Plots

Lucas Salamuni - 7429674

2025-08-13

Packages

```
packages <- c("dplyr", "knitr", "tinytex", "readxl", "tidyr", "fastDummies",
              "sandwich", "lmtest", "estimatr", "purrr", "tibble", "writexl",
              "ggplot2", "scales", "ggrepel", "sessioninfo")

if(sum(as.numeric(!packages %in% installed.packages())) != 0){
  instalador <- packages[!packages %in% installed.packages()]
  for(i in 1:length(instalador)) {
    install.packages(instalador, dependencies = T)
    break()}
  sapply(packages, require, character = T)
} else {
  sapply(packages, require, character = T)
}
```

```
##       dplyr       knitr     tinytex      readxl       tidyr fastDummies
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE
##    sandwich      lmtest    estimatr       purrr      tibble     writexl
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE
##     ggplot2      scales     ggrepel sessioninfo
##        TRUE        TRUE        TRUE        TRUE
```

Session info

```
session_info()
```

```
## - Session info ---------------------------------------------------------------
##  setting  value
##  version  R version 4.4.1 (2024-06-14 ucrt)
##  os       Windows 11 x64 (build 26100)
##  system   x86_64, mingw32
##  ui       RTerm
##  language (EN)
##  collate  Portuguese_Brazil.utf8
```

```
##  ctype    Portuguese_Brazil.utf8
##  tz       Europe/Berlin
##  date     2025-08-22
##  pandoc   3.4 @ C:/Program Files/RStudio/resources/app/bin/quarto/bin/tools/ (via rmarkdown)
##
## - Packages ------------------------------------------------------------------
##  package      * version  date (UTC) lib source
##  cellranger     1.1.0    2016-07-27 [1] CRAN (R 4.4.1)
##  cli            3.6.3    2024-06-21 [1] CRAN (R 4.4.1)
##  coda           0.19-4.1 2024-01-31 [1] CRAN (R 4.4.1)
##  codetools      0.2-20   2024-03-31 [2] CRAN (R 4.4.1)
##  colorspace     2.1-1    2024-07-26 [1] CRAN (R 4.4.1)
##  digest         0.6.36   2024-06-23 [1] CRAN (R 4.4.1)
##  dplyr        * 1.1.4    2023-11-17 [1] CRAN (R 4.4.1)
##  emmeans        1.10.3   2024-07-01 [1] CRAN (R 4.4.1)
##  estimability   1.5.1    2024-05-12 [1] CRAN (R 4.4.1)
##  estimatr     * 1.0.4    2024-03-31 [1] CRAN (R 4.4.2)
##  evaluate       0.24.0   2024-06-10 [1] CRAN (R 4.4.1)
##  fansi          1.0.6    2023-12-08 [1] CRAN (R 4.4.1)
##  fastDummies  * 1.7.5    2025-01-20 [1] CRAN (R 4.4.3)
##  fastmap        1.2.0    2024-05-15 [1] CRAN (R 4.4.1)
##  Formula        1.2-5    2023-02-24 [1] CRAN (R 4.4.0)
##  generics       0.1.3    2022-07-05 [1] CRAN (R 4.4.1)
##  ggplot2      * 3.5.1    2024-04-23 [1] CRAN (R 4.4.1)
##  ggrepel      * 0.9.5    2024-01-10 [1] CRAN (R 4.4.1)
##  glue           1.7.0    2024-01-09 [1] CRAN (R 4.4.1)
##  gtable         0.3.5    2024-04-22 [1] CRAN (R 4.4.1)
##  htmltools      0.5.8.1  2024-04-04 [1] CRAN (R 4.4.1)
##  httr           1.4.7    2023-08-15 [1] CRAN (R 4.4.1)
##  knitr        * 1.48     2024-07-07 [1] CRAN (R 4.4.1)
##  lattice        0.22-6   2024-03-20 [2] CRAN (R 4.4.1)
##  lifecycle      1.0.4    2023-11-07 [1] CRAN (R 4.4.1)
##  lmtest       * 0.9-40   2022-03-21 [1] CRAN (R 4.4.1)
##  magrittr       2.0.3    2022-03-30 [1] CRAN (R 4.4.1)
##  MASS           7.3-60.2 2024-04-26 [2] CRAN (R 4.4.1)
##  Matrix         1.7-0    2024-04-26 [2] CRAN (R 4.4.1)
##  multcomp       1.4-26   2024-07-18 [1] CRAN (R 4.4.1)
##  munsell        0.5.1    2024-04-01 [1] CRAN (R 4.4.1)
##  mvtnorm        1.2-5    2024-05-21 [1] CRAN (R 4.4.1)
##  pillar         1.9.0    2023-03-22 [1] CRAN (R 4.4.1)
##  pkgconfig      2.0.3    2019-09-22 [1] CRAN (R 4.4.1)
##  purrr        * 1.0.2    2023-08-10 [1] CRAN (R 4.4.1)
##  R6             2.5.1    2021-08-19 [1] CRAN (R 4.4.1)
##  Rcpp           1.0.13   2024-07-17 [1] CRAN (R 4.4.1)
##  readxl       * 1.4.3    2023-07-06 [1] CRAN (R 4.4.1)
##  rlang          1.1.4    2024-06-04 [1] CRAN (R 4.4.1)
##  rmarkdown      2.27     2024-05-17 [1] CRAN (R 4.4.1)
##  rstudioapi     0.16.0   2024-03-24 [1] CRAN (R 4.4.1)
##  sandwich     * 3.1-0    2023-12-11 [1] CRAN (R 4.4.1)
##  scales       * 1.3.0    2023-11-28 [1] CRAN (R 4.4.1)
##  sessioninfo  * 1.2.2    2021-12-06 [1] CRAN (R 4.4.2)
##  survival       3.6-4    2024-04-24 [2] CRAN (R 4.4.1)
##  texreg         1.39.4   2024-07-24 [1] CRAN (R 4.4.1)
##  TH.data        1.1-2    2023-04-17 [1] CRAN (R 4.4.1)
```

```
##  tibble       * 3.2.1    2023-03-20 [1] CRAN (R 4.4.1)
##  tidyr        * 1.3.1    2024-01-24 [1] CRAN (R 4.4.1)
##  tidyselect     1.2.1    2024-03-11 [1] CRAN (R 4.4.1)
##  tinytex      * 0.52     2024-07-18 [1] CRAN (R 4.4.1)
##  utf8           1.2.4    2023-10-22 [1] CRAN (R 4.4.1)
##  vctrs          0.6.5    2023-12-01 [1] CRAN (R 4.4.1)
##  withr          3.0.1    2024-07-31 [1] CRAN (R 4.4.1)
##  writexl      * 1.5.0    2024-02-09 [1] CRAN (R 4.4.1)
##  xfun           0.46     2024-07-18 [1] CRAN (R 4.4.1)
##  xtable         1.8-4    2019-04-21 [1] CRAN (R 4.4.1)
##  yaml           2.3.10   2024-07-26 [1] CRAN (R 4.4.1)
##  zoo          * 1.8-12   2023-04-13 [1] CRAN (R 4.4.1)
##
##  [1] C:/Users/Lucas/AppData/Local/R/win-library/4.4
##  [2] C:/Program Files/R/R-4.4.1/library
##
## ------------------------------------------------------------------------------
```

Part 1. Pure Replication

Plots for the first part.

