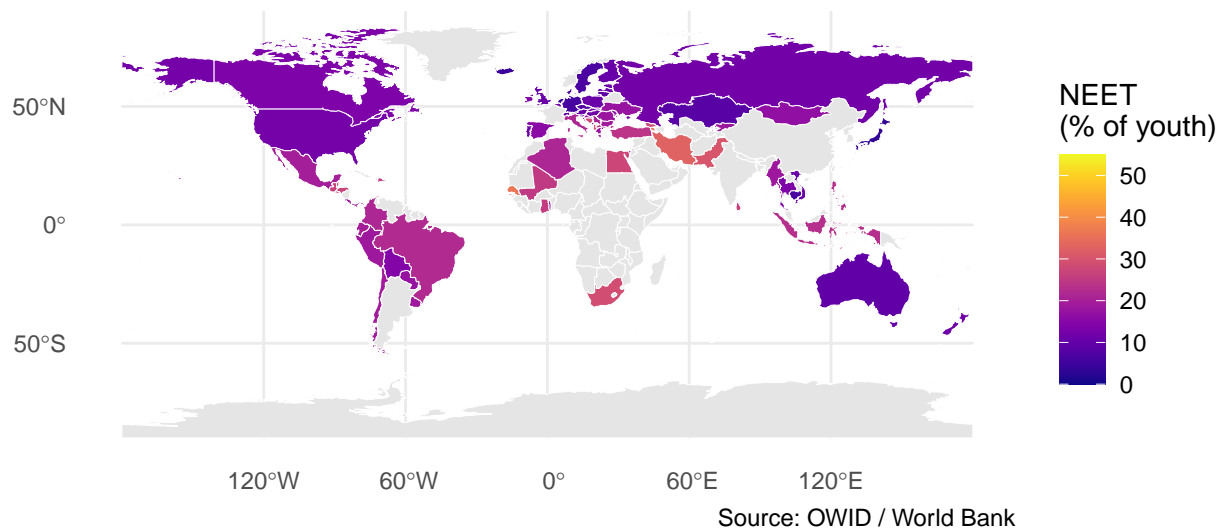
```
    title = "Youth Not in Employment, Education or Training (NEET) - 2020",
    subtitle = "Proportion of youth aged 15-24",
    caption = "Source: OWID / World Bank"
  ) +
  theme_minimal() +
  theme(
    legend.position = "right",
    plot.title = element_text(size = 16, face = "bold"),
    plot.subtitle = element_text(size = 12)
  )

print(p)
```

Youth Not in Employment, Education or Training (NEET) –

Proportion of youth aged 15–24

Source: OWID / World Bank

Cleaning the data

```
neet_both_years <- youth_neet %>%
  filter(Year %in% c(2015, 2020)) %>%
  rename(NEET = 4) %>%
  select(Entity, Code, Year, NEET) %>%
  filter(!is.na(Code), !is.na(NEET))
```

