Assignment 7

Libraries

```
library(tidyverse)
```

```
## -- Attaching packages ------------------ tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## Warning: package 'readr' was built under R version 3.6.3

## Warning: package 'forcats' was built under R version 3.6.3

## -- Conflicts --------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(XLConnect)
```

```
## Warning: package 'XLConnect' was built under R version 3.6.3

## Loading required package: XLConnectJars

## XLConnect 0.2-15 by Mirai Solutions GmbH [aut],
##   Martin Studer [cre],
##   The Apache Software Foundation [ctb, cph] (Apache POI),
##   Graph Builder [ctb, cph] (Curvesapi Java library)

## http://www.mirai-solutions.com
## https://github.com/miraisolutions/xlconnect
```

```
library(openxlsx)
```

```
##
## Attaching package: 'openxlsx'

## The following objects are masked from 'package:XLConnect':
##
##     getTables, loadWorkbook, mergeCells, saveWorkbook
```

```
library(countrycode)
```

```
## Warning: package 'countrycode' was built under R version 3.6.3
```

```
library(purrr)
library(dplyr)
library(ggplot2)
library(gganimate)
```

```
## Warning: package 'gganimate' was built under R version 3.6.3
```

```
theme_set(theme_bw())
library(modelr)
library(broom)
```

```
## Warning: package 'broom' was built under R version 3.6.3
```

```
##
## Attaching package: 'broom'
```

```
## The following object is masked from 'package:modelr':
##
##     bootstrap
```

Part 1: Creating our own gapminder animation

All the important URLS

```
if(!file.exists("./data")) {dir.create("./data")}


fileUrls <- c(
  "https://docs.google.com/spreadsheet/pub?key=0AkBd6lyS3EmpdHo5S0J6ekhVOF9QaVhod05QSGV4T3c&output=xlsx
  "https://docs.google.com/spreadsheet/pub?key=phAwcNAVuyj2tPLxKvvnNPA&output=xlsx",
  "https://docs.google.com/spreadsheet/pub?key=tSUr_yZVbM6a3AGJEq_Z2Pw&output=xlsx",
  "https://docs.google.com/spreadsheet/pub?key=0ArfEDsV3bBwCdHBzUVVSMDlTX1ZCUnNJQ3ZFdkFXVFE&output=xlsx
  "https://docs.google.com/spreadsheet/pub?key=phAwcNAVuyj0XOoBL_n5tAQ&output=xlsx")

var_names <- c("GDP","life_expectancy", "alt_GDP", "blood press", "population")
```

get_clean function

```
round_df <- function(x, digits) {
    # round all numeric variables
    # x: data frame
    # digits: number of digits to round
    numeric_columns <- sapply(x, mode) == 'numeric'
    x[numeric_columns] <-  round(x[numeric_columns], digits)
    x
```

```
}

get_clean <- function(url_in, var_name){
  tmp = tempfile(fileext = ".xlsx")
  download.file(url_in, destfile = tmp, mode="wb")
  data<-readWorkbook(tmp)

  data1<-data %>%
    rename(country=colnames(data[1])) %>%
    pivot_longer(-country,names_to = "year",values_to = var_name) %>%
    na.omit()

  data2<-round_df(data1,0)
  data2$year<-as.numeric(data2$year)
  return(data2)
}

setwd("data/")
out1 <- get_clean(fileUrls[1],var_names[1])
head(out1)
```

```
## # A tibble: 6 x 3
##   country  year   GDP
##   <chr>   <dbl> <dbl>
## 1 Albania  1980  1061
## 2 Albania  1981  1100
## 3 Albania  1982  1111
## 4 Albania  1983  1101
## 5 Albania  1984  1065
## 6 Albania  1985  1060
```

```
all_data<-map2(fileUrls,var_names,get_clean)
head(all_data)
```

```
## [[1]]
## # A tibble: 7,988 x 3
##    country  year   GDP
##    <chr>   <dbl> <dbl>
##  1 Albania  1980  1061
##  2 Albania  1981  1100
##  3 Albania  1982  1111
##  4 Albania  1983  1101
##  5 Albania  1984  1065
##  6 Albania  1985  1060
##  7 Albania  1986  1092
##  8 Albania  1987  1054
##  9 Albania  1988  1014
## 10 Albania  1989  1092
## # ... with 7,978 more rows
##
## [[2]]
## # A tibble: 43,857 x 3
##    country      year life_expectancy
```

```
##    <chr>       <dbl>           <dbl>
##  1 Afghanistan  1800              28
##  2 Afghanistan  1801              28
##  3 Afghanistan  1802              28
##  4 Afghanistan  1803              28
##  5 Afghanistan  1804              28
##  6 Afghanistan  1805              28
##  7 Afghanistan  1806              28
##  8 Afghanistan  1807              28
##  9 Afghanistan  1808              28
## 10 Afghanistan  1809              28
## # ... with 43,847 more rows
##
## [[3]]
## # A tibble: 7,334 x 3
##    country      year alt_GDP
##    <chr>       <dbl>   <dbl>
##  1 Afghanistan  1970    1731
##  2 Afghanistan  1971    1748
##  3 Afghanistan  1972    2120
##  4 Afghanistan  1973    2119
##  5 Afghanistan  1974    2148
##  6 Afghanistan  1975    2263
##  7 Afghanistan  1976    2270
##  8 Afghanistan  1977    2121
##  9 Afghanistan  1978    2205
## 10 Afghanistan  1979    2121
## # ... with 7,324 more rows
##
## [[4]]
## # A tibble: 5,771 x 3
##    country      year `blood press`
##    <chr>       <dbl>         <dbl>
##  1 Afghanistan  1980           122
##  2 Afghanistan  1981           122
##  3 Afghanistan  1982           122
##  4 Afghanistan  1983           123
##  5 Afghanistan  1984           123
##  6 Afghanistan  1985           123
##  7 Afghanistan  1986           123
##  8 Afghanistan  1987           123
##  9 Afghanistan  1988           124
## 10 Afghanistan  1989           124
## # ... with 5,761 more rows
##
## [[5]]
## # A tibble: 20,176 x 3
##    country      year population
##    <chr>       <dbl>      <dbl>
##  1 Afghanistan  1800    3280000
##  2 Afghanistan  1810    3280000
##  3 Afghanistan  1820    3323519
##  4 Afghanistan  1830    3448982
##  5 Afghanistan  1840    3625022
```

```
##  6 Afghanistan  1850    3810047
##  7 Afghanistan  1860    3973968
##  8 Afghanistan  1870    4169690
##  9 Afghanistan  1880    4419695
## 10 Afghanistan  1890    4710171
## # ... with 20,166 more rows
```

2. Join the outputs into one tibble with a column for each variable (hint – perhaps use one of your purrr functions)

```
dat1<-all_data[[1]]
dat2<-all_data[[2]]
dat3<-all_data[[3]]
dat4<-all_data[[4]]
dat5<-all_data[[5]]

dat6<-full_join(dat2,dat1)
```

```
## Joining, by = c("country", "year")
```

```
dat7<-full_join(dat3,dat4)
```

```
## Joining, by = c("country", "year")
```

```
dat8<-full_join(dat6,dat7)
```

```
## Joining, by = c("country", "year")
```

```
join_data<-full_join(dat8,dat5)
```

```
## Joining, by = c("country", "year")
```

```
join_data
```

```
## # A tibble: 48,270 x 7
##    country      year life_expectancy   GDP alt_GDP `blood press` population
##    <chr>       <dbl>           <dbl> <dbl>   <dbl>         <dbl>      <dbl>
##  1 Afghanistan  1800              28    NA      NA            NA    3280000
##  2 Afghanistan  1801              28    NA      NA            NA         NA
##  3 Afghanistan  1802              28    NA      NA            NA         NA
##  4 Afghanistan  1803              28    NA      NA            NA         NA
##  5 Afghanistan  1804              28    NA      NA            NA         NA
##  6 Afghanistan  1805              28    NA      NA            NA         NA
##  7 Afghanistan  1806              28    NA      NA            NA         NA
##  8 Afghanistan  1807              28    NA      NA            NA         NA
##  9 Afghanistan  1808              28    NA      NA            NA         NA
## 10 Afghanistan  1809              28    NA      NA            NA         NA
## # ... with 48,260 more rows
```

```
new_gapminder<-join_data %>%
  mutate(continent=countrycode(sourcevar = country,
                               origin = "country.name",
                               destination = "continent"))
```

```
## Warning in countrycode(sourcevar = country, origin = "country.name", destination = "continent"): Some
```

```
## Warning in countrycode(sourcevar = country, origin = "country.name", destination = "continent"): Some
```

```
new_gapminder<-new_gapminder %>%
  mutate(continent=case_when(country=="South Yemen (former)"~"Asia",
                             country=="Akrotiri and Dhekelia"~"Europe",
                             country=="Central African Rep."~"Africa",
                             country=="Channel Islands"~"Europe",
                             country=="Cocos Island"~"Asia",
                             country=="Czechoslovakia"~"Europe",
                             country=="East Germany"~"Europe",
                             country=="Eritrea and Ethiopia"~"Africa",
                             country=="Kosovo"~"Europe",
                             country=="North Yemen (former)"~"Asia",
                             country=="North Yemen (former)"~"Americas",
                             country=="St. Martin"~"Asia",
                             country=="Yugoslavia"~"Europe",
                             country=="Serbia and Montenegro"~"Asia",
                             TRUE~continent))
```

```
new_gapminder %>%
  arrange(country, year)
```

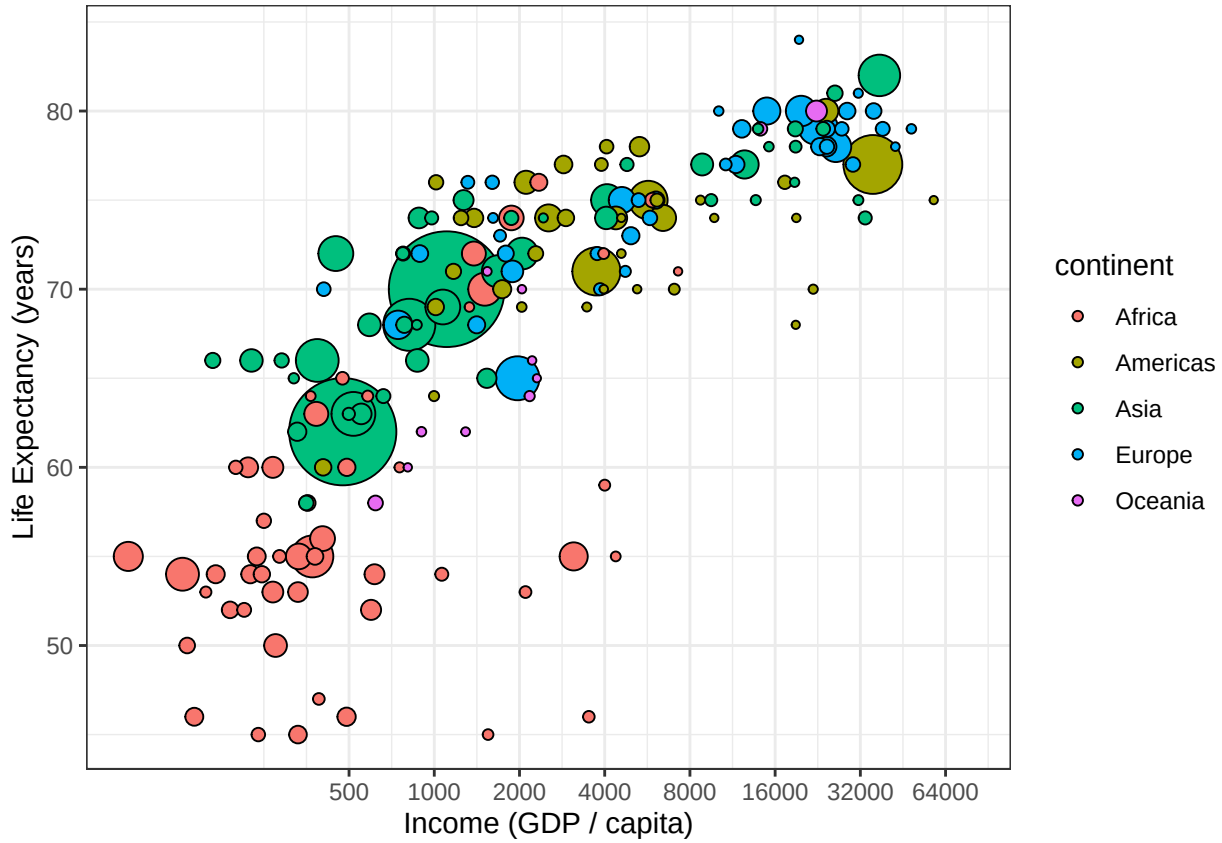
```
## # A tibble: 48,270 x 8
##    country  year life_expectancy   GDP alt_GDP `blood press` population
##    <chr>   <dbl>           <dbl> <dbl>   <dbl>         <dbl>      <dbl>
##  1 Afghan~  1800              28    NA      NA            NA    3280000
##  2 Afghan~  1801              28    NA      NA            NA         NA
##  3 Afghan~  1802              28    NA      NA            NA         NA
##  4 Afghan~  1803              28    NA      NA            NA         NA
##  5 Afghan~  1804              28    NA      NA            NA         NA
##  6 Afghan~  1805              28    NA      NA            NA         NA
##  7 Afghan~  1806              28    NA      NA            NA         NA
##  8 Afghan~  1807              28    NA      NA            NA         NA
##  9 Afghan~  1808              28    NA      NA            NA         NA
## 10 Afghan~  1809              28    NA      NA            NA         NA
## # ... with 48,260 more rows, and 1 more variable: continent <chr>
```

Just for my practice

```
p<-new_gapminder %>%
  filter(year=="2002") %>%
  arrange(desc(population)) %>%
  ggplot(aes(x=GDP,y=life_expectancy))+
```

```
  geom_point(aes(size=population,fill=continent),shape=21)+
  scale_x_log10(breaks = 2^(-1:7)*1000)+
  scale_size(range = c(1,20),guide=F)+
  labs(
    x="Income (GDP / capita)",
    y="Life Expectancy (years)"
  )+
  theme_bw()
p
```

