

Package: gapminder (via r-universe)

July 2, 2024

Title Data from Gapminder

Version 1.0.0.9000

Description An excerpt of the data available at Gapminder.org. For each of 142 countries, the package provides values for life expectancy, GDP per capita, and population, every five years, from 1952 to 2007.

License CC0

URL <https://github.com/jennybc/gapminder>,
<https://www.gapminder.org/data/>,
<https://doi.org/10.5281/zenodo.594018>,
<https://jennybc.github.io/gapminder/>

BugReports <https://github.com/jennybc/gapminder/issues>

Depends R (>= 3.1.0)

Imports tibble

Suggests dplyr, ggplot2, knitr, rmarkdown, testthat (>= 3.0.0)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Config/testthat/edition 3

VignetteBuilder knitr

Repository <https://jennybc.r-universe.dev>

RemoteUrl <https://github.com/jennybc/gapminder>

RemoteRef HEAD

RemoteSha b89587237a16a32c922340b4f43aef142873f237

Contents

country_codes	2
country_colors	3
gapminder	4
gapminder_unfiltered	6
Index	7

country_codes	<i>Country codes</i>
---------------	----------------------

Description

Country codes

Usage

country_codes

Format

Data frame of Gapminder country names and ISO 3166-1 country codes:

country Country name.

iso_alpha The 3-letter **ISO 3166-1 alpha-3** code.

iso_num The 3-digit **ISO 3166-1 numeric-3** code.

Also includes the countries covered by the supplemental data frame [gapminder_unfiltered](#).

Examples

```
if (require("dplyr")) {
  gapminder %>%
    filter(year == 2007, country %in% c("Kenya", "Peru", "Syria")) %>%
    select(country, continent) %>%
    left_join(country_codes)
}
```

country_colors	<i>Gapminder color schemes.</i>
----------------	---------------------------------

Description

Color schemes for the countries and continents in the Gapminder data.

Usage

```
country_colors
```

Format

Named character vectors giving country and continent colors:

country_colors colors for the 142 countries

continent_colors colors for the 5 continents

See Also

[gapminder](#) for a description of the dataset

Examples

```
# ggplot2 examples are below these base graphics examples!

# using country_colors with base graphics

# for convenience, integrate the country colors into the data.frame
gap_with_colors <-
  data.frame(gapminder,
             cc = I(country_colors[match(
               gapminder$country,
               names(country_colors)
             )]))

# bubble plot, focus just on Africa and Europe in 2007
keepers <- with(
  gap_with_colors,
  continent %in% c("Africa", "Europe") & year == 2007
)
plot(lifeExp ~ gdpPercap, gap_with_colors,
     subset = keepers, log = "x", pch = 21,
     cex = sqrt(gap_with_colors$pop[keepers] / pi) / 1500,
     bg = gap_with_colors$cc[keepers]
)

if (require(ggplot2)) {
```

```

# with ggplot2, just provide country_colors to scale_color_manual():
# ... + scale_color_manual(values = country_colors) + ...

# simple line plot for 5 countries
h_countries <- c("Egypt", "Haiti", "Romania", "Thailand", "Venezuela")
h_dat <- droplevels(subset(gapminder, country %in% h_countries))
h_dat$country <- with(h_dat, reorder(country, lifeExp, max))
ggplot(h_dat, aes(x = year, y = lifeExp)) +
  geom_line(aes(color = country)) +
  scale_colour_manual(values = country_colors) +
  guides(color = guide_legend(reverse = TRUE))

# spaghetti plot for lots of countries
ggplot(
  subset(gapminder, continent != "Oceania"),
  aes(x = year, y = lifeExp, group = country, color = country)
) +
  geom_line(lwd = 1, show_guide = FALSE) +
  facet_wrap(~continent) +
  scale_color_manual(values = country_colors) +
  theme_bw() +
  theme(strip.text = element_text(size = rel(1.1)))

# bubble plot for lots of countries
gap_bit <- subset(gapminder, year == 2007 & continent != "Oceania")
gap_bit <- gap_bit[with(gap_bit, order(continent, -1 * pop)), ]
ggplot(gap_bit, aes(x = gdpPercap, y = lifeExp, size = pop)) +
  scale_x_log10(limits = c(150, 115000)) +
  ylim(c(16, 96)) +
  geom_point(pch = 21, color = "grey20", show_guide = FALSE) +
  scale_size_area(max_size = 40) +
  facet_wrap(~continent) +
  coord_fixed(ratio = 1 / 43) +
  aes(fill = country) +
  scale_fill_manual(values = country_colors) +
  theme_bw() +
  theme(strip.text = element_text(size = rel(1.1)))
}

```

gapminder

Gapminder data

Description

Excerpt of the Gapminder data on life expectancy, GDP per capita, and population by country.

Usage

gapminder

Format

The main data frame `gapminder` has 1704 rows and 6 variables:

country factor with 142 levels
continent factor with 5 levels
year ranges from 1952 to 2007 in increments of 5 years
lifeExp life expectancy at birth, in years
pop population
gdpPercap GDP per capita (US\$, inflation-adjusted)

The supplemental data frame `gapminder_unfiltered` was not filtered on year or for complete data and has 3313 rows.

Source

<https://www.gapminder.org/data/>

See Also

[country_colors](#) for a nice color scheme for the countries

Examples

```
str(gapminder)
head(gapminder)
summary(gapminder)
table(gapminder$continent)
aggregate(lifeExp ~ continent, gapminder, median)
plot(lifeExp ~ year, gapminder, subset = country == "Cambodia", type = "b")
plot(lifeExp ~ gdpPercap, gapminder, subset = year == 2007, log = "x")

if (require("dplyr")) {
  gapminder %>%
    filter(year == 2007) %>%
    group_by(continent) %>%
    summarise(lifeExp = median(lifeExp))

  # how many unique countries does the data contain, by continent?
  gapminder %>%
    group_by(continent) %>%
    summarize(n_obs = n(), n_countries = n_distinct(country))

  # by continent, which country experienced the sharpest 5-year drop in
  # life expectancy and what was the drop?
  gapminder %>%
    group_by(continent, country) %>%
    select(country, year, continent, lifeExp) %>%
    mutate(le_delta = lifeExp - lag(lifeExp)) %>%
    summarize(worst_le_delta = min(le_delta, na.rm = TRUE)) %>%
    filter(min_rank(worst_le_delta) < 2) %>%
```

```
    arrange(worst_le_delta)  
  }
```

gapminder_unfiltered *Gapminder data, unfiltered.*

Description

The supplemental data frame `gapminder_unfiltered` was not filtered on year or for complete data and has 3313 rows. Everything else is as documented in [gapminder](#).

Usage

```
gapminder_unfiltered
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 3313 rows and 6 columns.

Index

* datasets

country_codes, 2

country_colors, 3

gapminder, 4

gapminder_unfiltered, 6

continent_colors (country_colors), 3

country_codes, 2

country_colors, 3, 5

gapminder, 3, 4, 6

gapminder_unfiltered, 2, 5, 6