Join continents dataset with NEET values and calculate population-weighted NEET value by continent and year

```r
neet_with_continents <- neet_both_years %>%
  left_join(continents %>% select(Entity, Code, Continent) %>% distinct(),
            by = c("Entity", "Code"))

neet_complete <- neet_with_continents %>%
  left_join(
    K2 %>% select(Entity, Code, Year, Youth_population),
    by = c("Entity", "Code", "Year")
  )

neet_weighted <- neet_complete %>%
  filter(!is.na(Continent), !is.na(Youth_population)) %>%
  group_by(Continent, Year) %>%
  summarise(
    weighted_NEET = sum(NEET * Youth_population) / sum(Youth_population),
    n_countries = n(),
    .groups = "drop"
  ) %>%
  mutate(Year = factor(Year, levels = c("2020", "2015")))
```
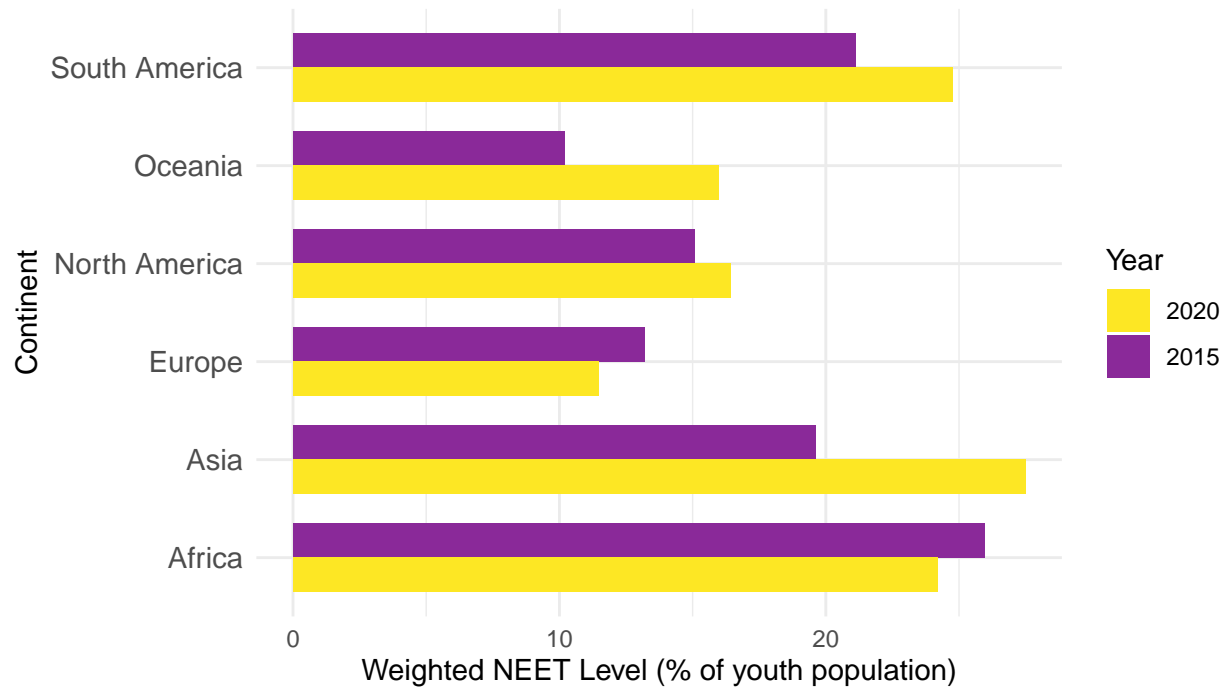
Plot it on the bar chart

```r
p_comparison <- ggplot(neet_weighted, aes(x = Continent, y = weighted_NEET, fill = Year)) +
  geom_col(position = "dodge", width = 0.7) +
  coord_flip() +
  scale_fill_manual(values = c("2015" = "#8a2999", "2020" = "#FDE724"),
                    name = "Year") +
  labs(
    title = "Population-Weighted Youth NEET Level by Continent",
    subtitle = "Comparison between 2015 and 2020",
    x = "Continent",
    y = "Weighted NEET Level (% of youth population)",
    caption = "Source: OWID / World Bank\nWeighted by youth population (ages 15-24)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 14, face = "bold"),
    plot.subtitle = element_text(size = 12),
    axis.text.y = element_text(size = 11),
    legend.position = "right"
  )
print(p_comparison)
```

Population–Weighted Youth NEET Level by Continent

Comparison between 2015 and 2020

Source: OWID / World Bank
Weighted by youth population (ages 15–24)

Find the most recent year in GDP data and filter it for valid rows

```r
latest_year <- max(gdp$Year, na.rm = TRUE)

gdp_latest <- gdp_per_capita %>%
  filter(
    Year == latest_year,
    !is.na(Code),
    !is.na(GDP_pc)
  ) %>%
  transmute(
    iso_a3 = Code,
    gdp_pc = GDP_pc
  )
```

find quantiles

```r
cuts <- quantile(gdp_latest$gdp_pc, probs = c(0.25, 0.50, 0.75), na.rm = TRUE)
```