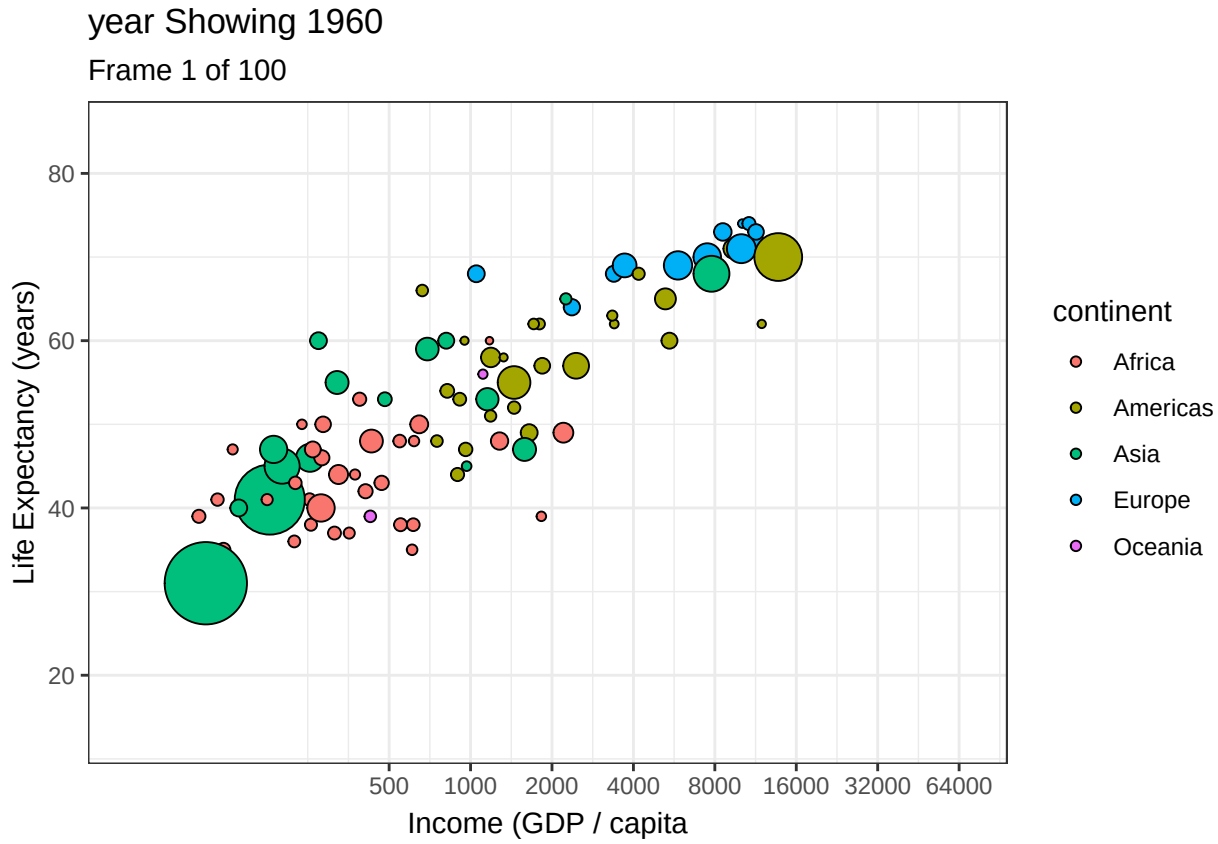
```
## Warning: Removed 89 rows containing missing values (geom_point).
```

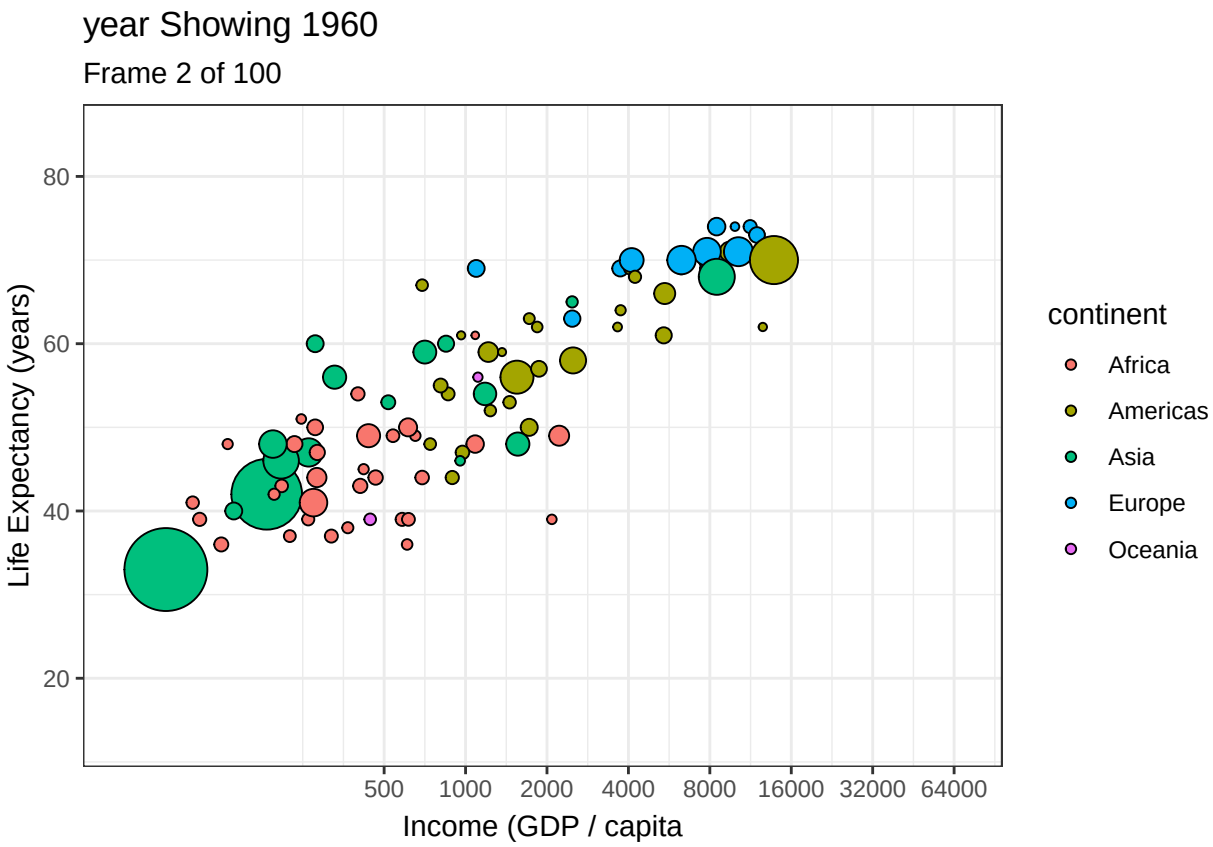


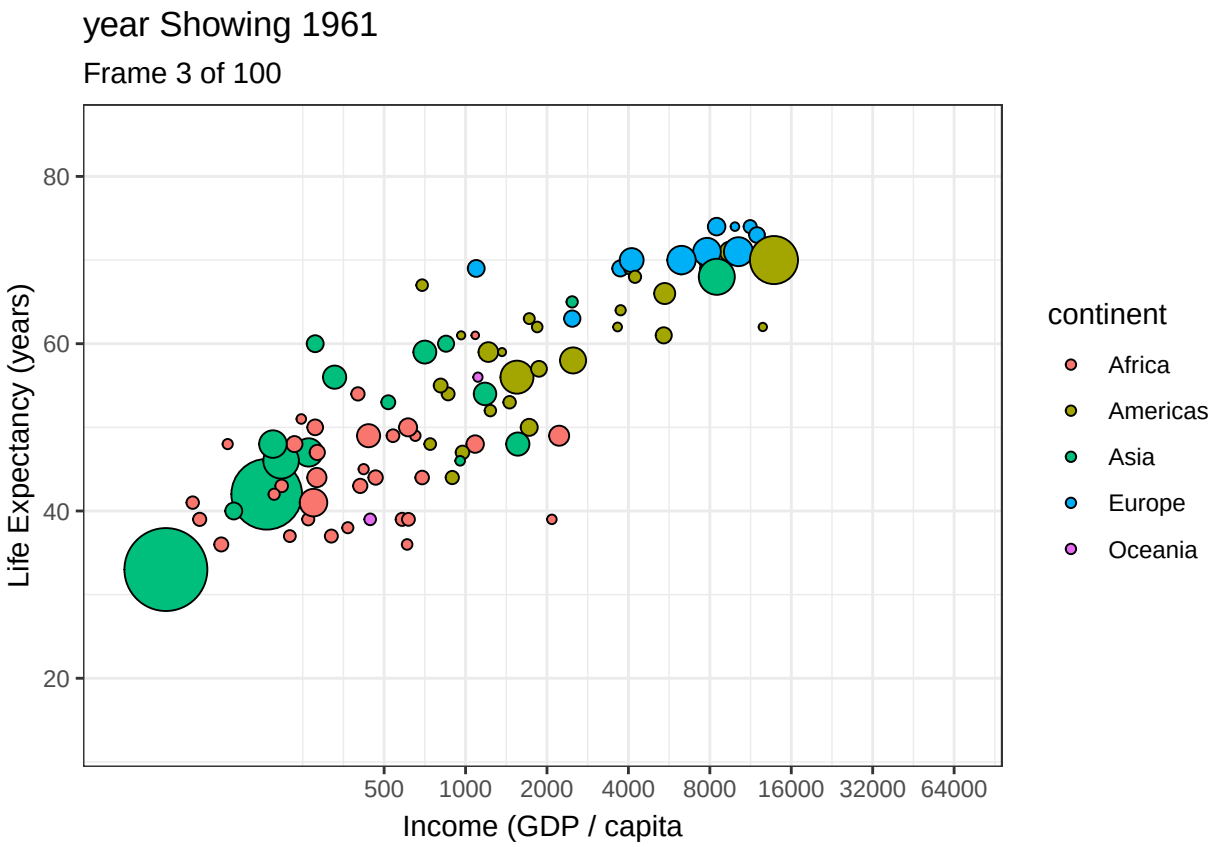
animation

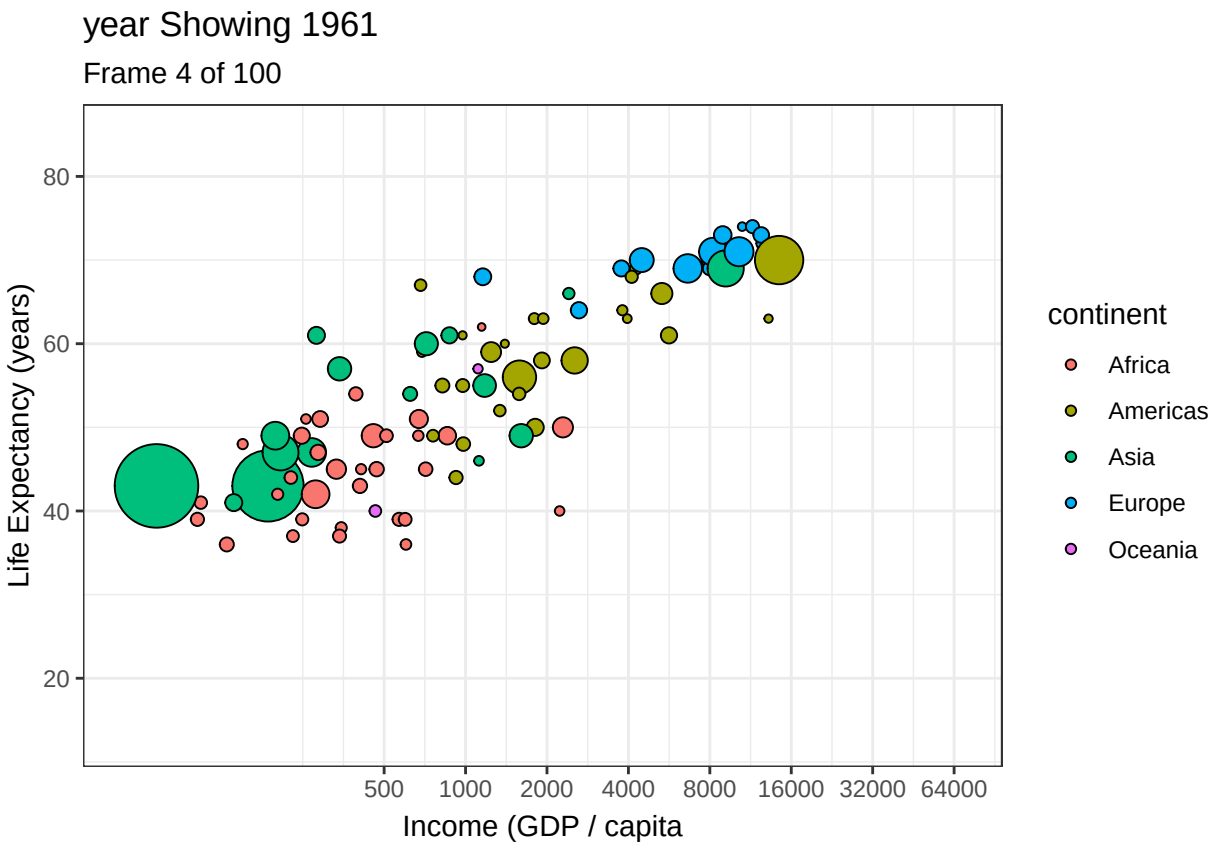
```
plot_data<-new_gapminder %>%
  select(country,continent,year,GDP,life_expectancy,population) %>%
  drop_na()


p <- ggplot(
  plot_data,
  aes(x = GDP, y=life_expectancy)) +
```

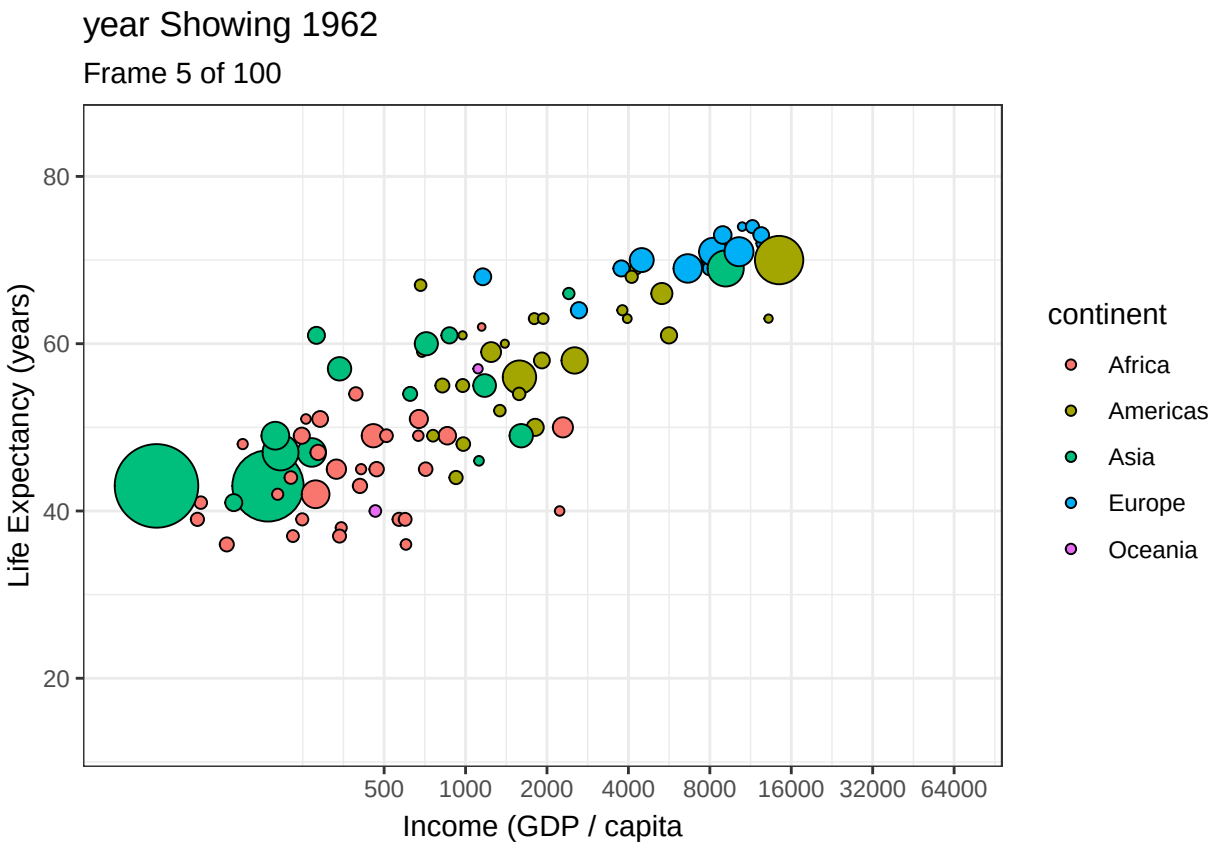
```
  geom_point(aes(size = population, fill = continent),shape=21)+
  labs(
    x="Income (GDP / capita",
    y="Life Expectancy (years)"
  )+
  theme_bw()+
  scale_x_log10(breaks = 2^(-1:7)*1000)+
  scale_size(range = c(1,20),guide=F)

#p + transition_time(year) +
#  labs(title = "Year: {frame_time}")

p<-p+transition_states(year,transition_length = 1,state_length = 1)+
  ggtitle("year Showing {closest_state}",subtitle = "Frame {frame} of {nframes}")

p
```

