

Vẽ đồ thị xuất sắc với R

ThS. Nguyễn Tấn Đức

Cập nhật: 03-09-2023

Table of contents

Giới thiệu	3
1 Hướng dẫn vẽ đồ thị hai trục tung	4
2 Vẽ đồ thị histogram	12
Tài liệu tham khảo	13

Giới thiệu

Source code vẽ đồ thị.

1 Hướng dẫn vẽ đồ thị hai trục tung

Tình huống: Bạn có bộ dữ liệu về dân số SP.POP.TOTL và GDP bình quân NY.GDP.PCAP.CD của Việt Nam từ năm 1960 đến năm 2022 như sau. Bạn có thể vẽ hai dãy số liệu này trên cùng một đồ thị khi sử dụng 2 trục tung khác đơn vị đo.

```
# download dataset
# download.file(url = "https://tuhocr.netlify.app/dataset/vn_data_ok.csv",
#               destfile = "vn_data_ok.csv",
#               mode = "wb")

# import dataset
df <- read.csv("vn_data_ok.csv")

# read dataset
knitr::kable(df, "simple")
```

country	year	SP.POP.TOTL	NY.GDP.PCAP.CD
Vietnam	2022	98186856	4163.51430
Vietnam	2021	97468029	3756.48912
Vietnam	2020	96648685	3586.34730
Vietnam	2019	95776716	3491.09128
Vietnam	2018	94914330	3267.22501
Vietnam	2017	94033048	2992.07175
Vietnam	2016	93126529	2760.71710
Vietnam	2015	92191398	2595.23498
Vietnam	2014	91235504	2558.77892
Vietnam	2013	90267739	2367.49954
Vietnam	2012	89301326	2190.23228
Vietnam	2011	88349117	1953.55698
Vietnam	2010	87411012	1684.01167
Vietnam	2009	86482923	1225.84501
Vietnam	2008	85597241	1158.10163
Vietnam	2007	84762269	913.31233
Vietnam	2006	83951800	790.59252
Vietnam	2005	83142095	693.18984
Vietnam	2004	82311227	551.90351
Vietnam	2003	81475825	485.45091

country	year	SP.POP.TOTL	NY.GDP.PCAP.CD
Vietnam	2002	80642308	434.81029
Vietnam	2001	79817777	409.49773
Vietnam	2000	79001142	394.58314
Vietnam	1999	78123713	367.15688
Vietnam	1998	77128424	352.78307
Vietnam	1997	76058603	352.93444
Vietnam	1996	74946448	329.00119
Vietnam	1995	73759110	281.13360
Vietnam	1994	72501087	224.63709
Vietnam	1993	71176405	185.18712
Vietnam	1992	69788747	141.38369
Vietnam	1991	68358820	140.63100
Vietnam	1990	66912613	96.71930
Vietnam	1989	65466361	96.13036
Vietnam	1988	64037514	397.01436
Vietnam	1987	62630787	585.30494
Vietnam	1986	61221107	430.18850
Vietnam	1985	59811313	235.65254
Vietnam	1984	58406863	NA
Vietnam	1983	57011444	NA
Vietnam	1982	55632153	NA
Vietnam	1981	54280394	NA
Vietnam	1980	52968270	NA
Vietnam	1979	51831389	NA
Vietnam	1978	50701458	NA
Vietnam	1977	49418150	NA
Vietnam	1976	48163573	NA
Vietnam	1975	46969616	NA
Vietnam	1974	45898698	NA
Vietnam	1973	44891281	NA
Vietnam	1972	43906019	NA
Vietnam	1971	42916624	NA
Vietnam	1970	41928849	NA
Vietnam	1969	41015865	NA
Vietnam	1968	40145287	NA
Vietnam	1967	39282564	NA
Vietnam	1966	38388210	NA
Vietnam	1965	37466077	NA
Vietnam	1964	36509166	NA
Vietnam	1963	35526727	NA
Vietnam	1962	34533889	NA
Vietnam	1961	33621982	NA
Vietnam	1960	32718461	NA

```

# đổi tên cho gọn hơn
names(df) <- c("country", "year", "pop", "gdp")

# lệnh này giúp vẽ trực tiếp trên graphics device
# windows(width = 8, height = 6)

# xuất ra file ảnh
png(filename = "p1.png", width = 8, height = 6, units = "in", res = 300)

library(ggplot2)
library(extrafont) # hỗ trợ font Times New Roman
library(ggtext) # hỗ trợ format text

scale_truc_y_second <- 45

limit_truc_y_first <- c(0, 120)

# vẽ đồ thị
ggplot(df) +

geom_line(aes(x = year,
              y = pop/1000000,
              colour = "Dân số (triệu người)",
              linetype = "Dân số (triệu người)",
              linewidth = 1) +

geom_line(aes(x = year,
              y = gdp/scale_truc_y_second,
              colour = "GDP bình quân (USD/người)",
              linetype = "GDP bình quân (USD/người)",
              linewidth = 1) +

scale_x_continuous(name = "Năm",
                  expand = c(0, 0),
                  limits = c(1955, 2025),
                  breaks = c(1960, 1970, 1980, 1985, 1990, 2000, 2010, 2022)) +

scale_y_continuous(name = "Dân số (triệu người)", ### customize the first y-axis
                  expand = c(0, 0),
                  limits = limit_truc_y_first, # set limit chung cho 2 trục
                  breaks = round(c(limit_truc_y_first[1],
                                na.omit(df$pop)[1]/1000000,
                                40,
                                60,