1.1. Retrieve Part 1's data

```r
# I. Load data from "Final Project.Rmd"'s first part
load("Datasets/replication_results_part1.RData")
```

1.2. Figure 1's replication

```r
df_fig1.1 <- df %>%
  filter(contcod %in% c("BRA", "CHN", "DEU", "IND", "RUS"))

percentile_points <- df_fig1.1 %>%
  filter(group %in% c(1, 25, 50, 75, 100))

deu_min_income <- df %>%
  filter(contcod == "DEU" & group == 1) %>%
  select(inc) %>%
  as.numeric()

economist_colors <- c("BRA" = "#009B3A",
                      "CHN" = "#d5001c",
                      "DEU" = "#ed8b00",
                      "IND" = "#FFD700",
                      "RUS" = "#006ba6")
```

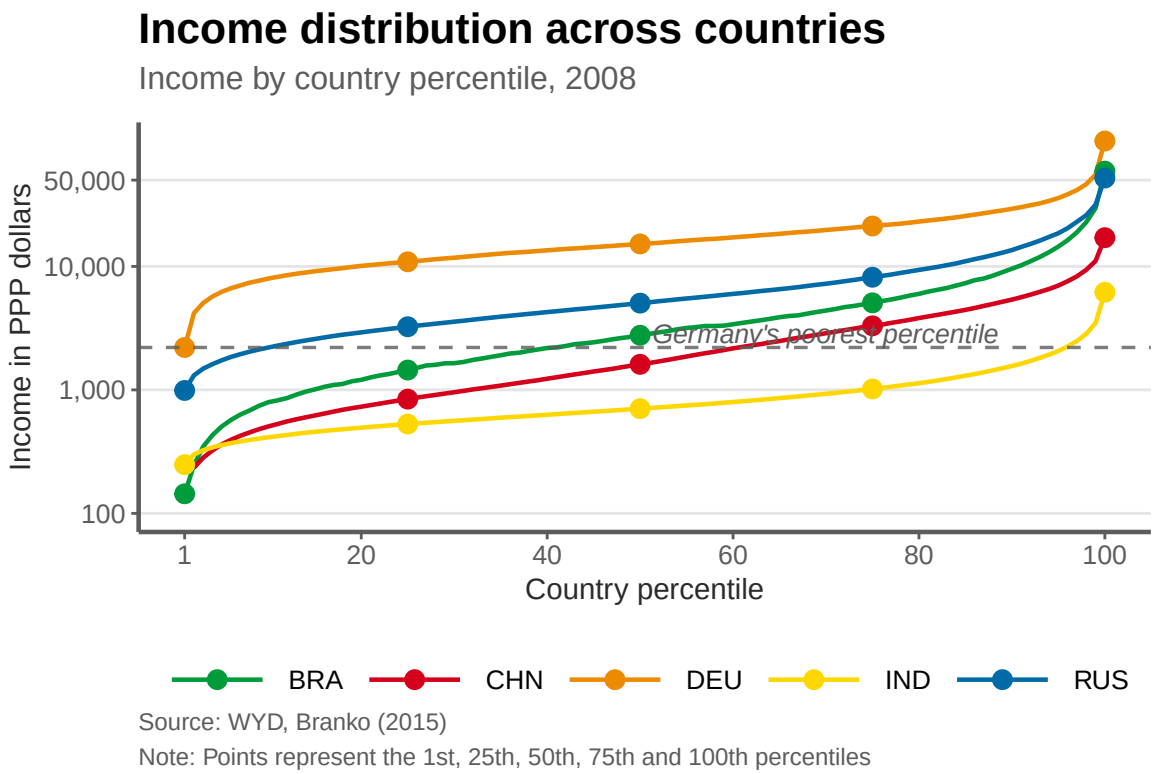
```r
plot_1.1 <- ggplot(data = df_fig1.1,
                   aes(y = inc, x = group,
                       group = contcod,
                       colour = as.factor(contcod))) +
  geom_line(size = 0.8) +
  geom_point(data = percentile_points,
             size = 3,
             alpha = 1) +
  geom_hline(yintercept = deu_min_income,
             linetype = "dashed",
             color = "#5c5c5c",
             size = 0.6,
             alpha = 0.8) +
  scale_y_log10(labels = scales::label_comma(),
                breaks = c(100, 1000, 10000, 50000),
                limits = c(100, NA)) +
  scale_x_continuous(breaks = c(1, 20, 40, 60, 80, 100),
                     limits = c(1, 100)) +
  scale_colour_manual(values = economist_colors) +
  labs(title = "Income distribution across countries",
       subtitle = "Income by country percentile, 2008",
       y = "Income in PPP dollars",
       x = "Country percentile",
       colour = NULL,
       caption = "Source: WYD, Branko (2015) \nNote: Points represent the 1st, 25th, 50th, 75th and 100
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0, lineheight = 1.2),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = "bottom",
        legend.text = element_text(size = 10),
        legend.key.height = unit(0.5, "cm"),
        legend.key.width = unit(1.5, "cm"),
        legend.margin = margin(t = 10),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20)) +
  annotate("text",
           x = 70,
           y = deu_min_income * 1.3,
           label = "Germany's poorest percentile",
           size = 3.5,
           color = "#5c5c5c",
           fontface = "italic")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
```

```
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.

## Warning: The ‘size‘ argument of ‘element_line()‘ is deprecated as of ggplot2 3.4.0.
## i Please use the ‘linewidth‘ argument instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
print(plot_1.1)
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (‘geom_line()‘).

## Warning: Removed 1 row containing missing values or values outside the scale range
## (‘geom_point()‘).
```




```
ggsave(filename = "Plots/plot_1.1.png",
       plot = plot_1.1,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 1 row containing missing values or values outside the scale range
## (`geom_point()`).
```