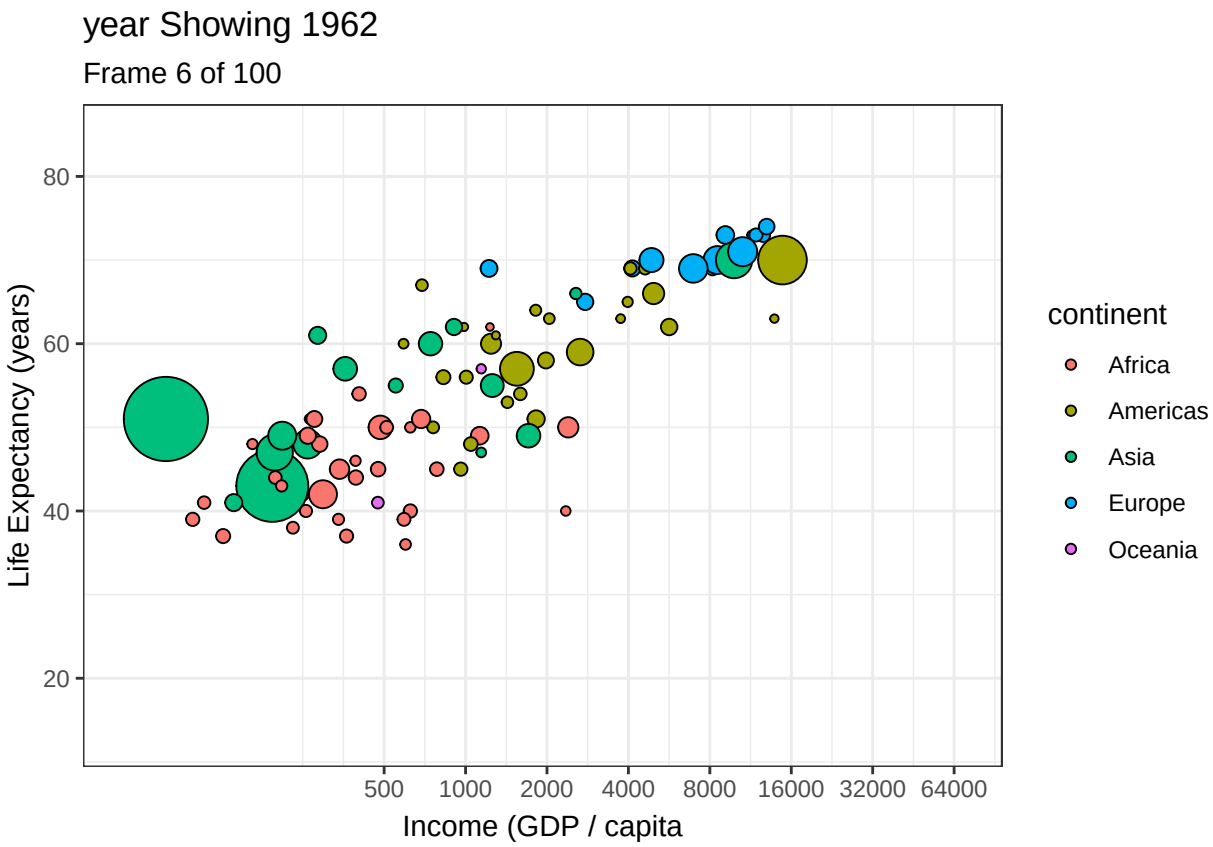


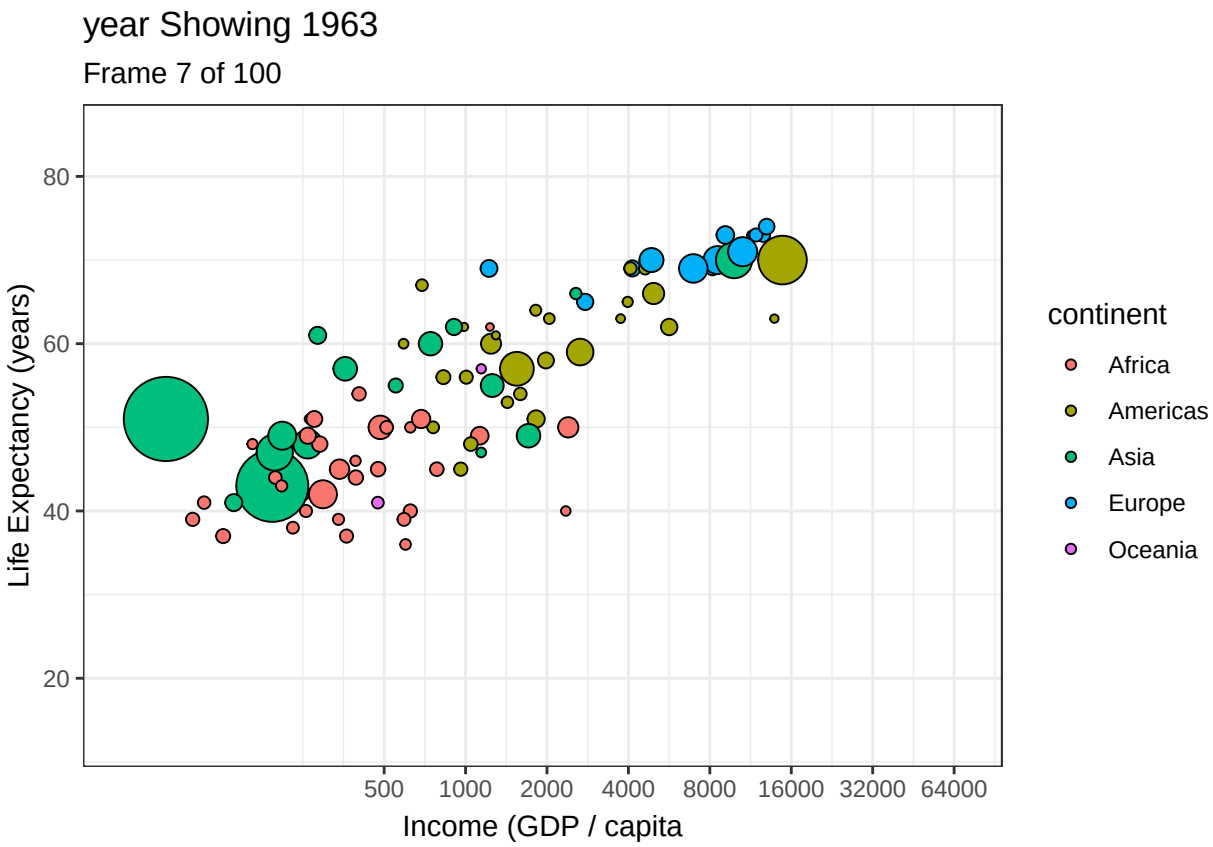

year Showing 1960
Frame 2 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

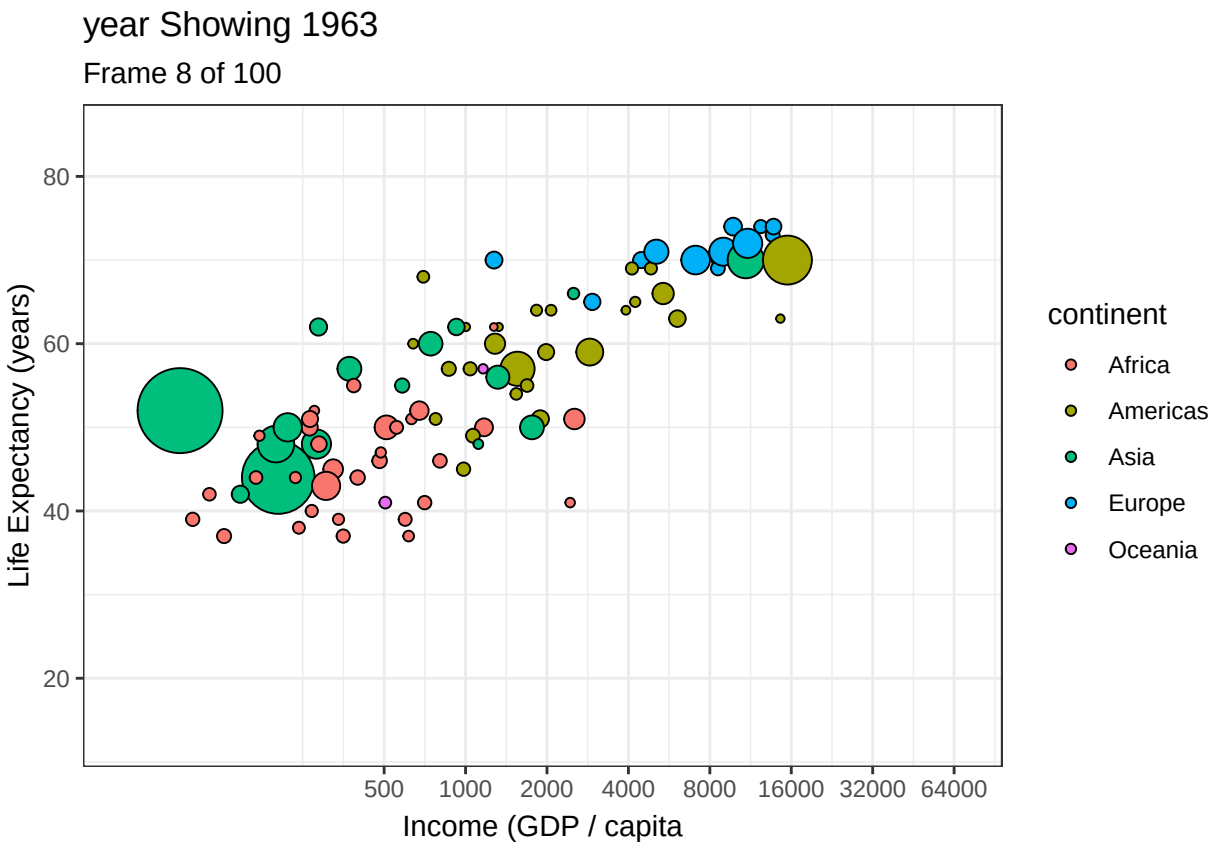

year Showing 1961
Frame 3 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

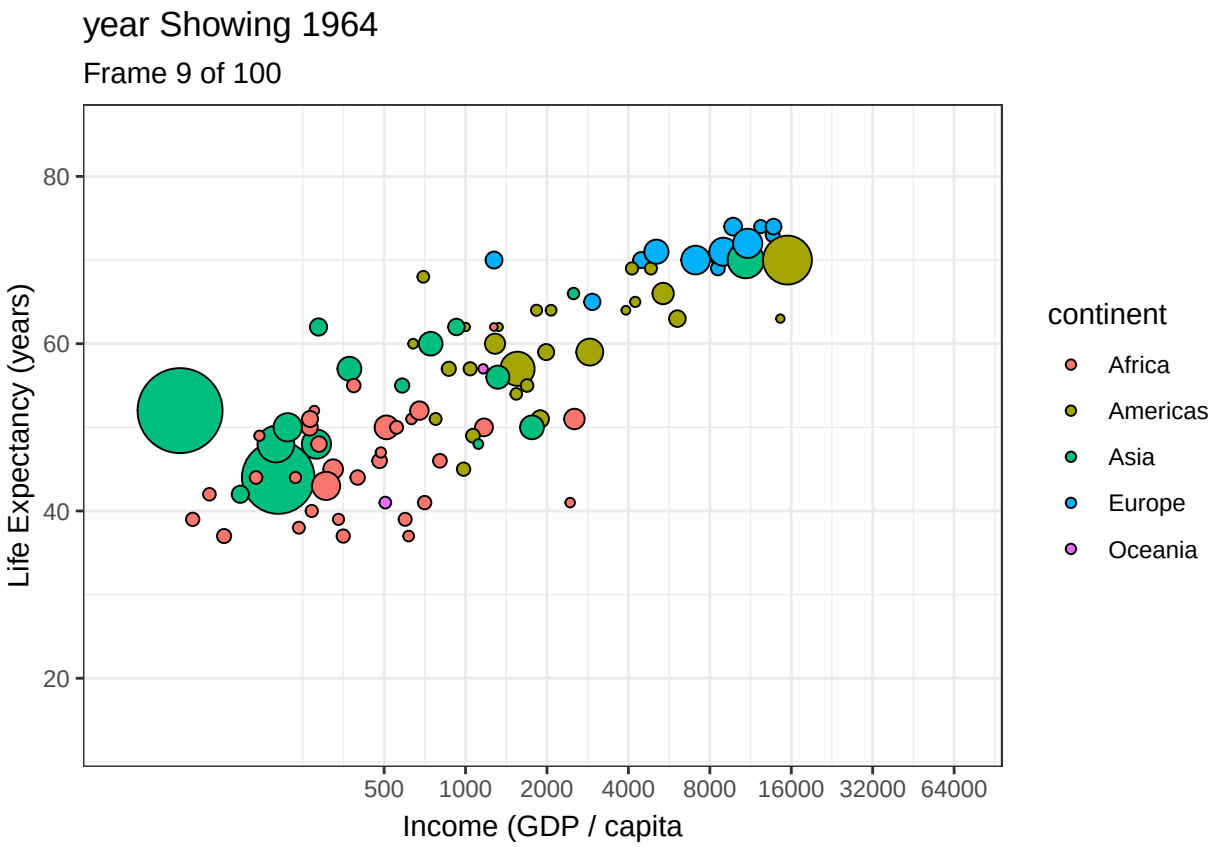

year Showing 1961
Frame 4 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

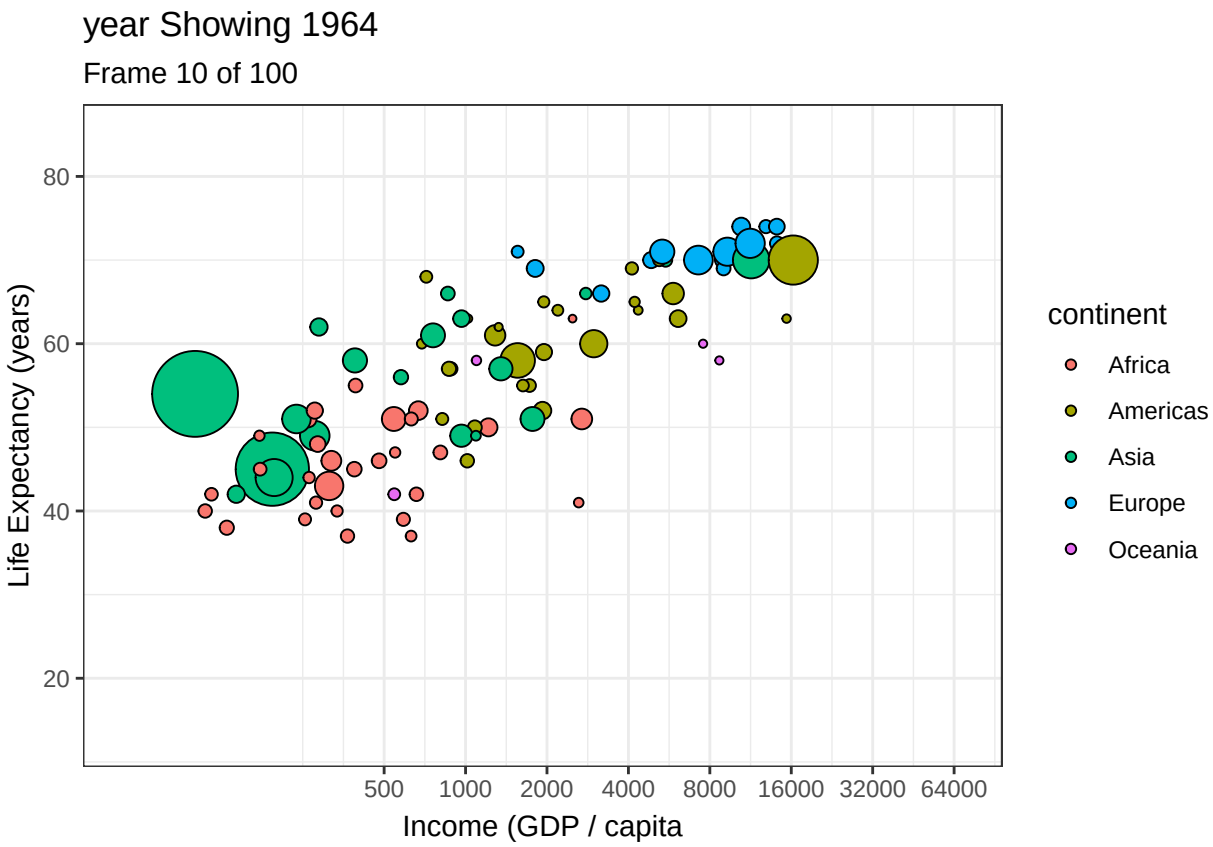

year Showing 1962
Frame 5 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

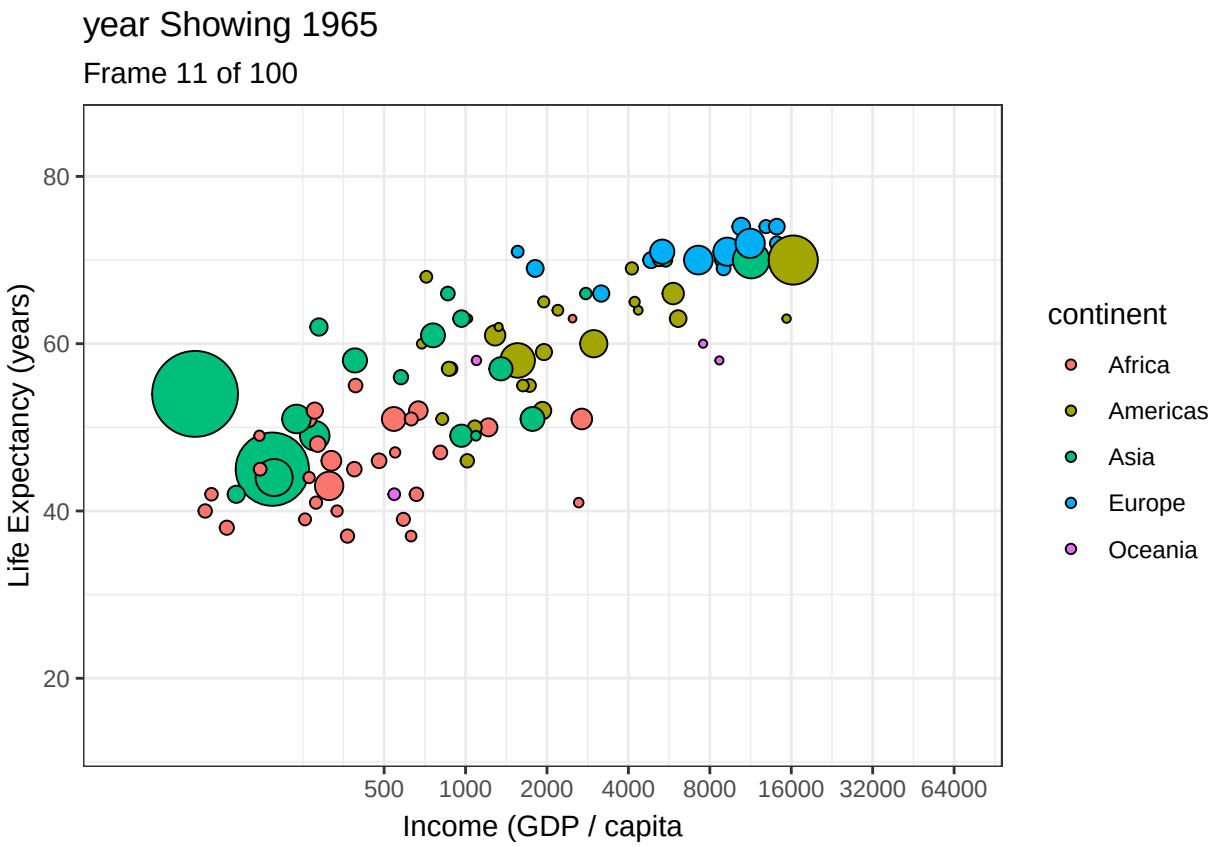