```

```

80,
na.omit(df$pop)[length(na.omit(df$pop))]/1000000,
110,
limit_truc_y_first[2]),
digits = 0),

### customize the second y-axis (scale là để điều chỉnh cho phù hợp)
sec.axis = sec_axis(~ . * scale_truc_y_second,
name = "GDP per capita (current US$)",
breaks = round(c(limit_truc_y_first[1] * scale_truc_y_second,
na.omit(df$gdp)[1],
na.omit(df$gdp)[length(na.omit(df$gdp))],
# min(df$gdp, na.rm = TRUE),
1000,
2000,
3000,
4000,
max(df$gdp, na.rm = TRUE),
limit_truc_y_first[2] * scale_truc_y_second),
digits = 0))) +

scale_colour_manual(values = c("red", "blue"),
# name = c("Chú thích"),
labels = c("Dân số (triệu người)", "GDP bình quân (USD/người)"),
guide = guide_legend(title = NULL,
title.position = "top",
byrow = TRUE,
nrow = 1)) +

scale_linetype_manual(values = c("solid", "solid"),
# name = c("Chú thích"),
labels = c("Dân số (triệu người)", "GDP bình quân (USD/người)"),
guide = guide_legend(title = NULL,
title.position = "top",
byrow = TRUE,
nrow = 1)) +

# customize đồ thị
labs(title = "Dân số và GDP bình quân đầu người ở Việt Nam giai đoạn 1960-2022",
subtitle = "Nguồn: WORLD BANK",
caption = "Đồ thị: Duc Nguyen | tuhocr.com") +

# panel và margin