Part 2. Pure Replication

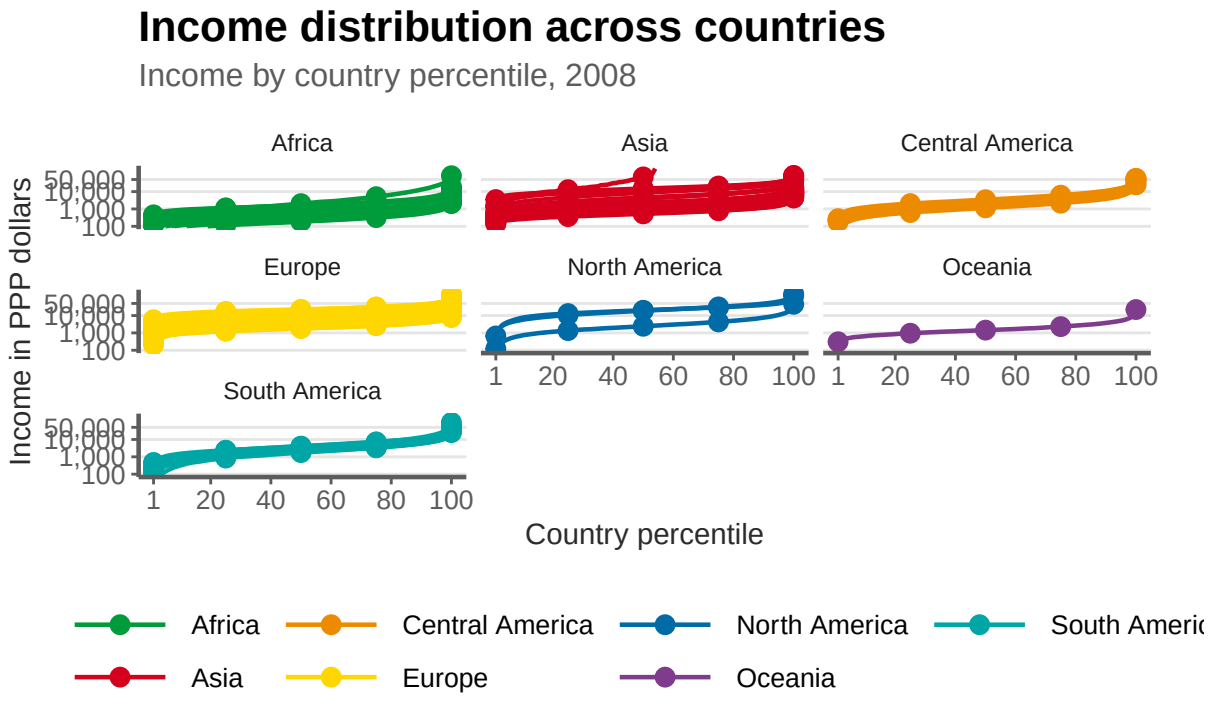
2.1. Continent *facet_wrap* comparison

```
df_fig2.1 <- read_excel(path = "Datasets/WYD_reg.xlsx")

percentile_points <- df_fig2.1 %>%
  filter(group %in% c(1, 25, 50, 75, 100))

economist_colors <- c("Africa" = "#009B3A",
                      "Asia" = "#D5001C",
                      "Central America" = "#ED8B00",
                      "Europe" = "#FFD700",
                      "North America" = "#006BA6",
                      "Oceania" = "#7F3C8D",
                      "South America" = "#00A6A6")

plot_2.1 <- ggplot(data = df_fig2.1,
                   aes(y = inc, x = group,
                       group = contcod,
                       colour = as.factor(reg))) +
  geom_line(size = 0.8) +
  geom_point(data = percentile_points,
             size = 3,
             alpha = 1) +
  facet_wrap(facets = "reg") +
  scale_y_log10(labels = scales::label_comma(),
                breaks = c(100, 1000, 10000, 50000),
                limits = c(100, NA)) +
  scale_x_continuous(breaks = c(1, 20, 40, 60, 80, 100),
                     limits = c(1, 100)) +
  scale_colour_manual(values = economist_colors) +
  labs(title = "Income distribution across countries",
       subtitle = "Income by country percentile, 2008",
       y = "Income in PPP dollars",
       x = "Country percentile",
       colour = NULL,
       caption = "Source: WYD, Branko (2015) \nNote: Points represent the 1st, 25th, 50th, 75th and 100
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0, lineheight = 1.2),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
```

```
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = "bottom",
        legend.text = element_text(size = 10),
        legend.key.height = unit(0.5, "cm"),
        legend.key.width = unit(1.5, "cm"),
        legend.margin = margin(t = 10),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20))

print(plot_2.1) # graph looks too "polluted" and barely offers any insights (probably will not be using
```

```
## Warning: Removed 72 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 23 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



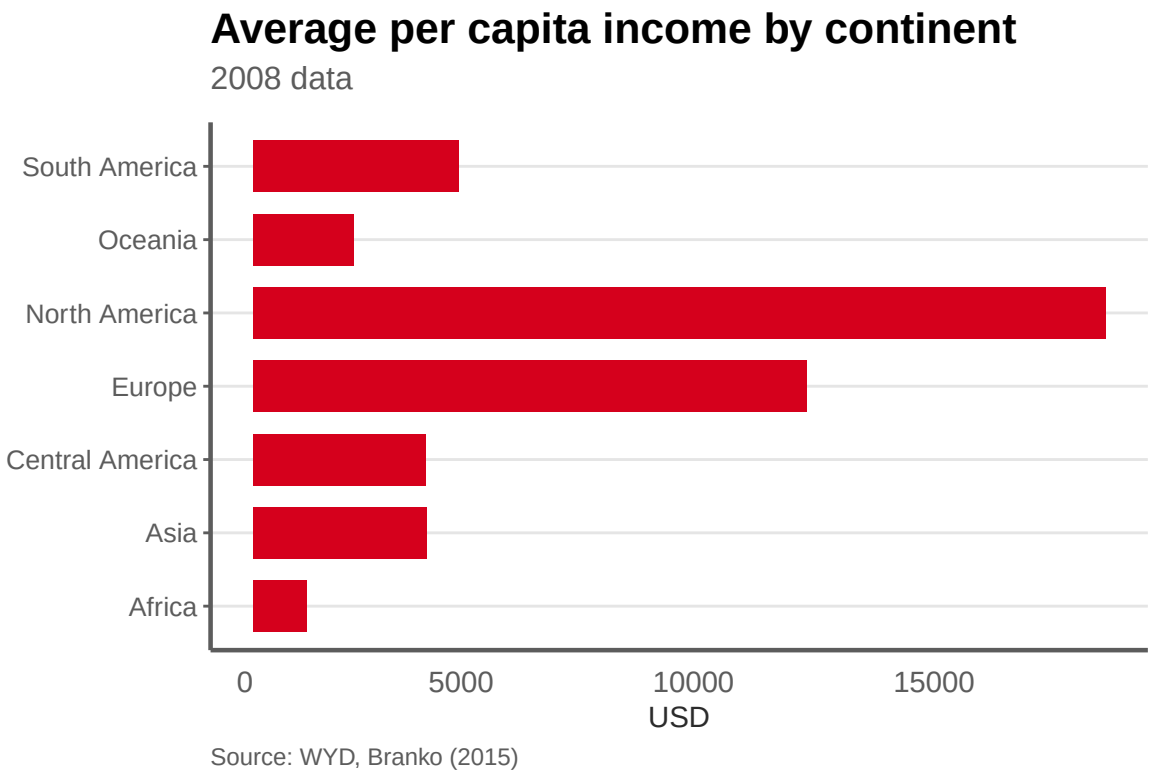
2.2. Cross-continent comparison I (income, population, gdp, gini)