year Showing 1962
Frame 6 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

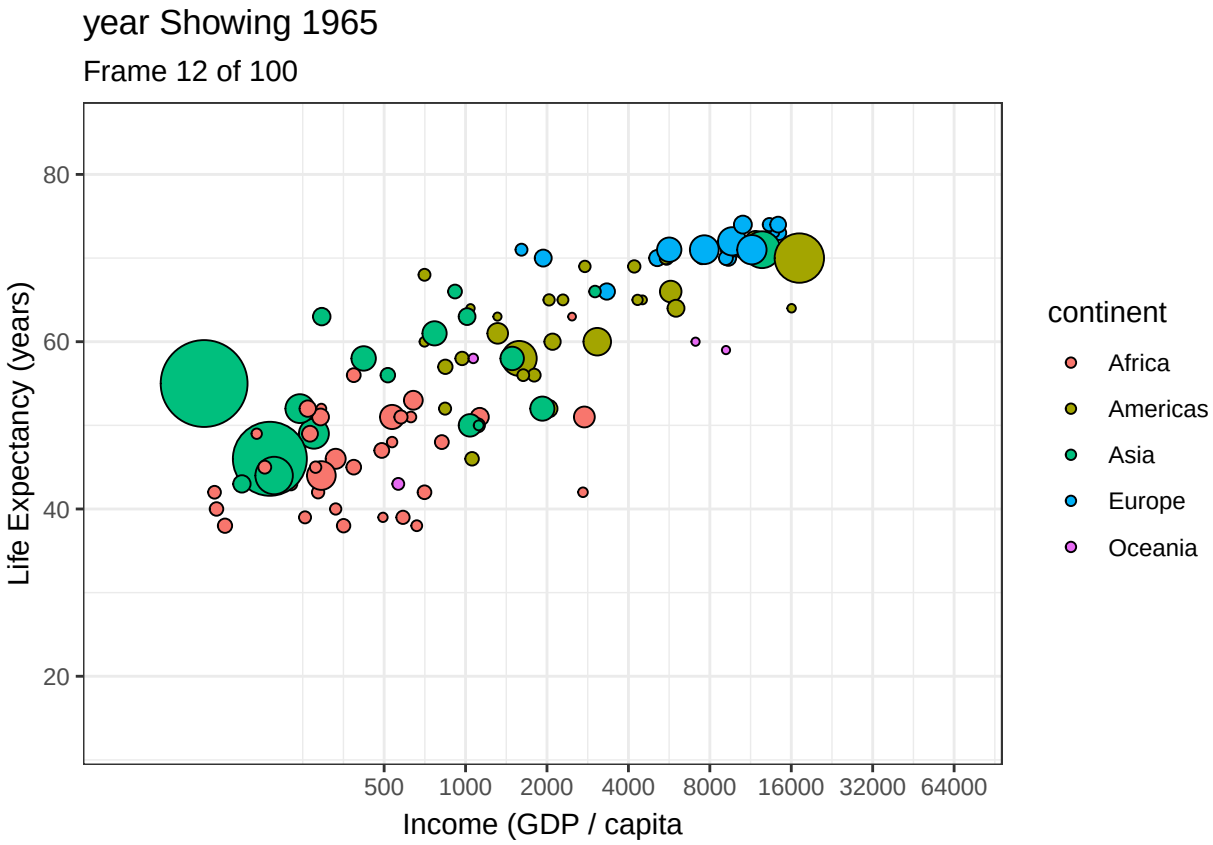

year Showing 1963
Frame 7 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

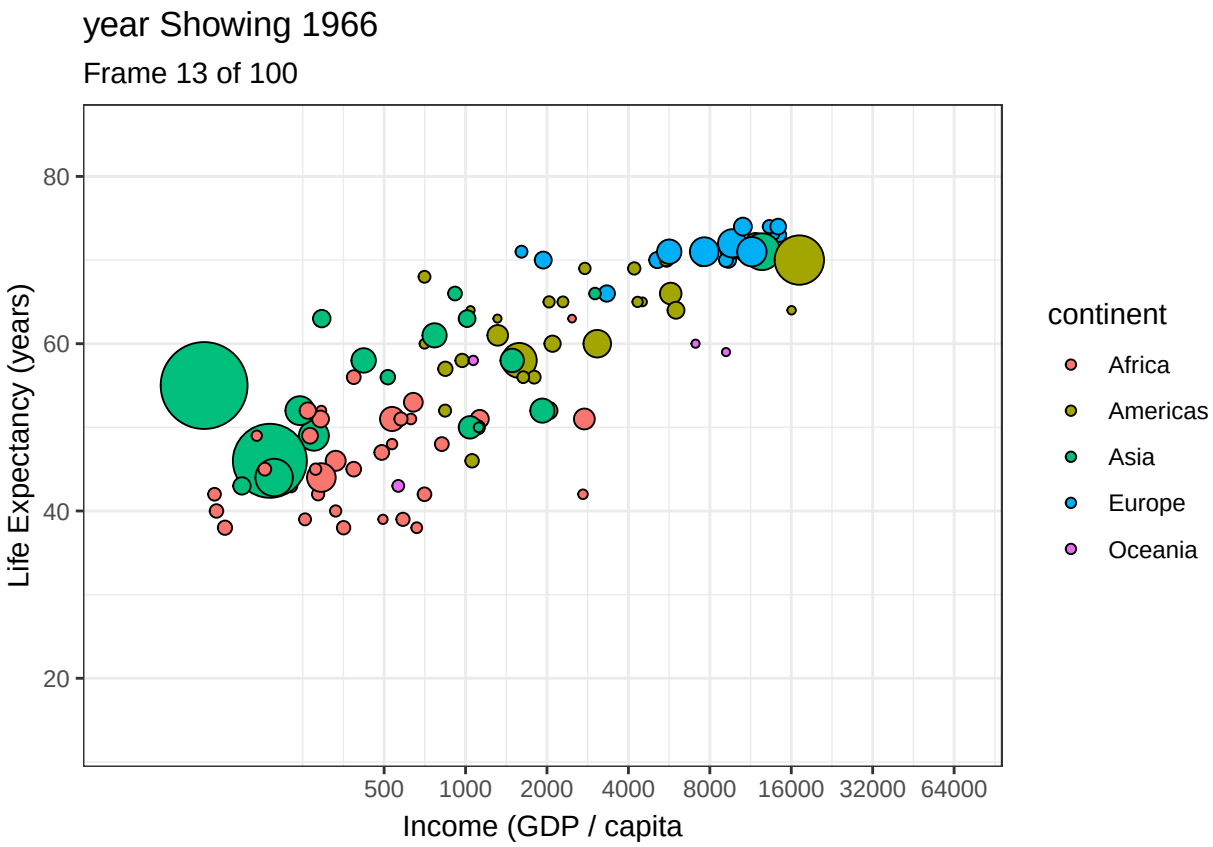

year Showing 1963
Frame 8 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

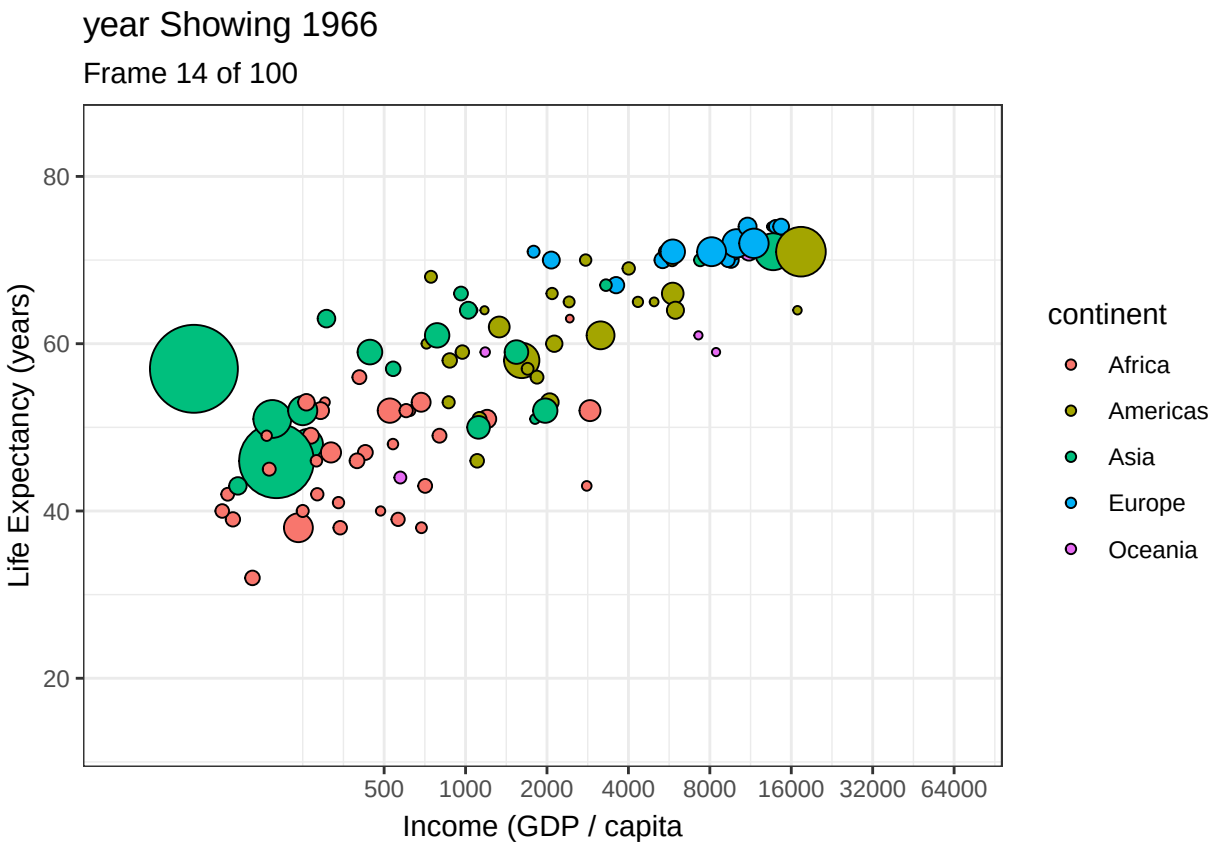

year Showing 1964
Frame 9 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

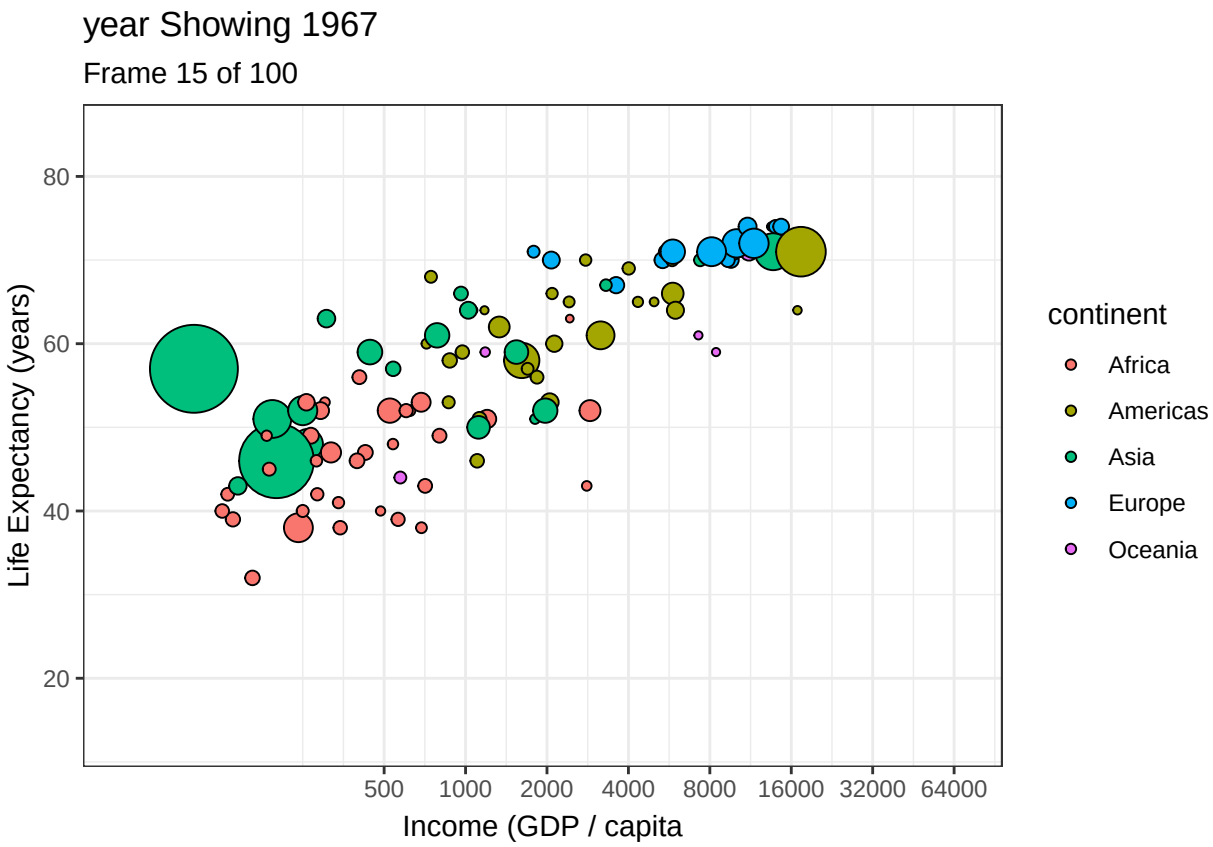

year Showing 1964
Frame 10 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