```

```

theme(plot.margin = margin(t = 1, r = 1, b = 1, l = 1, "cm")) +

theme(panel.background = element_rect(fill = "white"),
      panel.border = element_rect(colour = "darkgreen",
                                   fill = NA,
                                   linewidth = 0.75),
      plot.background = element_rect(fill = "#fcfccd",
                                      colour = "black",
                                      linewidth = 0.75)) +

# trục đồ thị
theme(axis.title.x.bottom = element_text(face = "bold",
                                           colour = "black",
                                           size = 11,
                                           family = "Times New Roman",
                                           margin = margin(t = 10, r = 1,
                                                           b = 1, l = 1)),
      axis.title.y.left = element_text(face = "bold",
                                       family = "Times New Roman",
                                       size = 11, colour = "red",
                                       margin = margin(t = 1, r = 5,
                                                       b = 1, l = 1)),
      axis.title.y.right = element_text(face = "bold",
                                         size = 11, colour = "blue",
                                         angle = 90,
                                         family = "Times New Roman")) +

theme(axis.text.x.bottom = element_text(face = "bold",
                                         colour = "black",
                                         size = 10,
                                         family = "Times New Roman"),
      axis.text.y.left = element_text(face = "bold",
                                       colour = "red",
                                       size = 10,
                                       family = "Times New Roman"),
      axis.text.y.right = element_text(face = "bold",
                                       colour = "blue",
                                       size = 10,
                                       family = "Times New Roman")) +

theme(axis.ticks.y.left = element_line(linewidth = 0.75, colour = "red"),
      axis.ticks.y.right = element_line(linewidth = 0.75, colour = "blue"),
      axis.ticks.x.bottom = element_line(linewidth = 0.75, colour = "black"),
      axis.ticks.length = unit(0.25, "cm")) +

```



```

# customize legend
theme(legend.text = element_text(face = "bold",
                                   colour = "black",
                                   size = 10,
                                   family = "Times New Roman")) +

theme(legend.key = element_rect(fill = "transparent",
                                   colour = "transparent"),
      legend.key.size = unit(1.1, "cm"),
      legend.direction = "horizontal",
      legend.justification = c(0.5, 1),
      legend.position = c(0.5, 1),
      legend.background = element_rect(fill = "transparent",
                                         linewidth = 1,
                                         linetype = "solid",
                                         colour = "transparent")) +

# tiêu đề đồ thị
theme(plot.title.position = "plot",
      plot.title = ggtext::element_textbox_simple(size = 12,
                                                    face = "bold",
                                                    family = "Times New Roman",
                                                    # margin: top, right, bottom, left
                                                    padding = margin(t = 5,
                                                                    r = 5,
                                                                    b = 3,
                                                                    l = 100)),
      plot.subtitle = ggtext::element_textbox_simple(size = 11,
                                                       face = "plain",
                                                       family = "Times New Roman",
                                                       padding = margin(t = 5,
                                                                       r = 5,
                                                                       b = 6,
                                                                       l = 100)),
      plot.caption = element_text(size = 10,
                                   hjust = 1,
                                   family = "Times New Roman")) +

## chú thích đồ thị
geom_point(aes(x = 1993, y = 33), colour = "darkgreen") +

annotate(geom = "curve",
         x = 1993,
         y = 33,