```
df_fig2.2 <- read_excel(path = "Datasets/WYD_cont.xlsx")

df_fig2.2 <- df_fig2.2 %>%
  group_by(reg) %>%
  summarise(avg_inc = mean(inc),
            pop = mean(pop),
            gdpppp = mean(gdpppp),
            gini = mean(gini))

head(df_fig2.2)
```

```
## # A tibble: 6 x 5
##   reg             avg_inc      pop gdpppp  gini
##   <chr>             <dbl>    <dbl>  <dbl> <dbl>
## 1 Africa            1116.  7.08     1996. 0.420
## 2 Asia              3615. 38.2      7236. 0.361
## 3 Central America   3584.  0.507    6641  0.522
## 4 Europe           11499.  7.35    24424. 0.306
## 5 North America    17710.  4.44    30863. 0.430
## 6 Oceania           2099.  0.00843  4300. 0.428
```

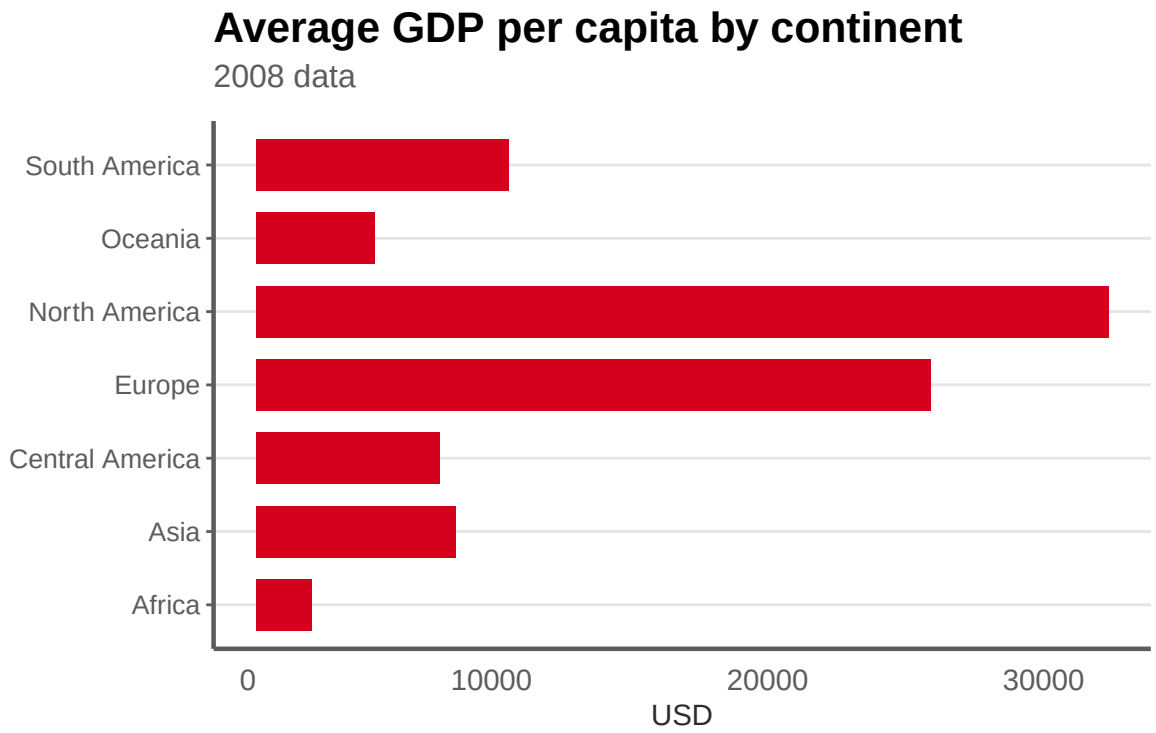
```
economist_colors <- c("Africa" = "#d5001c",
                      "Asia" = "#d5001c",
                      "Central America" = "#d5001c",
                      "Europe" = "#d5001c",
                      "North America" = "#d5001c",
                      "Oceania" = "#d5001c",
                      "South America" = "#d5001c")


# Plot 1
plot_2.2.inc <- ggplot(data = df_fig2.2,
                   aes(x = avg_inc,
                       y = reg,
                       fill = as.factor(reg))) +
  geom_col(width = 0.7) +
  scale_fill_manual(values = economist_colors) +
  labs(title = "Average per capita income by continent",
       subtitle = "2008 data",
       y = NULL,
       x = "USD",
       fill = "Continent",
       caption = "Source: WYD, Branko (2015)") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, hjust = 1, vjust = 1, margin = margin(t = 5)),
```

```
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "none")

print(plot_2.2.inc)
```



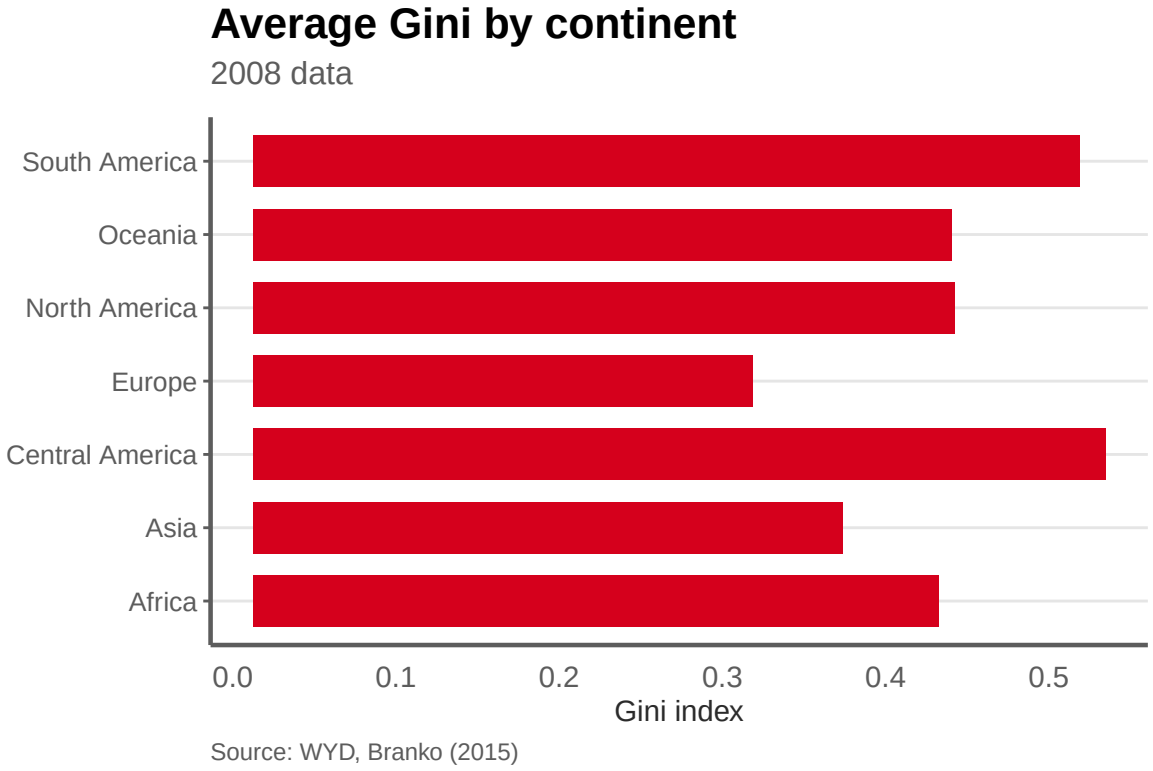

```
ggsave(filename = "Plots/plot_2.2.inc.png",
       plot = plot_2.2.inc,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")