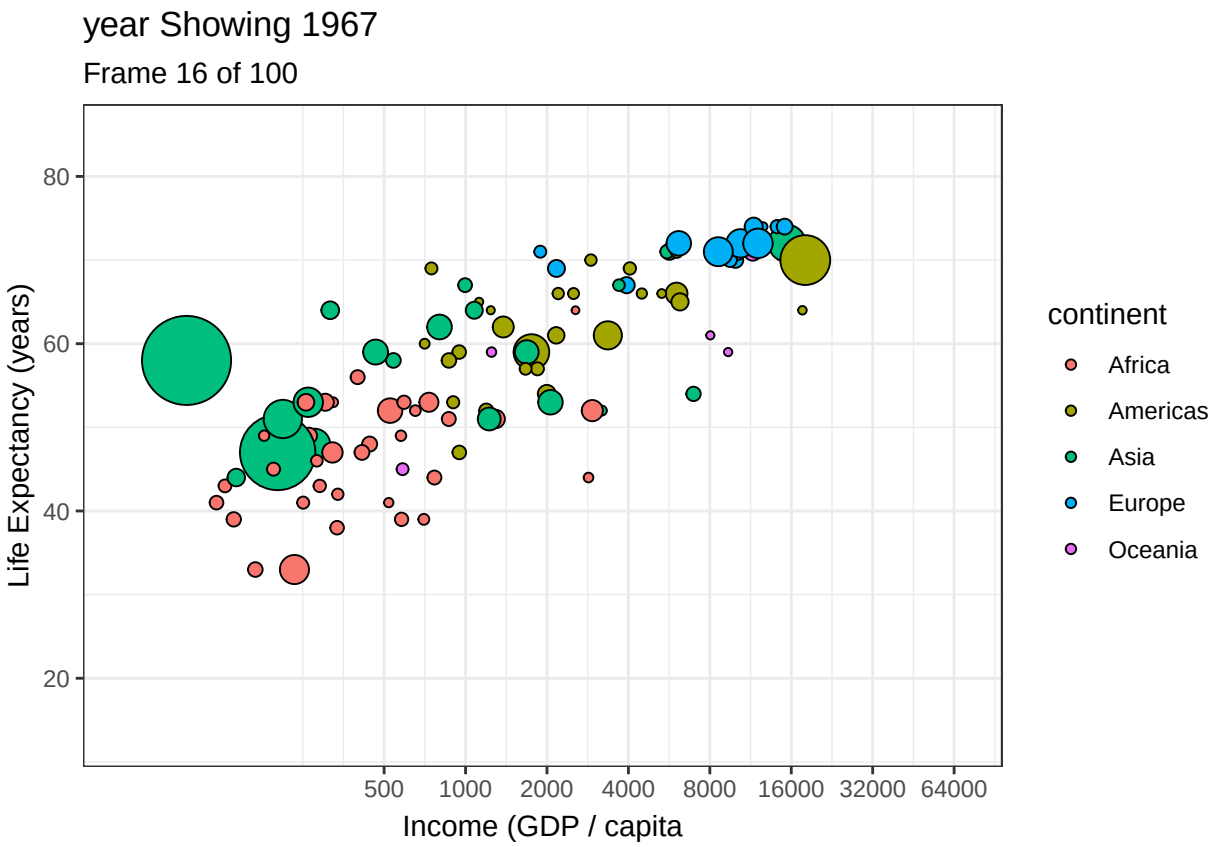

year Showing 1965
Frame 11 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

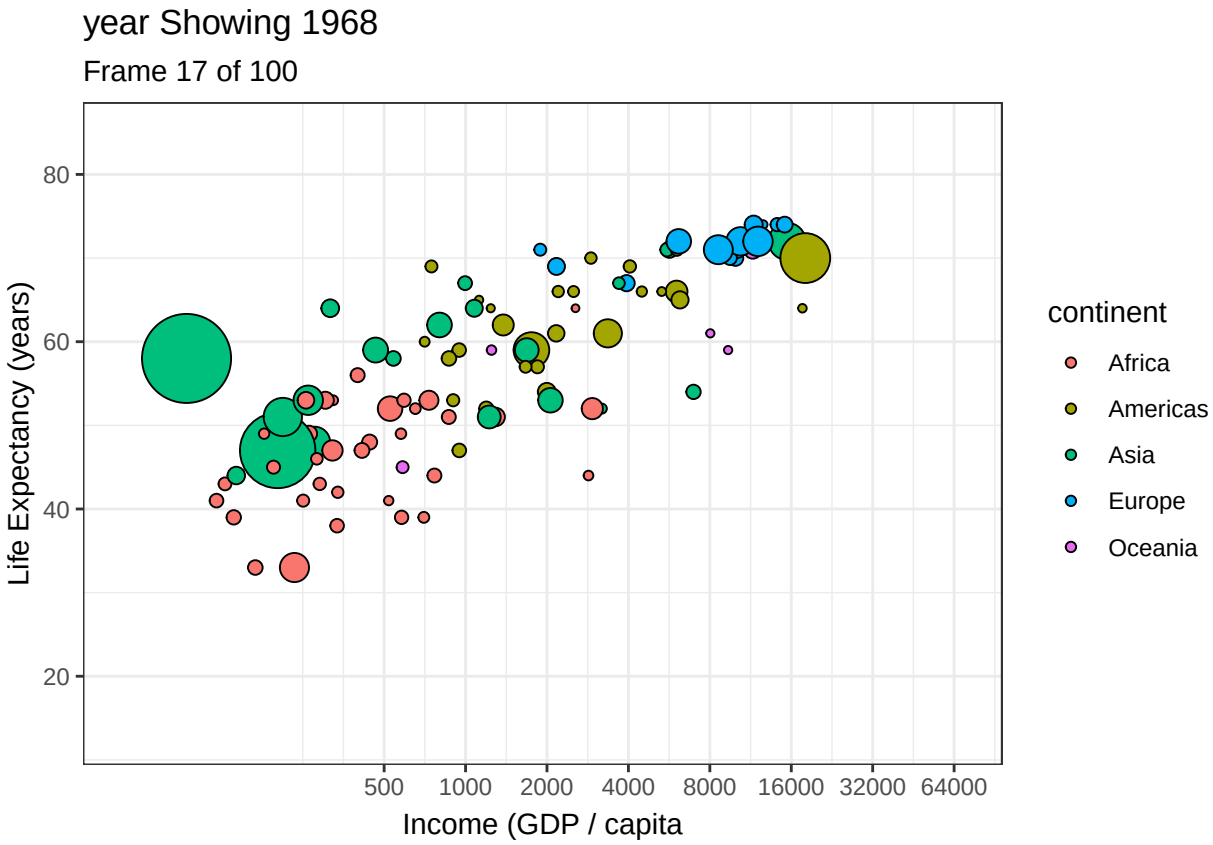

year Showing 1965
Frame 12 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

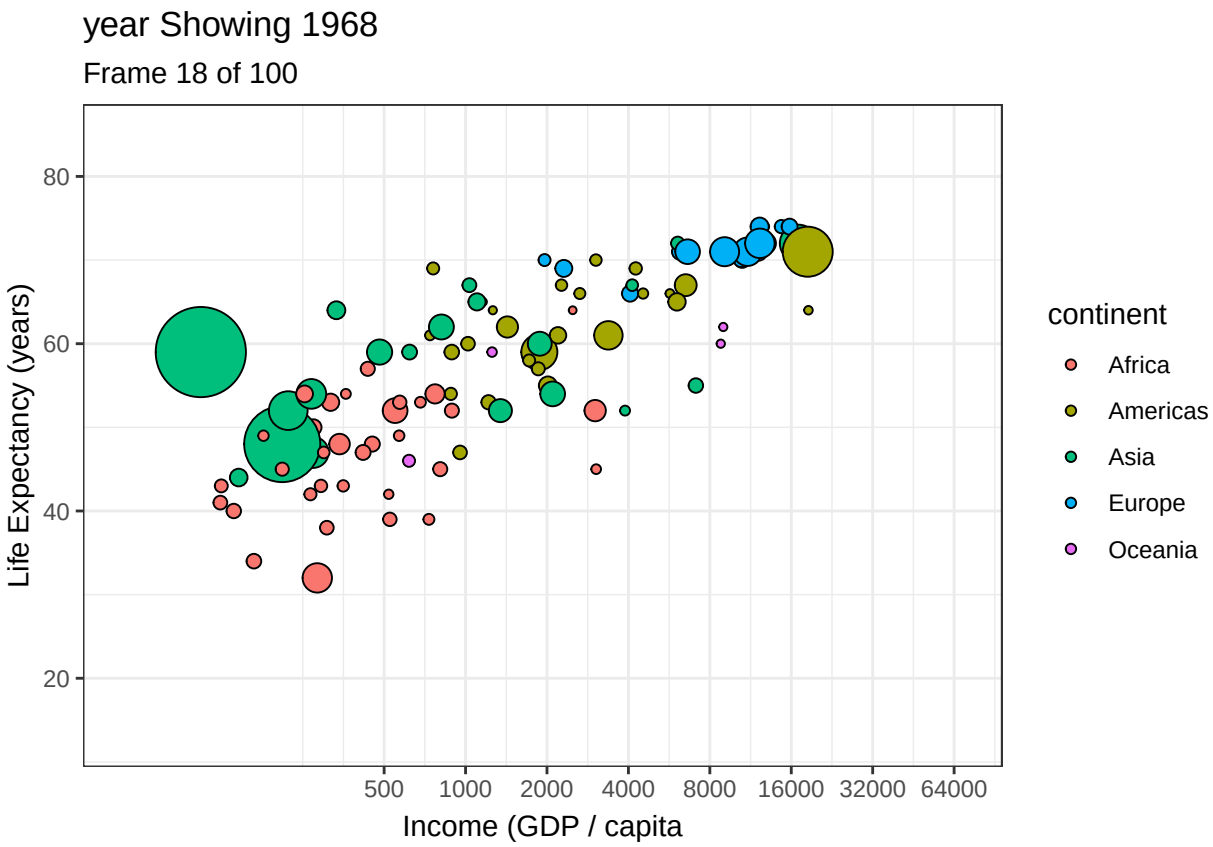

year Showing 1966
Frame 13 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

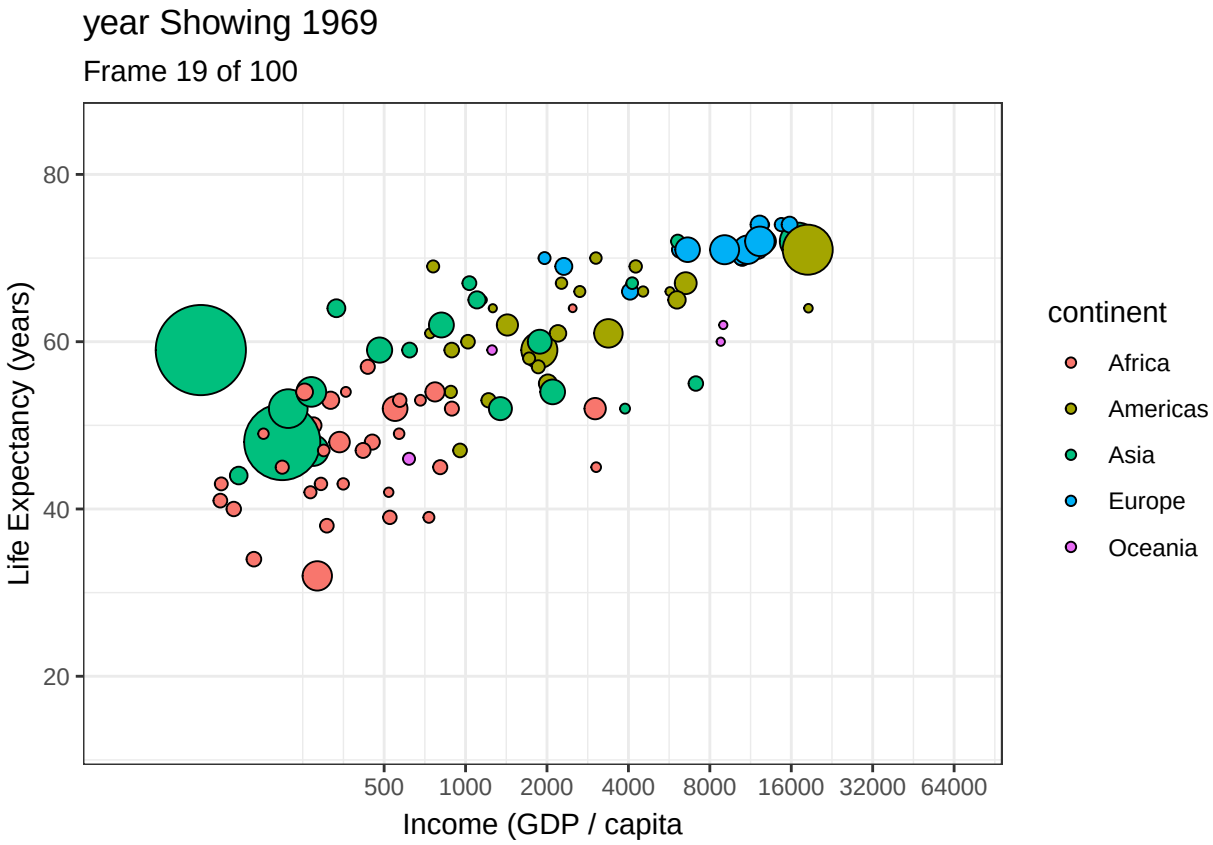

year Showing 1966
Frame 14 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

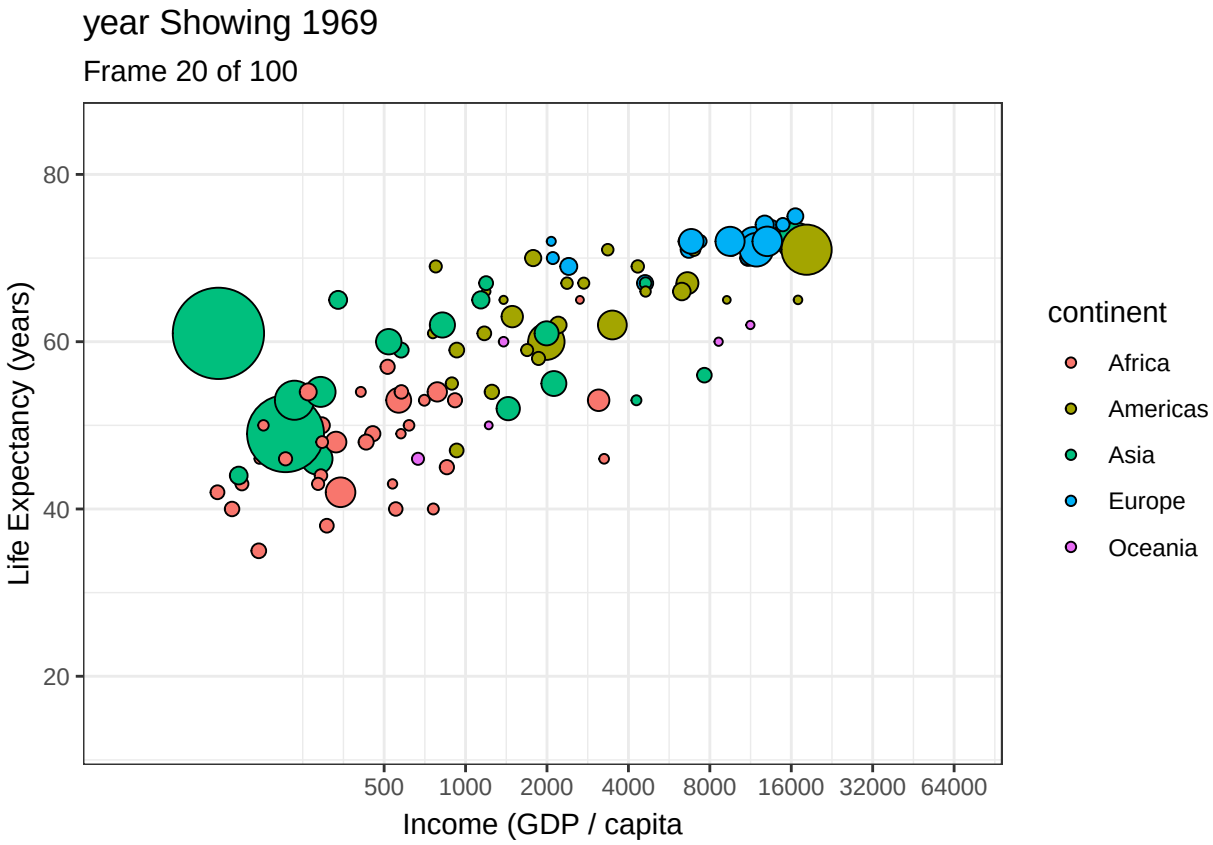