Filtering the data that we need and joining the data sets

```r
gdp_latest <- gdp_latest %>%
  mutate(
    income_group = case_when(
      gdp_pc <= cuts[1] ~ "Low-income countries",
```

```
      gdp_pc <= cuts[2] ~ "Lower-middle-income countries",
      gdp_pc <= cuts[3] ~ "Upper-middle-income countries",
      TRUE ~ "High-income countries"
    ),
    income_group = factor(
      income_group,
      levels = c(
        "Low-income countries",
        "Lower-middle-income countries",
        "Upper-middle-income countries",
        "High-income countries"
      )
    )
  )

world_income <- world %>%
  left_join(gdp_latest, by = "iso_a3")
```
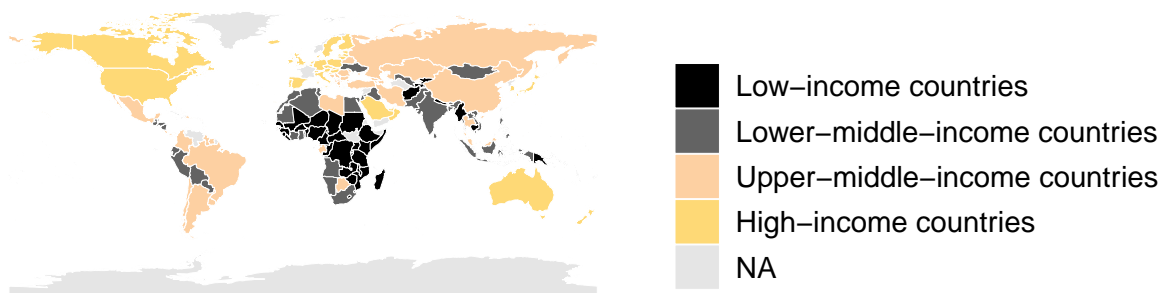
Plotitng the

```
ggplot(world_income) +
  geom_sf(aes(fill = income_group), color = "white", size = 0.12) +
  scale_fill_manual(
    values = c(
      "Low-income countries" = "#000000",
      "Lower-middle-income countries" = "#636363",
      "Upper-middle-income countries" = "#fdd0a2",
      "High-income countries" = "#fed976"
    ),
    na.value = "grey90",
    name = ""
  ) +
  theme_minimal(base_size = 14) +
  theme_minimal(base_size = 12) +
  theme(
    plot.title = element_text(
      size = 16,
      hjust = 0.5,
      margin = margin(b = 10)
    ),
    plot.caption = element_text(
      size = 9,
      hjust = 0.5,
      margin = margin(t = 10)
    ),
    legend.position = "right",
    legend.key.size = unit(0.6, "cm"),
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.title = element_blank(),
    plot.margin = margin(10, 10, 10, 10),
    legend.text = element_text(size = 11)
  ) +
```

```
labs(
  title = paste0("Income Groups Based on GDP per Capita (PPP), ", latest_year),
  caption = "Classification: Quartiles of GDP per capita (PPP)"
)
```

Groups Based on GDP per Capita (PPP), 2021

■ Low–income countries
■ Lower–middle–income countries
■ Upper–middle–income countries
■ High–income countries
■ NA

Classification: Quartiles of GDP per capita (PPP)

Focusing on 2015 to analyse the gender differences in youth NEET

```
continent_lookup <- continents %>%
  filter(Year == 2015) %>%
  select(Code, Continent) %>%
  distinct()
```

Creating a data set with NEET values by gender and continents, focusing on 2015-2020 period

```
neet_continent_year <- K2 %>%
  left_join(continents %>% select(Entity, Continent), by = "Entity") %>%
  group_by(Continent, Year) %>%
  summarise(
    neet_male   = mean(male_NEET,   na.rm = TRUE),
    neet_female = mean(female_NEET, na.rm = TRUE),
    # if you want a total, average of both
    neet_total  = mean(c(male_NEET, female_NEET), na.rm = TRUE),
    .groups = "drop"
  ) %>%
  filter(Year >= 2015, Year <= 2020)
```

Changing the data set from long to wide

```r
neet_2015_2020 <- neet_continent_year %>%
  filter(Year %in% c(2015, 2020)) %>%
  select(Continent, Year, neet_male, neet_female) %>%
  pivot_wider(
    names_from  = Year,
    values_from = c(neet_male, neet_female),
    names_glue  = "{.value}_{Year}"
  )
```

Extracting and renaming for clarity

```r
continent_neet_table <- neet_2015_2020 %>%
  select(
    Continent,
    Boys_2015   = neet_male_2015,
    Girls_2015  = neet_female_2015,
    Boys_2020   = neet_male_2020,
    Girls_2020  = neet_female_2020
  )
```

Drawing the plot for NEET values by gender

```r
neet_long_stacked <- neet_2015_2020 %>%
  pivot_longer(
    cols = c(neet_male_2015, neet_male_2020,
             neet_female_2015, neet_female_2020),
    names_to = c("gender", "year"),
    names_pattern = "neet_(.*)_(.*)",
    values_to = "neet_level"
  ) %>%
  mutate(
    gender = if_else(gender == "male", "Boys", "Girls"),
    year   = as.character(year)
  )

ggplot(neet_long_stacked,
       aes(x = Continent, y = neet_level,
           fill = interaction(gender, year))) +
  geom_col() +
  scale_fill_manual(
    name = "Legend",
    values = c(
      "Boys.2015"  = "yellow",
      "Girls.2015" = "magenta",
      "Boys.2020"  = "cyan",
      "Girls.2020" = "violet"
    ),
    labels = c(
      "Boys 2015",
      "Girls 2015",
      "Boys 2020",
```

```
      "Girls 2020"
   )
) +
labs(
  title = "Youth NEET levels by continent,\ngender and year (2015 vs 2020)",
  x = "Continent",
  y = "NEET (% of youth)"
) +
theme_minimal()
```

Youth NEET levels by continent,
gender and year (2015 vs 2020)