# Plot 2
plot_2.2.gdp <- ggplot(data = df_fig2.2,
                       aes(x = gdpppp,
                           y = reg,
```

```
                      fill = as.factor(reg))) +
  geom_col(width = 0.7) +
  scale_fill_manual(values = economist_colors) +
  labs(title = "Average GDP per capita by continent",
       subtitle = "2008 data",
       y = NULL,
       x = "USD",
       fill = "Continent",
       caption = "Source: WYD, Branko (2015)") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, hjust = 1, vjust = 1, margin = margin(t = 5)),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "none")

print(plot_2.2.gdp)
```




```
ggsave(filename = "Plots/plot_2.2.gdp.png",
       plot = plot_2.2.gdp,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")


# Plot 3
plot_2.2.gini <- ggplot(data = df_fig2.2,
                    aes(x = gini,
                        y = reg,
                        fill = as.factor(reg))) +
  geom_col(width = 0.7) +
  scale_fill_manual(values = economist_colors) +
  labs(title = "Average Gini by continent",
       subtitle = "2008 data",
       y = NULL,
       x = "Gini index",
       fill = "Continent",
       caption = "Source: WYD, Branko (2015)") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
```

```
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, hjust = 1, vjust = 1, margin = margin(t = 5)),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "none")

print(plot_2.2.gini)
```




```
ggsave(filename = "Plots/plot_2.2.gini.png",
       plot = plot_2.2.gini,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```

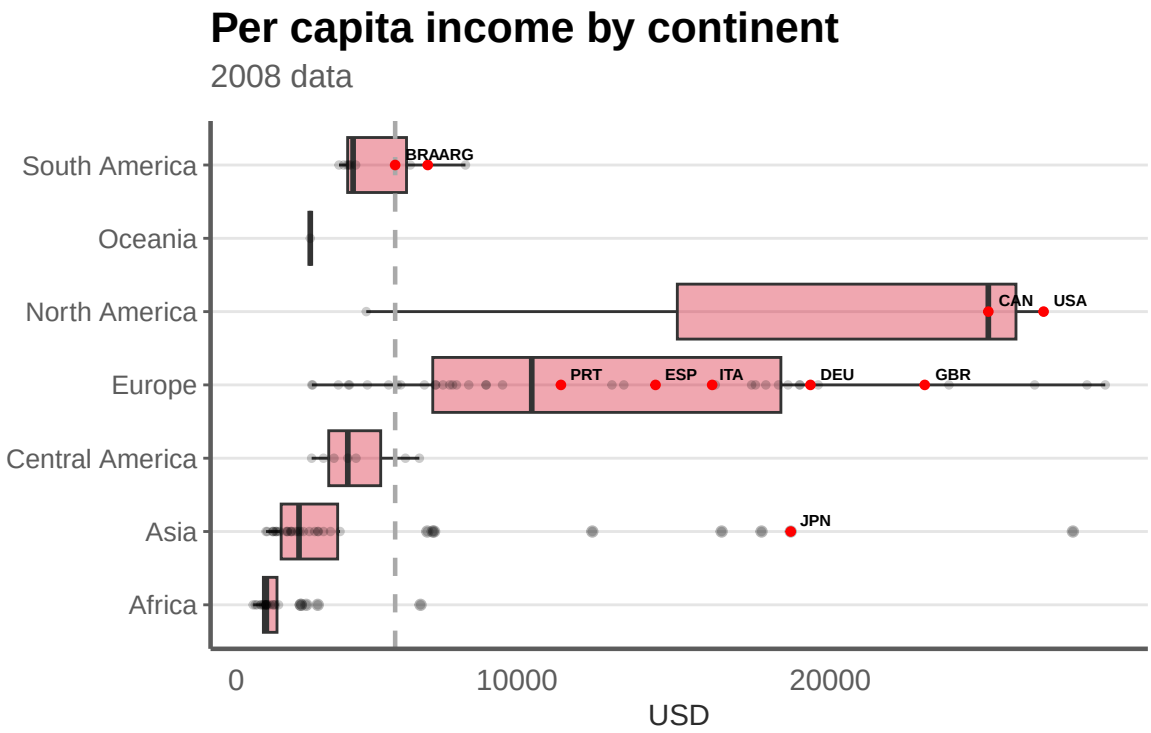
2.3. Cross-continent comparison II (income, population, gdp, gini)

```
df_fig2.3 <- read_excel(path = "Datasets/WYD_reg.xlsx")

df_fig2.3 <- df_fig2.3 %>%
  group_by(contcod, cont, reg) %>%
  summarise(inc = mean(inc),
            pop = sum(pop),
            gdpppp = mean(gdpppp, na.rm = TRUE),
            gini = mean(gini, na.rm = TRUE)) %>%
  arrange(reg)
```

```
## `summarise()` has grouped output by 'contcod', 'cont'. You can override using
## the `.groups` argument.
```