year Showing 1967
Frame 15 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

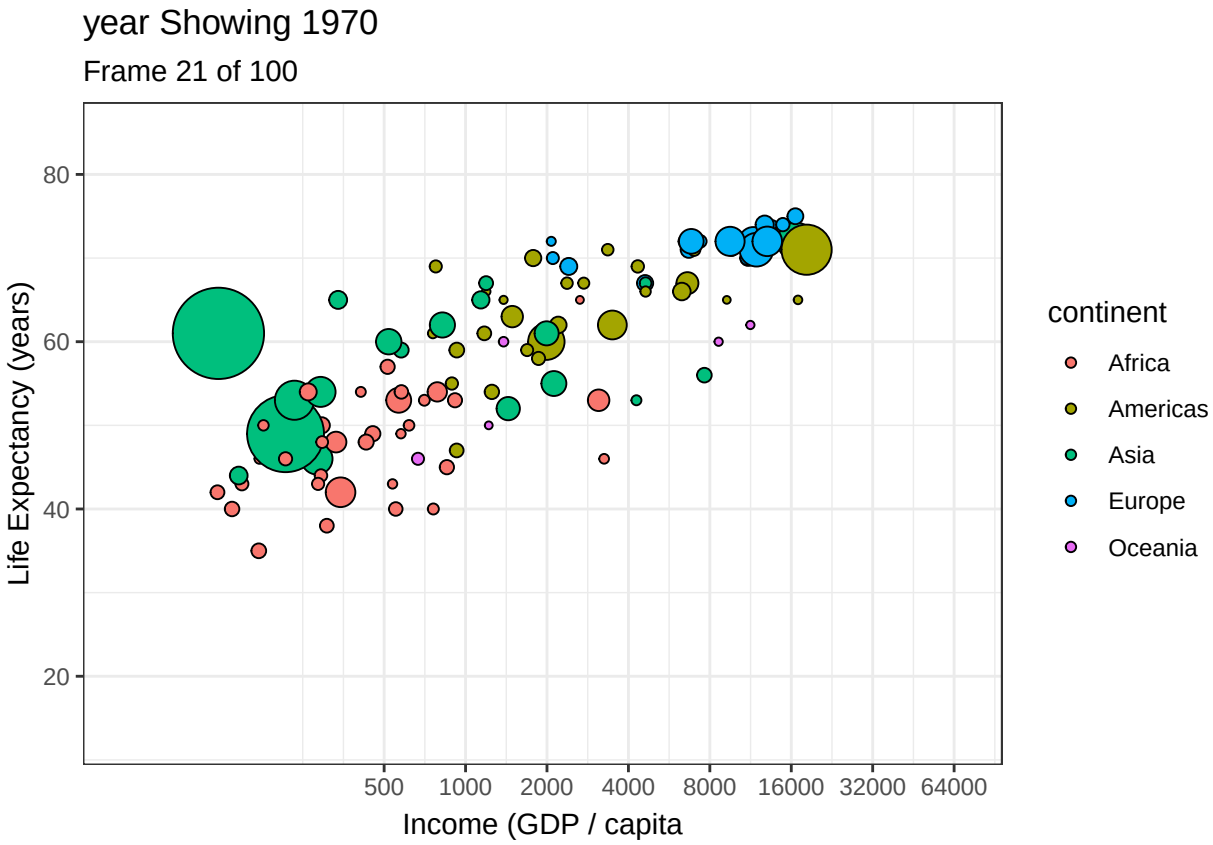

year Showing 1967
Frame 16 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

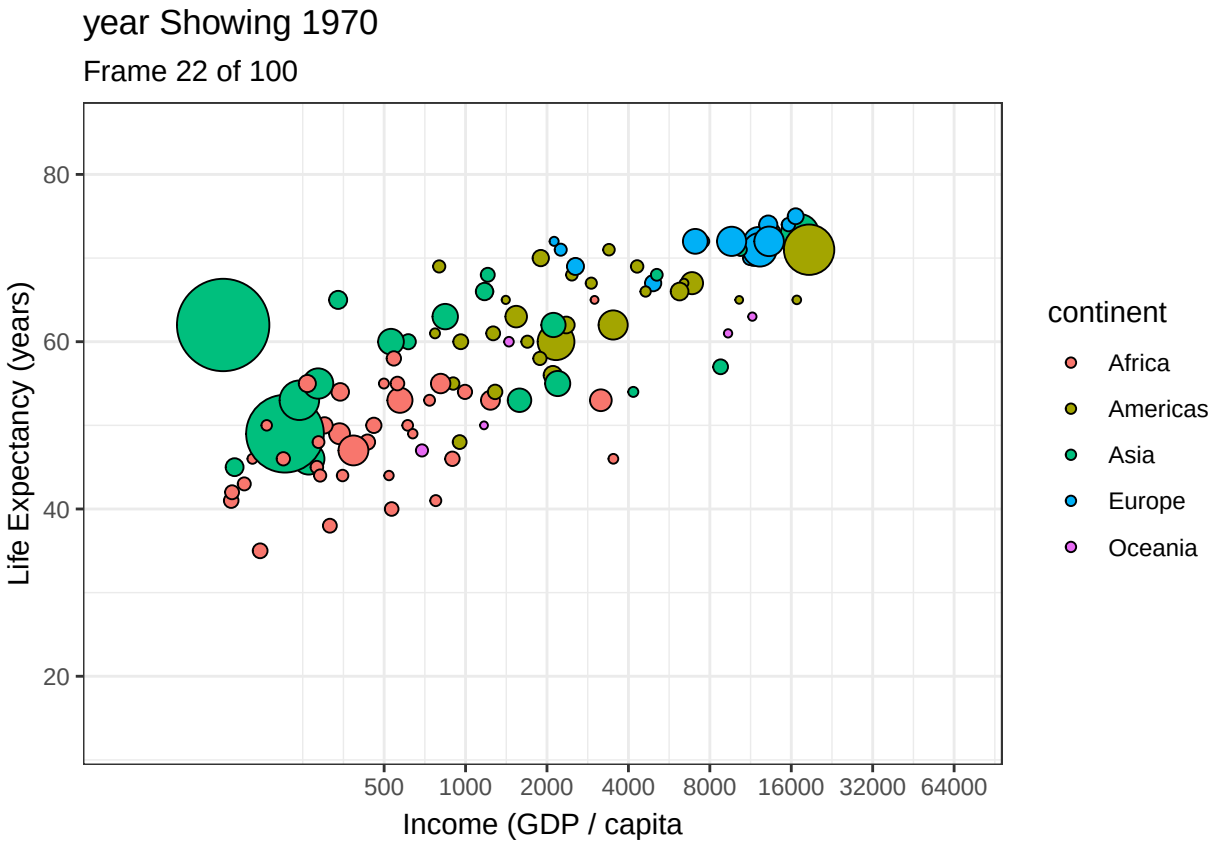

year Showing 1968
Frame 17 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

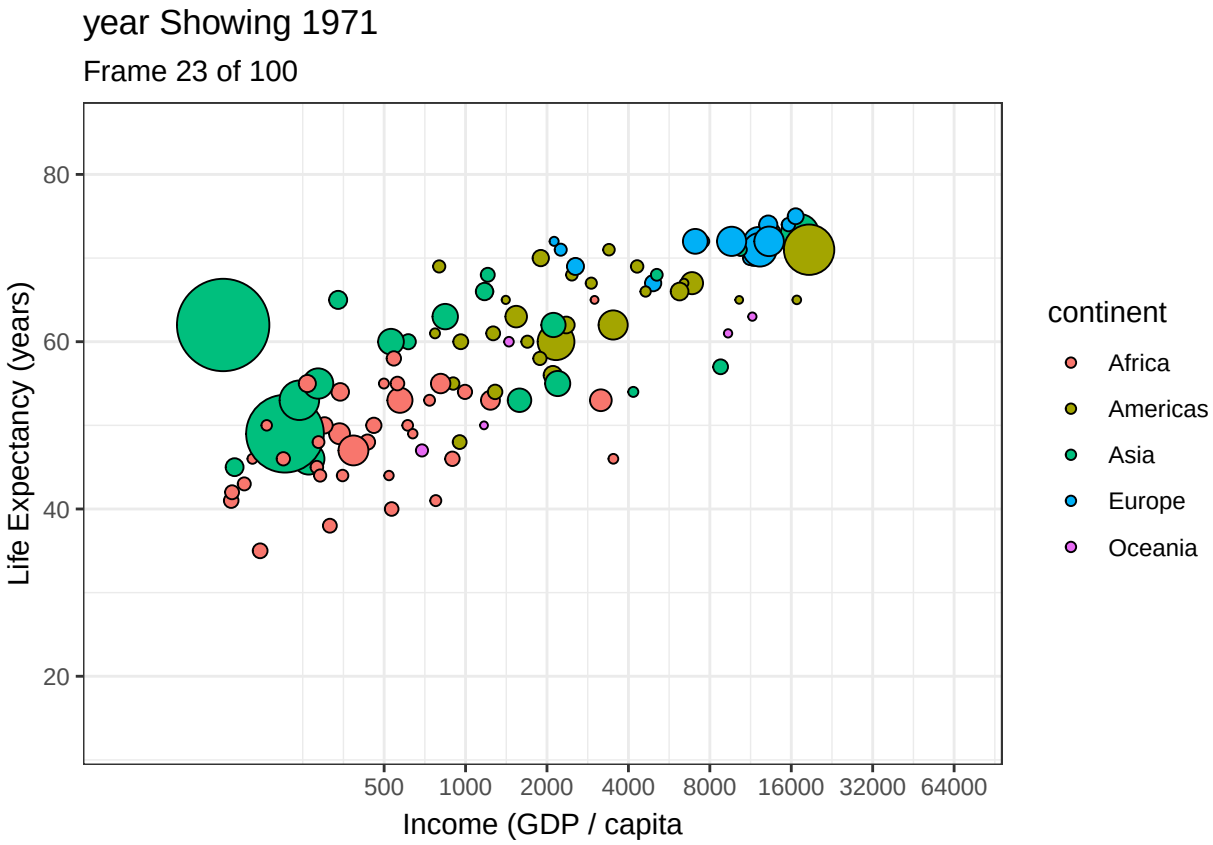

year Showing 1968
Frame 18 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

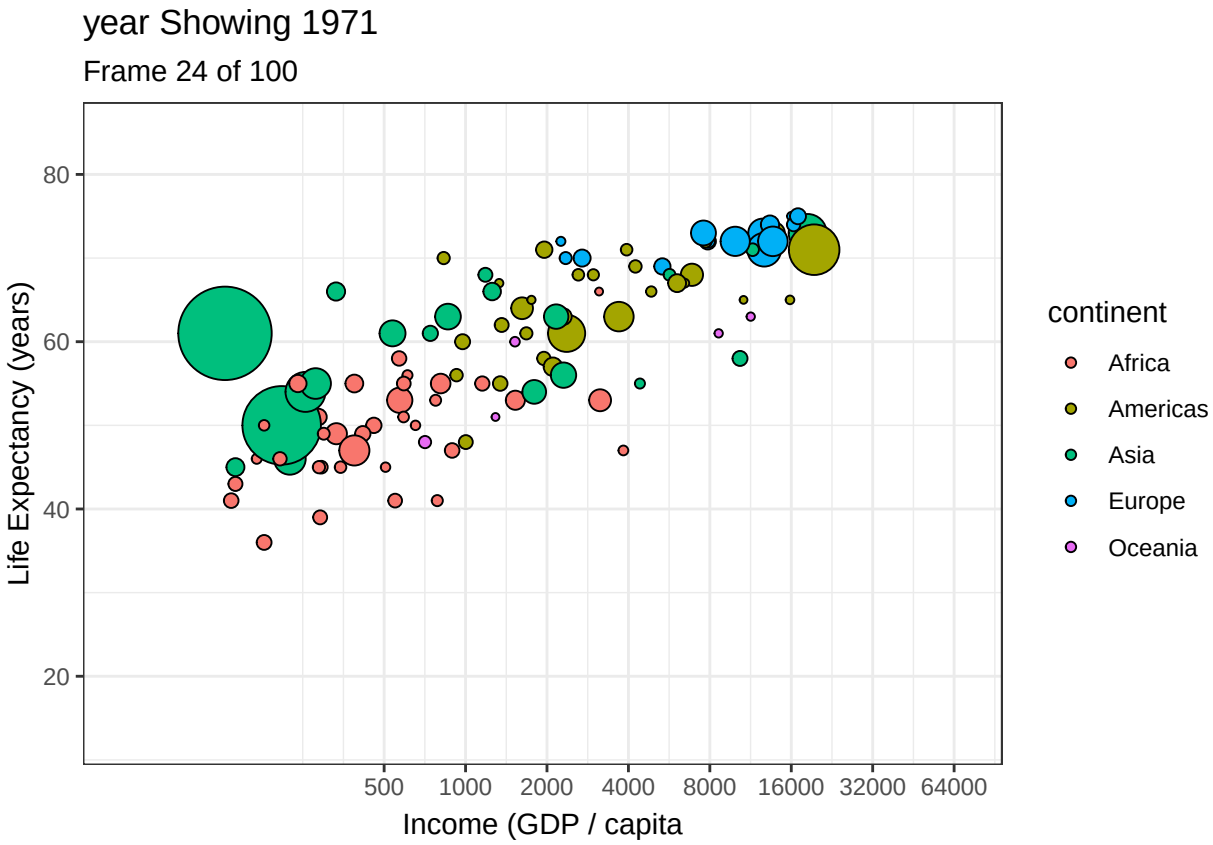

year Showing 1969
Frame 19 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