```

```

xend = 1990,
yend = 5,
curvature = -0.3,
arrow = arrow(length = unit(0.04, "npc")),
colour = "darkgreen") +

annotate(geom = "text", x = 1988, y = 41,
  label = "Quyết định bước ngoặt:\nMở cửa nền kinh tế",
  hjust = "left",
  colour = "darkgreen",
  fontface = "bold",
  family = "Times New Roman",
  size = 3.5) +

annotate("rect", xmin = 1988, xmax = 1992, ymin = 1, ymax = 10,
  alpha = 0.2, colour = "blue", fill = "red")

## chèn hình minh họa

library(grid)
library(gridExtra)
library(png)

# download.file(url = "https://tuhocr.netlify.app/vn.png",
#               destfile = "vn.png",
#               mode = "wb")

vn_image <- readPNG("vn.png")
grid.raster(vn_image, x = 0.16, y = 0.895, width = 0.08)

# download.file(url = "https://tuhocr.netlify.app/logor.png",
#               destfile = "logor.png",
#               mode = "wb")

logor <- readPNG("logor.png")

logor -> m
w <- matrix(rgb(m[, , 1], m[, , 2], m[, , 3], m[, , 4] * 0.5),
  nrow = dim(m)[1]) # ↑ 0.5 is alpha/opacity

grid.raster(w, x = 0.9, y = 0.1, width = 0.1)

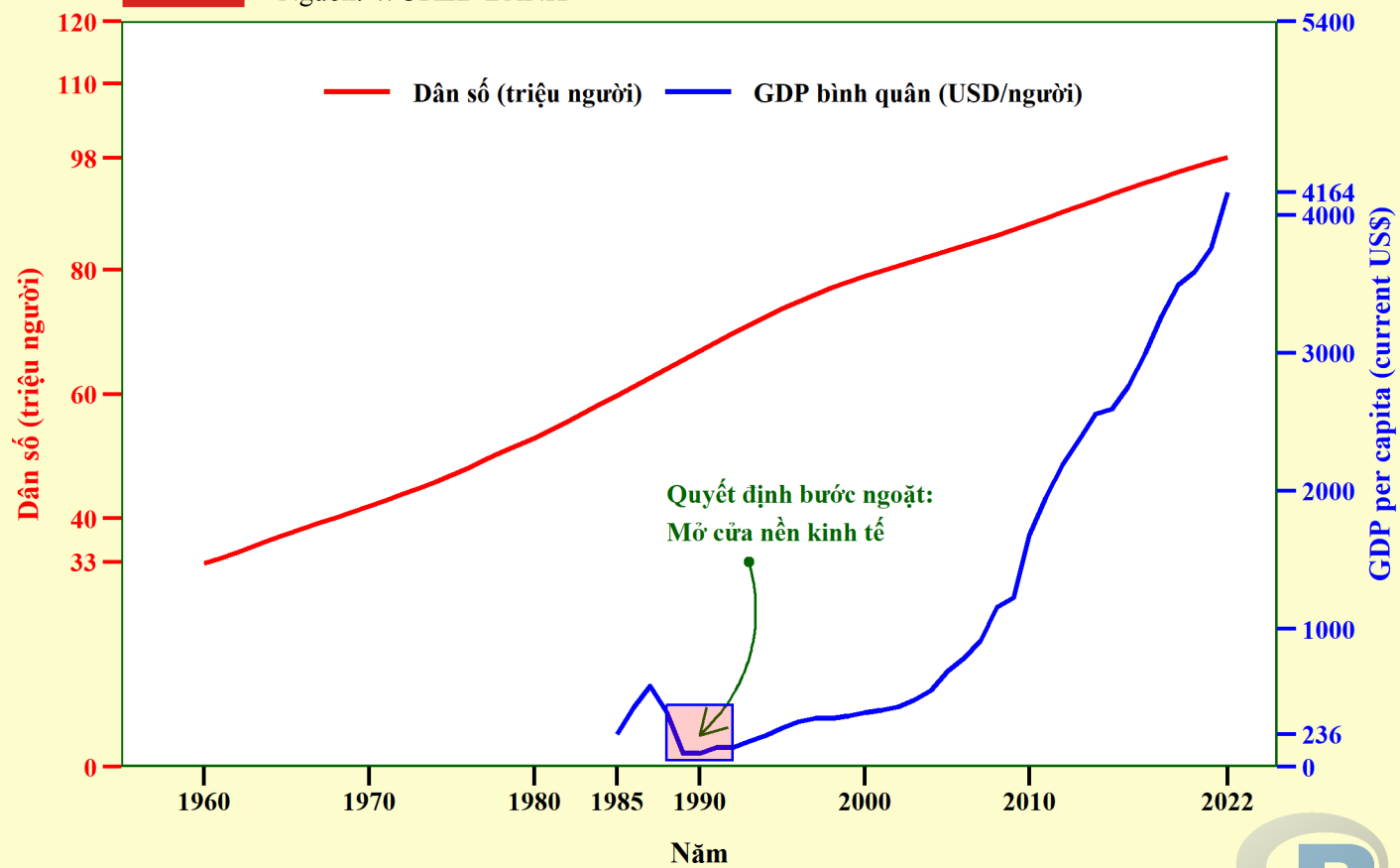
dev.off()

```



Dân số và GDP bình quân đầu người ở Việt Nam giai đoạn 1960–2022

Nguồn: WORLD BANK



Đồ thị: Duc Nguyen | tuhocr.com



2 Vẽ đồ thị histogram

Tài liệu tham khảo