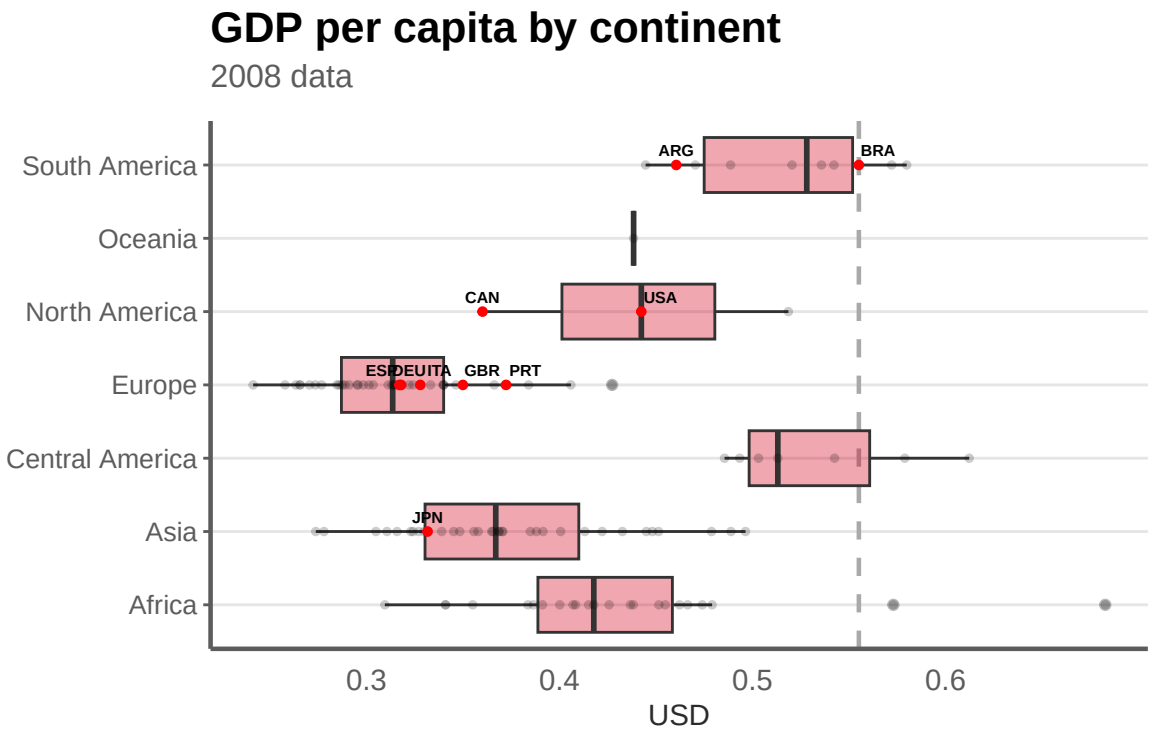
```
economist_colors <- c("Africa" = "#d5001c",
                      "Asia" = "#d5001c",
                      "Central America" = "#d5001c",
                      "Europe" = "#d5001c",
                      "North America" = "#d5001c",
                      "Oceania" = "#d5001c",
                      "South America" = "#d5001c")

top_cont <- c("USA", "PAR", "JPN", "GBR", "PRT", "ITA", "ESP", "DEU", "ARG", "CAN", "BRA")

df_top <- df_fig2.3 %>%
  filter(contcod %in% top_cont)

bra_income <- df_top %>%
  filter(contcod == "BRA") %>%
  pull(inc)

bra_gini <- df_top %>%
  filter(contcod == "BRA") %>%
  pull(gini)

# Plot 1
plot_2.3.inc <- ggplot(df_fig2.3, aes(x = reg,
                                      y = inc,
                                      fill = as.factor(reg))) +
  geom_boxplot(alpha = 0.35) +
  geom_point(alpha = 0.2, size = 1) +
  geom_hline(yintercept = bra_income,
             linetype = "dashed",
             color = "darkgray",
             size = 0.8) +
  geom_point(data = df_top,
             aes(x = reg, y = inc),
             color = "red",
             size = 1.2) +
  geom_text(data = df_top,
            aes(x = reg, y = inc, label = contcod),
```

```
            color = "black",
            size = 2.2,
            fontface = "bold",
            hjust = -0.3,
            nudge_x = 0.15) +
  coord_flip() +
  scale_fill_manual(values = economist_colors) +
  labs(title = "Per capita income by continent",
       subtitle = "2008 data",
       y = "USD",
       x = NULL,
       fill = "Continent",
       caption = "Source: WYD, Branko (2015)") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, hjust = 1, vjust = 1, margin = margin(t = 5)),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "none")

print(plot_2.3.inc)
```




```
ggsave(filename = "Plots/plot_2.3.inc.png",
       plot = plot_2.3.inc,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")

# Plot 2
df_top_labels <- df_top %>%
  mutate(nudge_y_custom = case_when(contcod == "USA" ~ 0.01,
                                    contcod == "PRT" ~ 0.01,
                                    contcod == "ESP" ~ -0.01,
                                    contcod == "ITA" ~ 0.01,
                                    contcod == "DEU" ~ 0.005,
                                    contcod == "GBR" ~ 0.01,
                                    contcod == "BRA" ~ 0.01,
                                    TRUE ~ 0))

plot_2.3.gini <- ggplot(df_fig2.3, aes(x = reg,
                                       y = gini,
                                       fill = as.factor(reg))) +
  geom_hline(yintercept = bra_gini,
             linetype = "dashed",
             color = "darkgray",
             size = 0.8) +
  geom_boxplot(alpha = 0.35) +
```

```r
  geom_point(alpha = 0.2, size = 1) +
  geom_point(data = df_top,
             aes(x = reg, y = gini),
             color = "red",
             size = 1.2) +
  geom_text(data = df_top_labels,
            aes(x = reg,
                y = gini + nudge_y_custom,
                label = contcod),
            color = "black",
            size = 2.2,
            fontface = "bold",
            nudge_x = 0.2) +
  coord_flip() +
  scale_fill_manual(values = economist_colors) +
  labs(title = "GDP per capita by continent",
       subtitle = "2008 data",
       y = "USD",
       x = NULL,
       fill = "Continent",
       caption = "Source: WYD, Branko (2015)") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, hjust = 1, vjust = 1, margin = margin(t = 5)),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "none")

print(plot_2.3.gini)
```

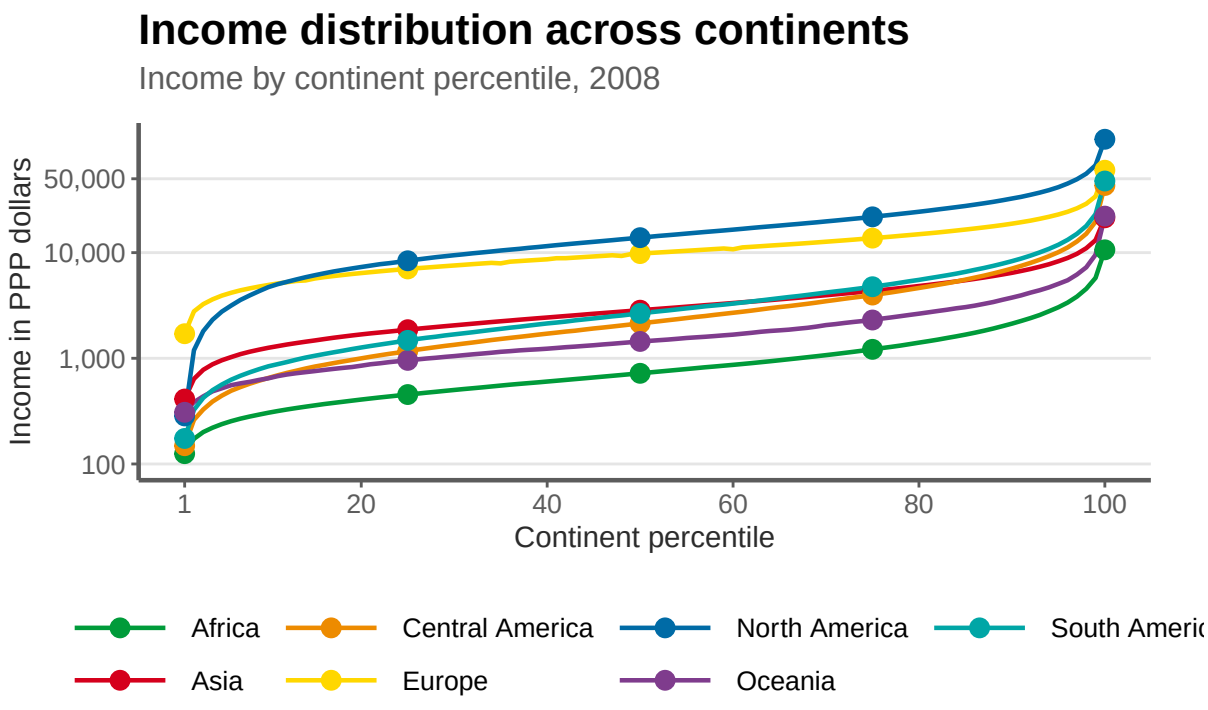



```
ggsave(filename = "Plots/plot_2.3.gdp.png",
       plot = plot_2.3.gini,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```

2.4. Income distribution across continents