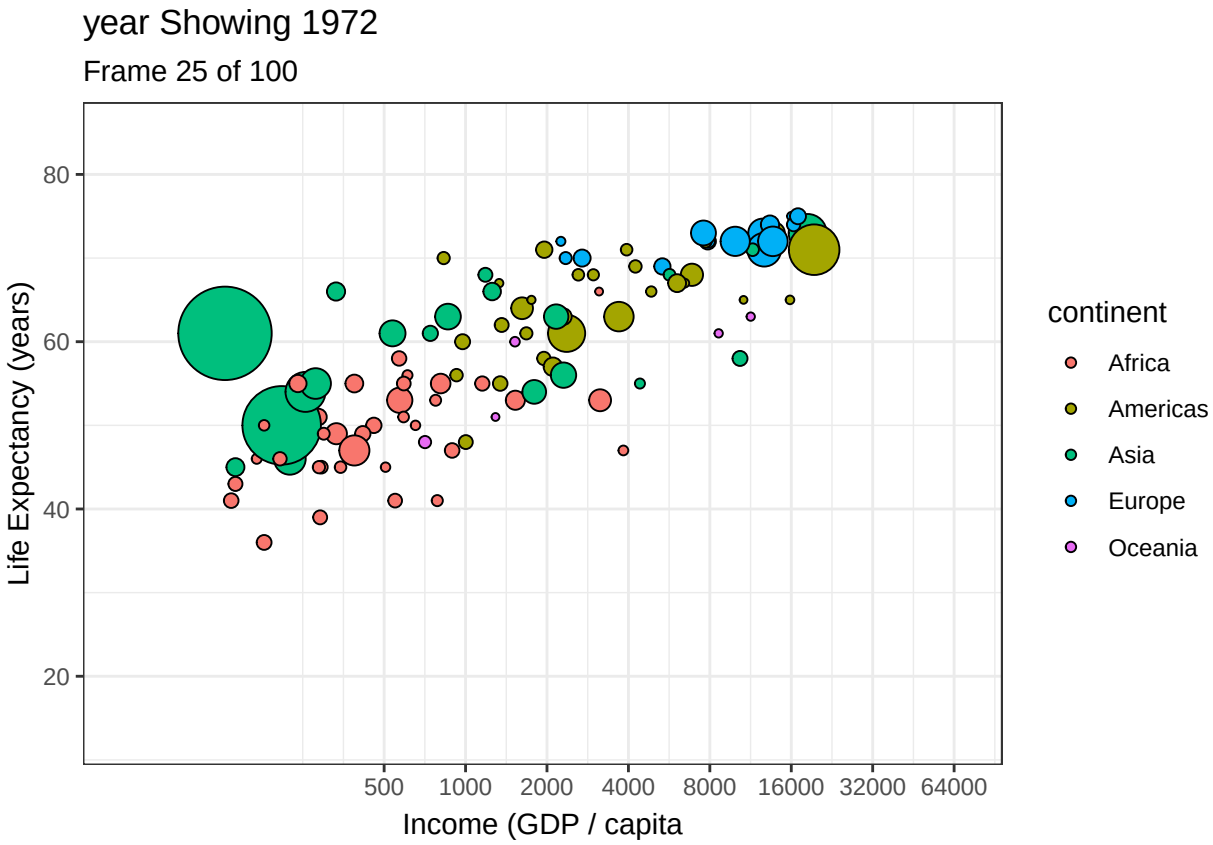

year Showing 1969
Frame 20 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

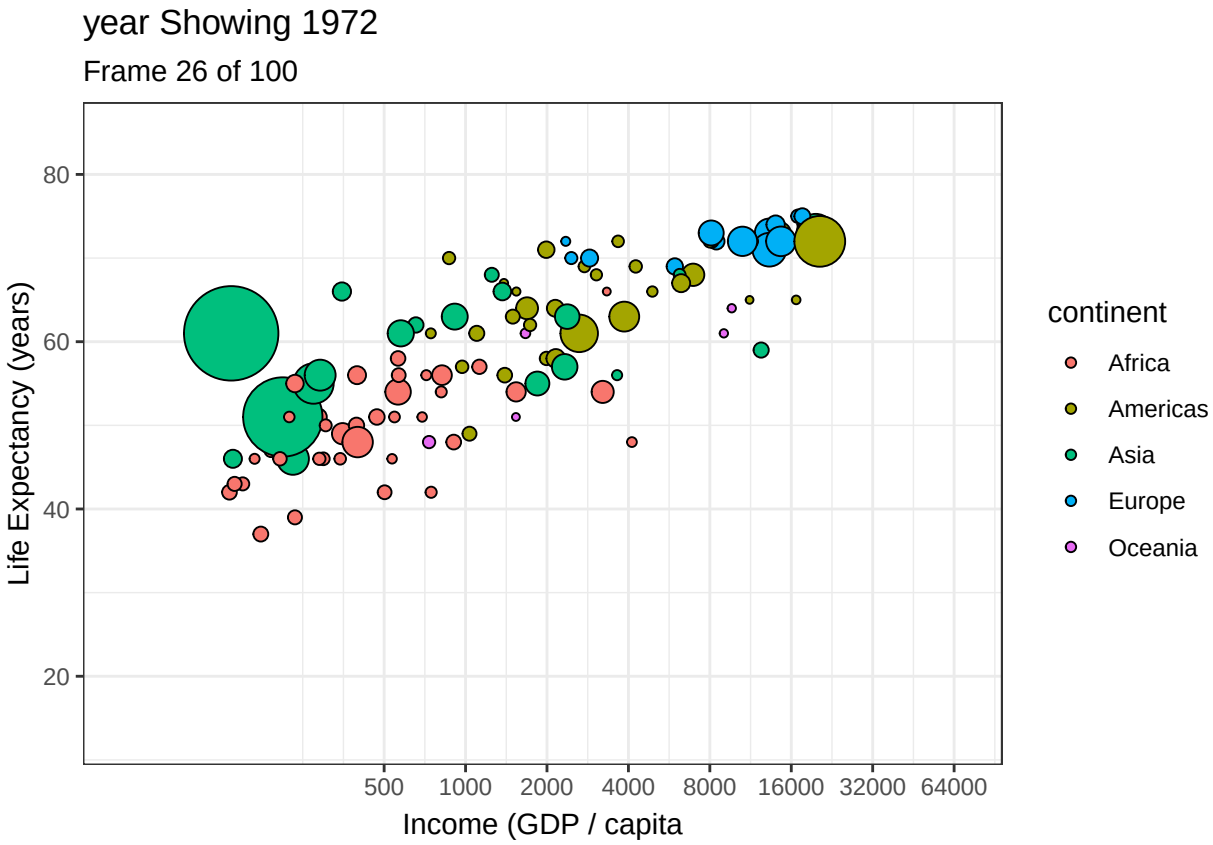

year Showing 1970
Frame 21 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

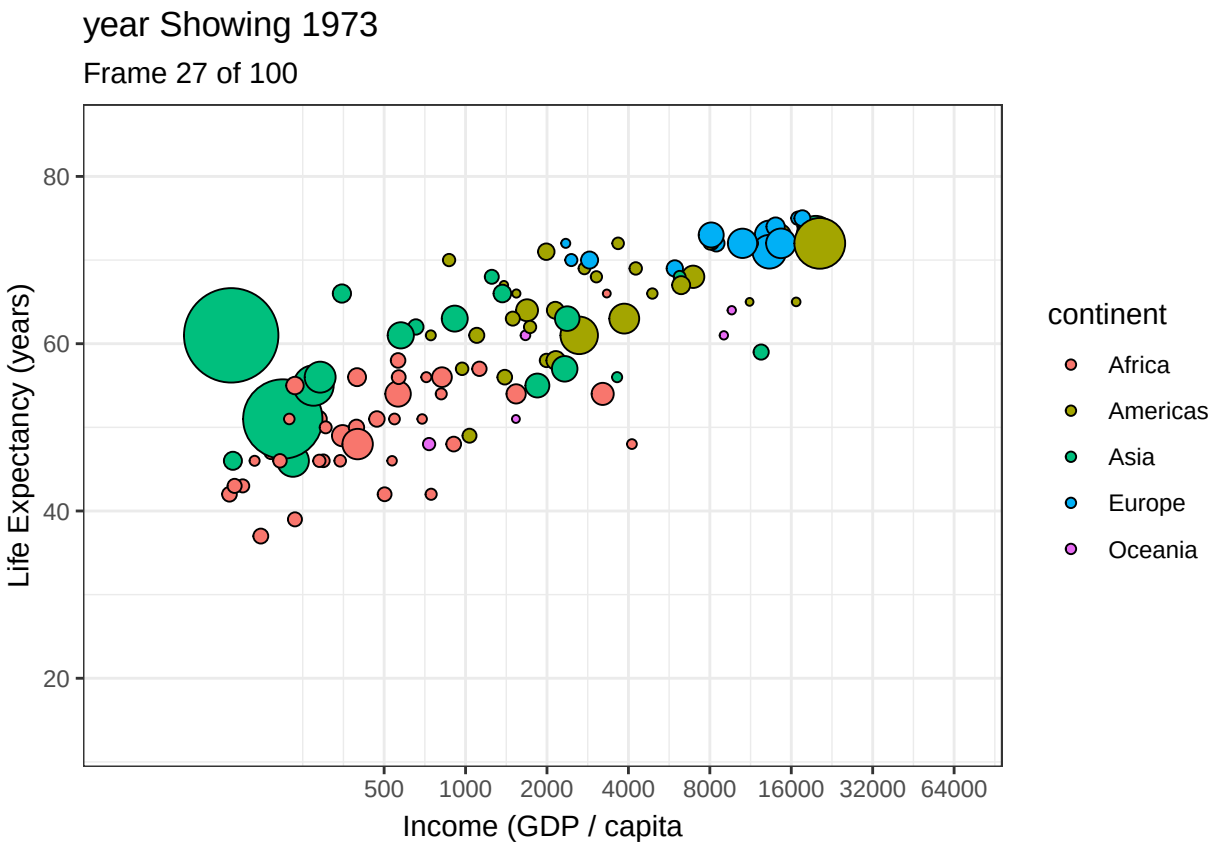

year Showing 1970
Frame 22 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

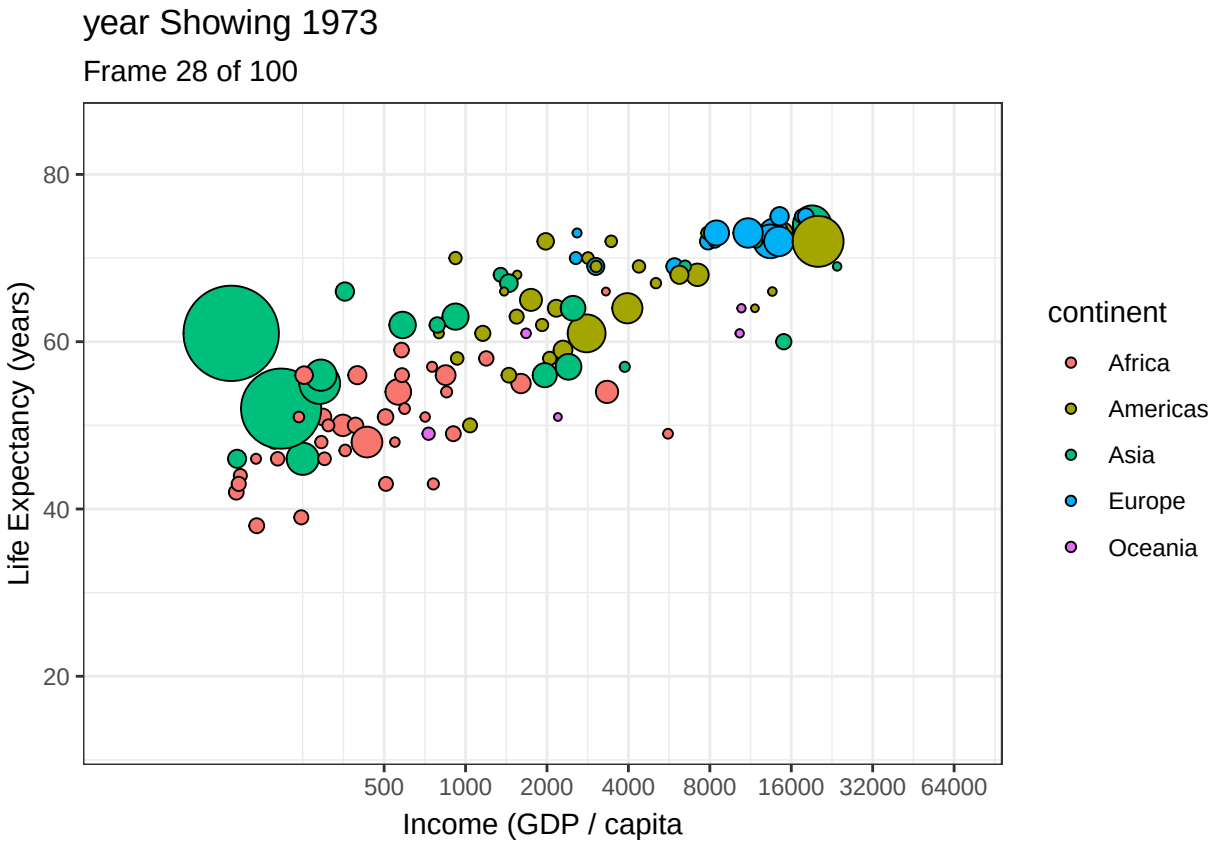

year Showing 1971
Frame 23 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

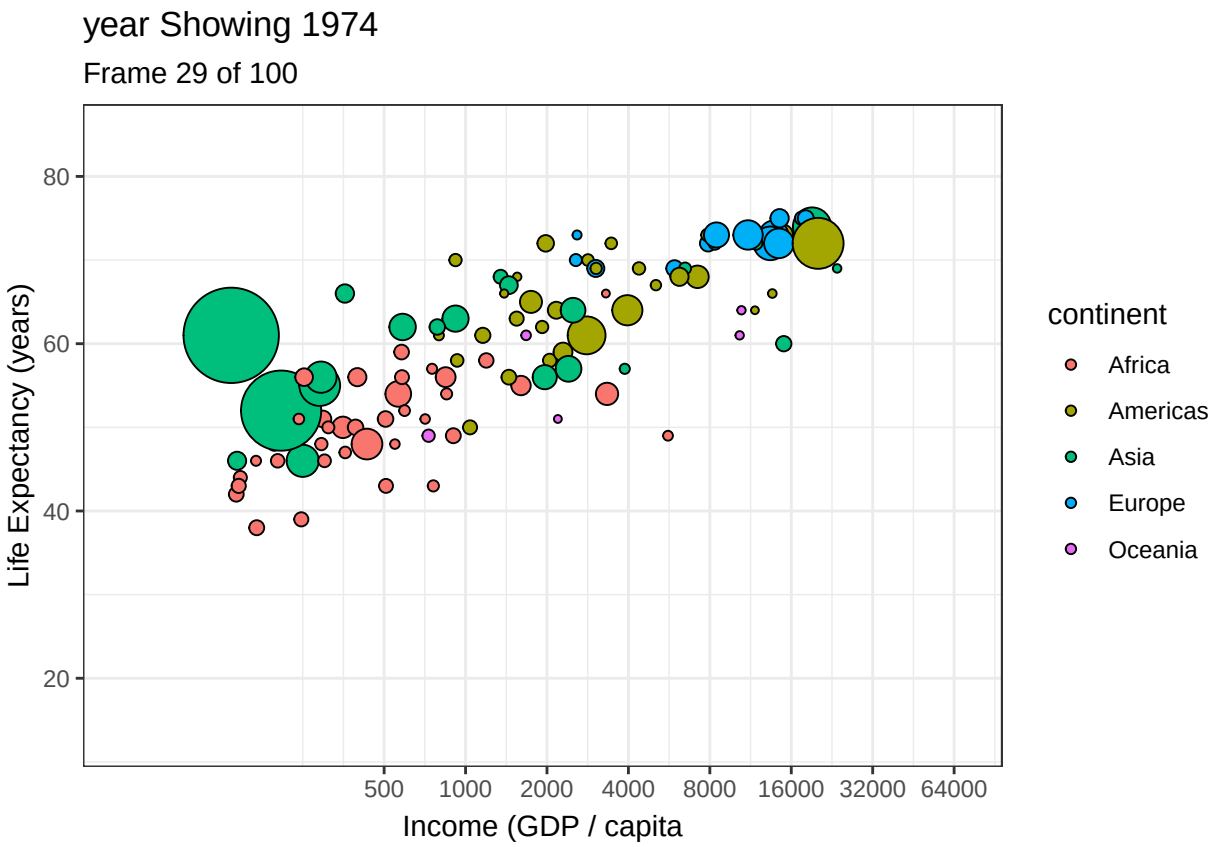

year Showing 1971