```
df_fig2.4 <- read_excel(path = "Datasets/WYD_cont.xlsx")

percentile_points <- df_fig2.4 %>%
  filter(group %in% c(1, 25, 50, 75, 100))

economist_colors <- c("Africa" = "#009B3A",
                      "Asia" = "#D5001C",
                      "Central America" = "#ED8B00",
                      "Europe" = "#FFD700",
                      "North America" = "#006BA6",
                      "Oceania" = "#7F3C8D",
                      "South America" = "#00A6A6")

plot_2.4 <- ggplot(data = df_fig2.4,
```

```
                  aes(y = inc, x = group,
                      group = reg,
                      colour = as.factor(reg))) +
  geom_line(size = 0.8) +
  geom_point(data = percentile_points,
             size = 3,
             alpha = 1) +
  scale_y_log10(labels = scales::label_comma(),
                breaks = c(100, 1000, 10000, 50000),
                limits = c(100, NA)) +
  scale_x_continuous(breaks = c(1, 20, 40, 60, 80, 100),
                     limits = c(1, 100)) +
  scale_colour_manual(values = economist_colors) +
  labs(title = "Income distribution across continents",
       subtitle = "Income by continent percentile, 2008",
       y = "Income in PPP dollars",
       x = "Continent percentile",
       colour = NULL,
       caption = "Source: WYD, Branko (2015) \nNote: Points represent the 1st, 25th, 50th, 75th and 100
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0, lineheight = 1.2),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = "bottom",
        legend.text = element_text(size = 10),
        legend.key.height = unit(0.5, "cm"),
        legend.key.width = unit(1.5, "cm"),
        legend.margin = margin(t = 10),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20))

print(plot_2.4)
```




```
ggsave(filename = "Plots/plot_2.4.inc.png",
       plot = plot_2.4,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```

Part 3. A closer look at Brazil

3.1. Top 10 destinations for Brazilian migrants (2008)

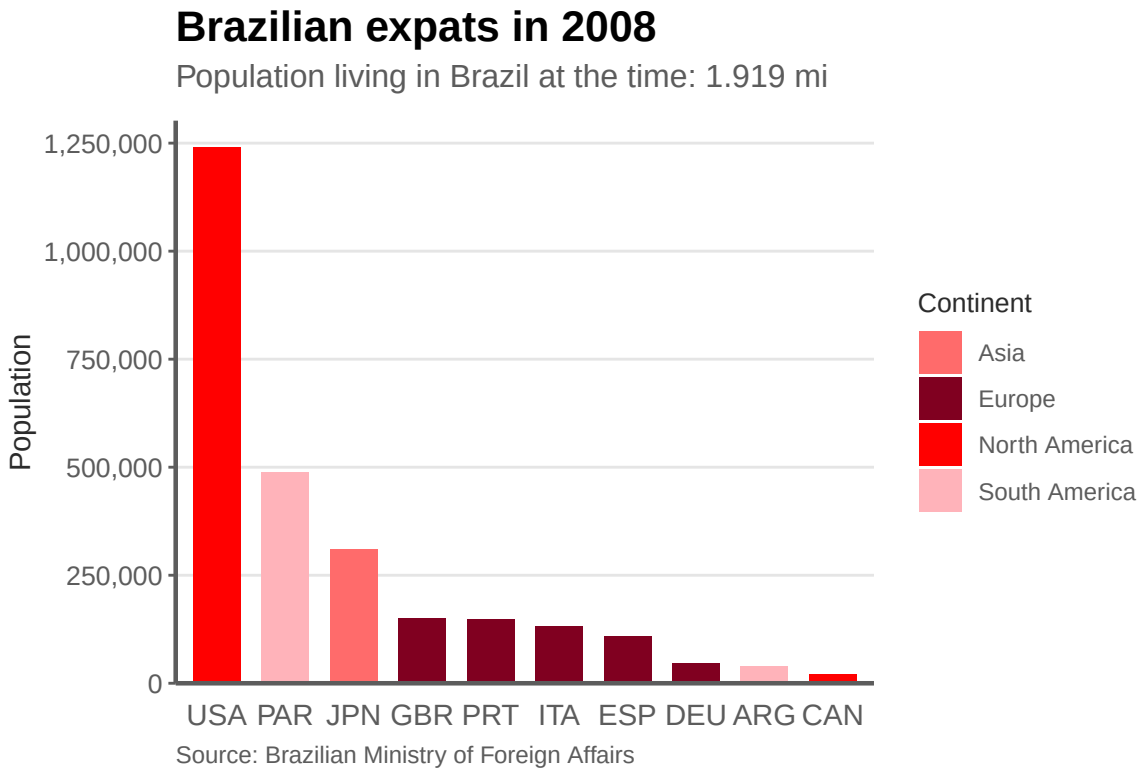
```
bra_pop <- df %>%
  filter(contcod == "BRA") %>%
  group_by(contcod) %>%
  summarise(pop = sum(pop)*1000000) %>%
  select(pop) %>%
  as.numeric()

mig <- tibble("contcod" = c("USA", "PRT", "GBR", "DEU", "ESP", "ITA", "CAN", "JPN", "ARG", "PAR"),
              "reg" = c("North America", "Europe", "Europe", "Europe", "Europe", "Europe", "North Americ
```

```
              "qtd" = c(1240000, 147500, 150000, 46200, 110000, 132000, 20650, 310000, 38500, 487500))

print(mig)
```

```
## # A tibble: 10 x 3
##    contcod reg              qtd
##    <chr>   <chr>          <dbl>
##  1 USA     North America 1240000
##  2 PRT     Europe         147500
##  3 GBR     Europe         150000
##  4 DEU     Europe          46200
##  5 ESP     Europe         110000
##  6 ITA     Europe         132000
##  7 CAN     North America   20650
##  8 JPN     Asia           310000
##  9 ARG     South America   38500
## 10 PAR     South America  487500
```

```
region_colors <-  c("Africa" = "#4B0000",
                    "Asia" = "#FF6B6B",
                    "Central America" = "#B22222",
                    "Europe" = "#800020",
                    "North America" = "#FF0000",
                    "Oceania" = "#FF6B6B",
                    "South America" = "#FFB3BA")

plot_3.1 <- ggplot(data = mig,
                   aes(x = reorder(contcod, -qtd),
                       y = qtd,
                       fill = as.factor(reg))) +
  geom_col(width = 0.7) +
  scale_fill_manual(values = region_colors) +
  scale_y_continuous(labels = scales::label_comma(),
                     expand = expansion(mult = c(0, 0.05))) +
  labs(title = "Brazilian expats in 2008",
       subtitle = "Population living in Brazil at the time: 1.919 mi",
       y = "Population",
       x = NULL,
       fill = "Continent",
       caption = "Source: Brazilian Ministry of Foreign Affairs") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0),
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.text.x = element_text(size = 11, margin = margin(t = 5)),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        axis.ticks.x = element_blank(),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
```

```
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20),
        legend.position = "right",
        legend.title = element_text(size = 10, color = "#2b2b2b"),
        legend.text = element_text(size = 9, color = "#5c5c5c"))

print(plot_3.1)
```



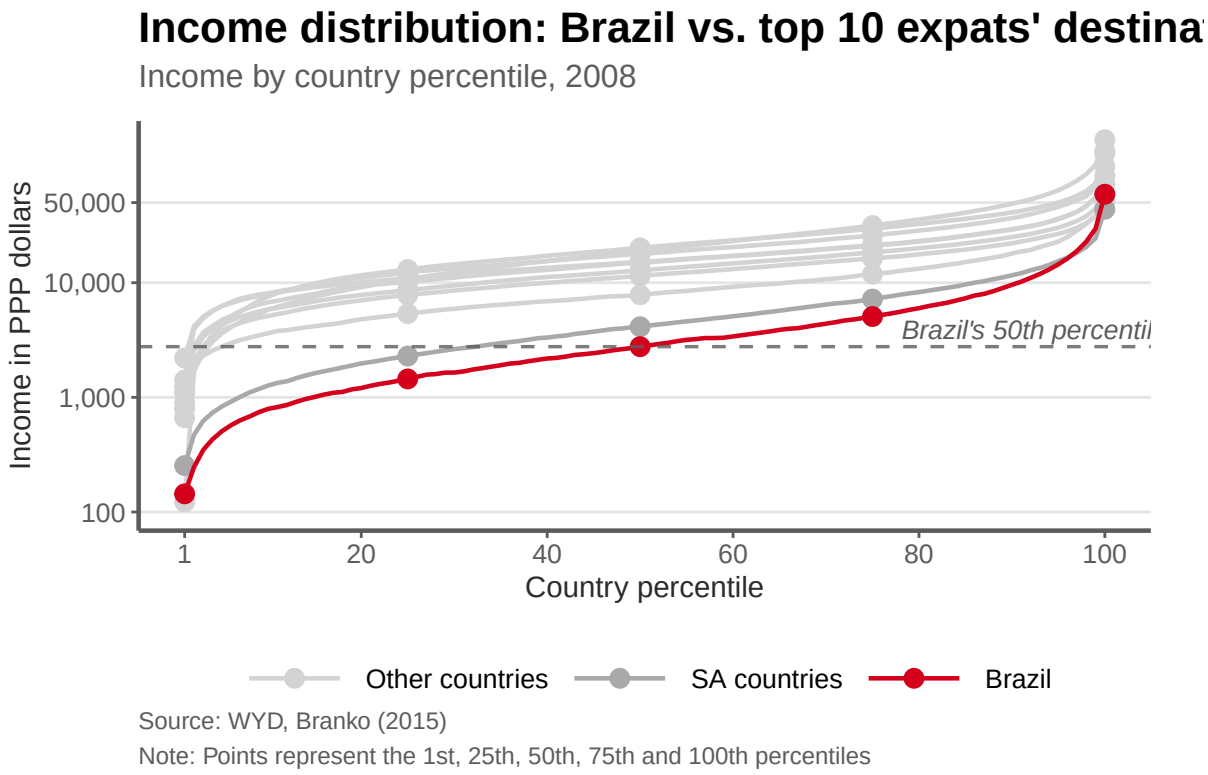

```
ggsave(filename = "Plots/plot_3.1.png",
       plot = plot_3.1,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```

3.2. Income comparison: Brazil vs. top 10 expats' destinations

```
df_fig3.2 <- df %>%
  filter(contcod %in% c("BRA", "USA", "PRT", "GBR", "DEU", "ESP", "ITA", "CAN", "JPN", "ARG", "PAR")) %
  mutate(bra = factor(case_when(contcod == "BRA" ~ "Brazil",
                                contcod %in% c("ARG", "PAR") ~ "SA countries",
                                TRUE ~ "Other countries"),
```

```
                      levels = c("Other countries", "SA countries", "Brazil")))

percentile_points <- df_fig3.2 %>%
  filter(group %in% c(1, 25, 50, 75, 100))

bra_mid_income <- df %>%
  filter(contcod == "BRA" & group == 50) %>%
  pull(inc)

economist_colors <- c("Brazil" = "#d5001c",
                      "SA countries" = "darkgrey",
                      "Other countries" = "lightgrey")

plot_3.2 <- ggplot(data = df_fig3.2,
                   aes(y = inc, x = group,
                       group = contcod,
                       colour = bra)) +
  geom_line(data = filter(df_fig3.2, bra == "Other countries"),
            size = 0.8) +
  geom_point(data = filter(percentile_points, bra == "Other countries"),
             size = 3,
             alpha = 1) +
  geom_line(data = filter(df_fig3.2, bra == "SA countries"),
            size = 0.8) +
  geom_point(data = filter(percentile_points, bra == "SA countries"),
             size = 3,
             alpha = 1) +
  geom_line(data = filter(df_fig3.2, bra == "Brazil"),
            size = 0.8) +
  geom_point(data = filter(percentile_points, bra == "Brazil"),
             size = 3,
             alpha = 1) +
  geom_hline(yintercept = bra_mid_income,
             linetype = "dashed",
             color = "#5c5c5c",
             size = 0.6,
             alpha = 0.8) +
  scale_y_log10(labels = scales::label_comma(),
                breaks = c(100, 1000, 10000, 50000),
                limits = c(100, NA)) +
  scale_x_continuous(breaks = c(1, 20, 40, 60, 80, 100),
                     limits = c(1, 100)) +
  scale_colour_manual(values = economist_colors) +
  labs(title = "Income distribution: Brazil vs. top 10 expats' destinations",
       subtitle = "Income by country percentile, 2008",
       y = "Income in PPP dollars",
       x = "Country percentile",
       colour = NULL,
       caption = "Source: WYD, Branko (2015)\nNote: Points represent the 1st, 25th, 50th, 75th and 100t
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", margin = margin(b = 5)),
        plot.subtitle = element_text(size = 12, color = "#5c5c5c", margin = margin(b = 10)),
        plot.caption = element_text(size = 9, color = "#5c5c5c", hjust = 0, lineheight = 1.2),
```

```
        axis.line = element_line(color = "#5c5c5c", size = 0.8),
        axis.text = element_text(size = 10, color = "#5c5c5c"),
        axis.title = element_text(size = 11, color = "#2b2b2b", face = "plain"),
        axis.ticks = element_line(color = "#5c5c5c", size = 0.5),
        panel.grid.major.y = element_line(color = "#e5e5e5", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = "bottom",
        legend.text = element_text(size = 10),
        legend.key.height = unit(0.5, "cm"),
        legend.key.width = unit(1.5, "cm"),
        legend.margin = margin(t = 10),
        plot.background = element_rect(fill = "white", color = NA),
        panel.background = element_rect(fill = "white", color = NA),
        plot.margin = margin(20, 20, 20, 20)) +
  annotate("text",
           x = 92.5,
           y = bra_mid_income * 1.4,
           label = "Brazil's 50th percentile",
           size = 3.5,
           color = "#5c5c5c",
           fontface = "italic")

print(plot_3.2)
```



```r
ggsave(filename = "Plots/plot_3.2.png",
       plot = plot_3.2,
       width = 10,
       height = 6,
       dpi = 300,
       bg = "white")
```