Frame 24 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

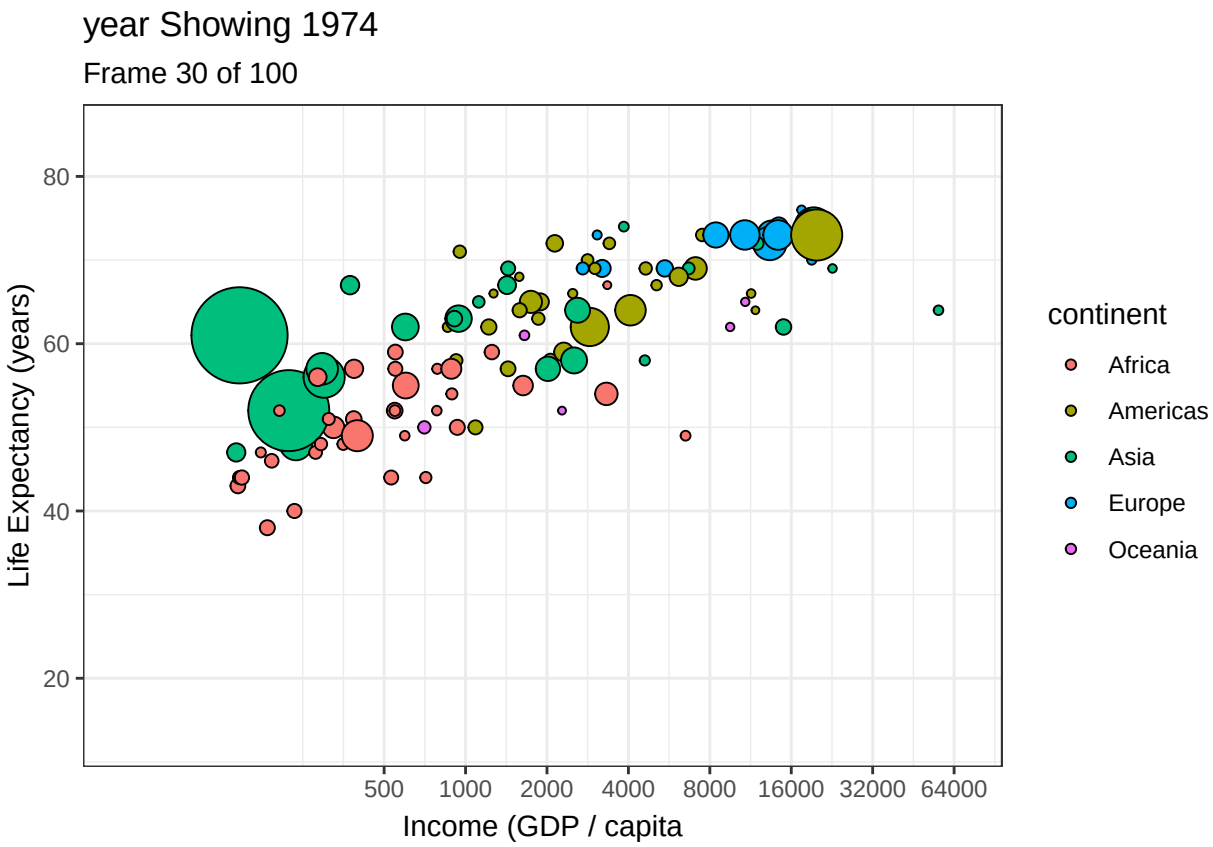

year Showing 1972
Frame 25 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

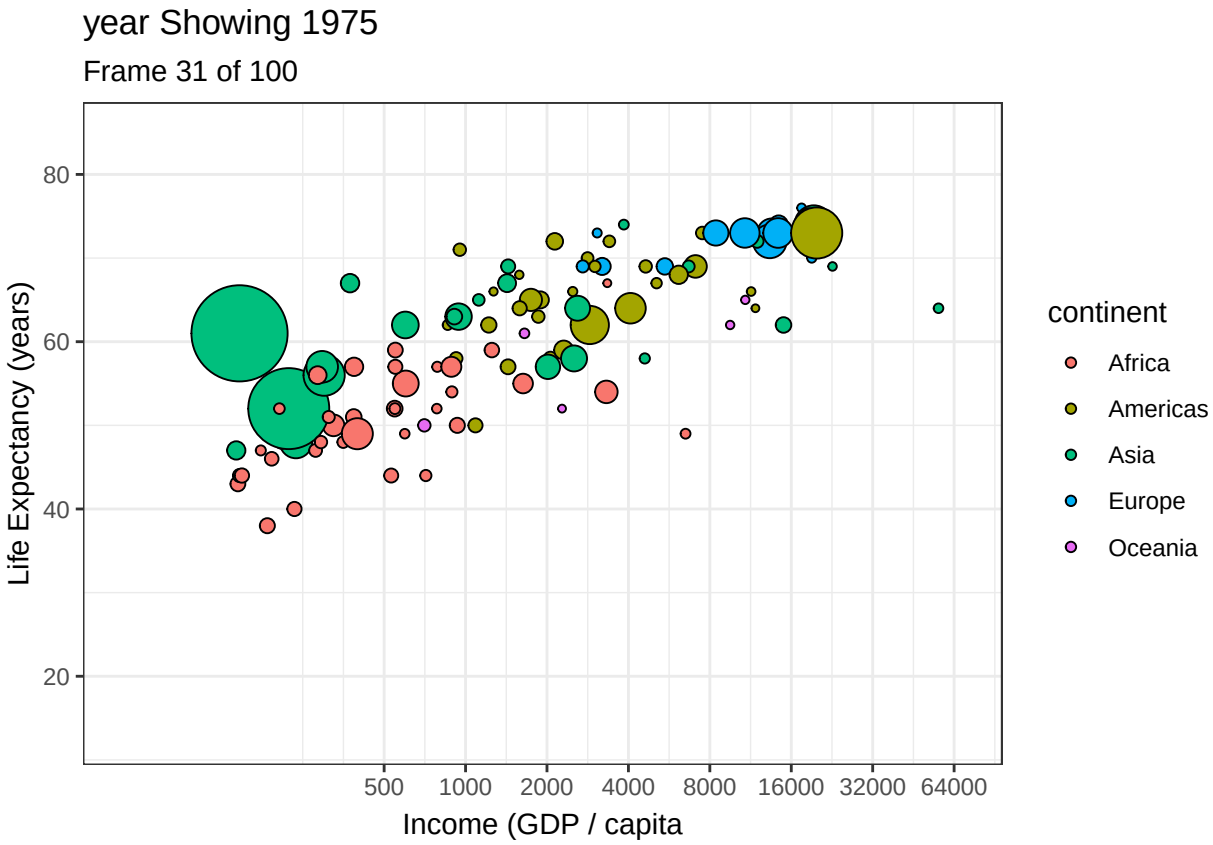

year Showing 1972
Frame 26 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

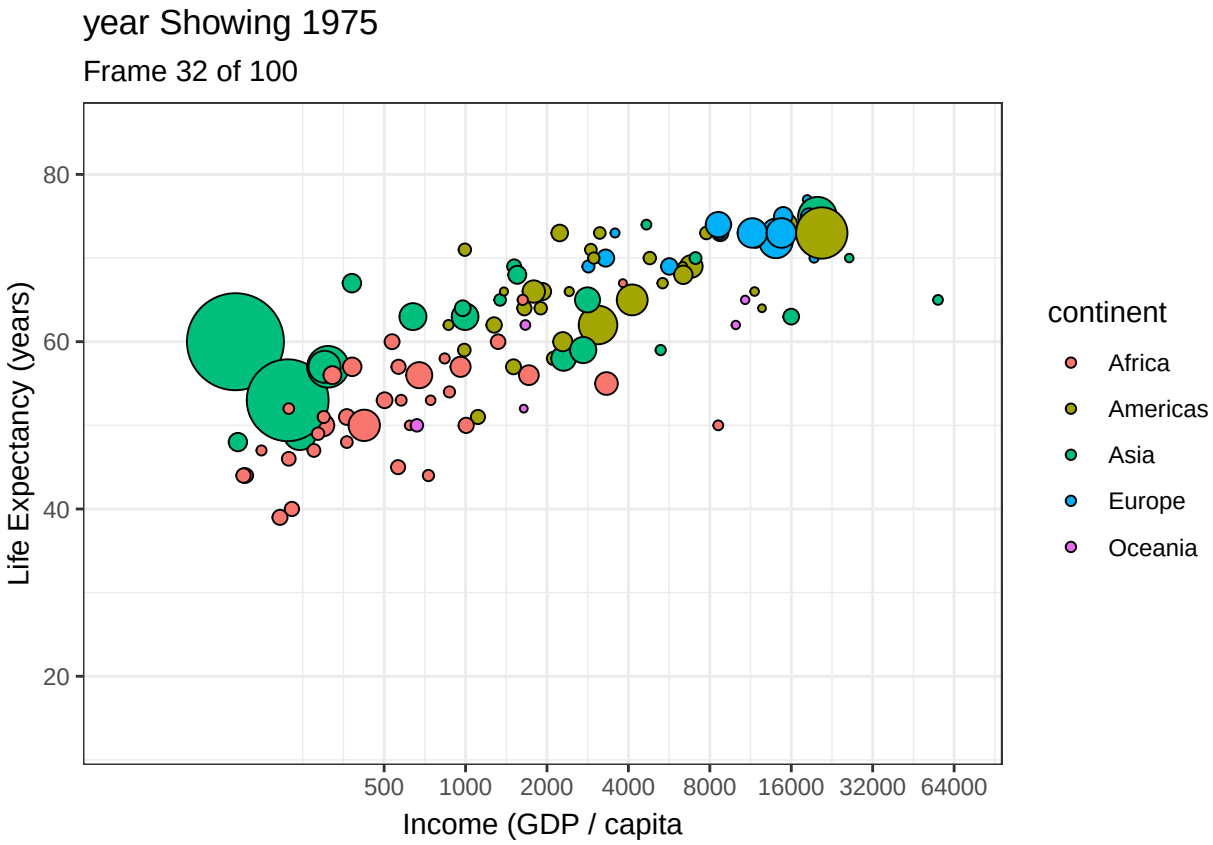

year Showing 1973
Frame 27 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

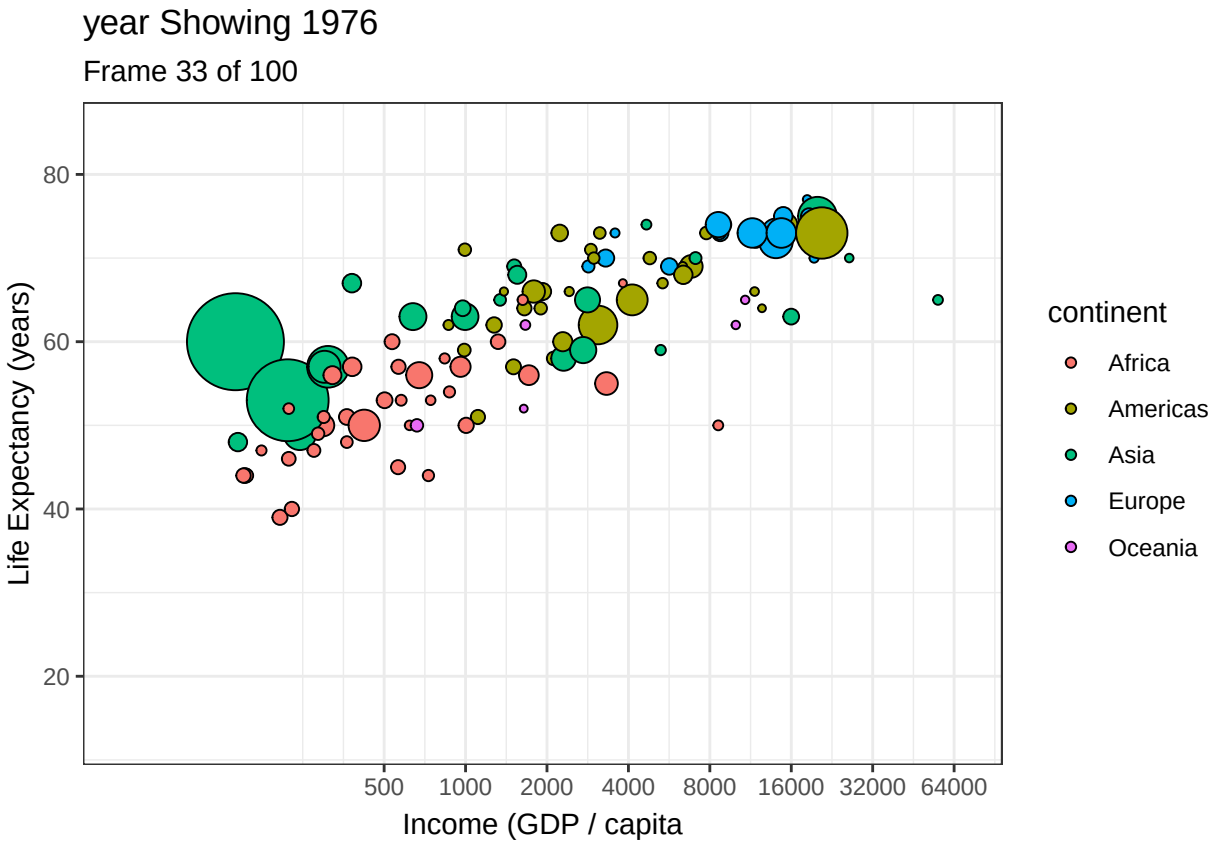

year Showing 1973
Frame 28 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

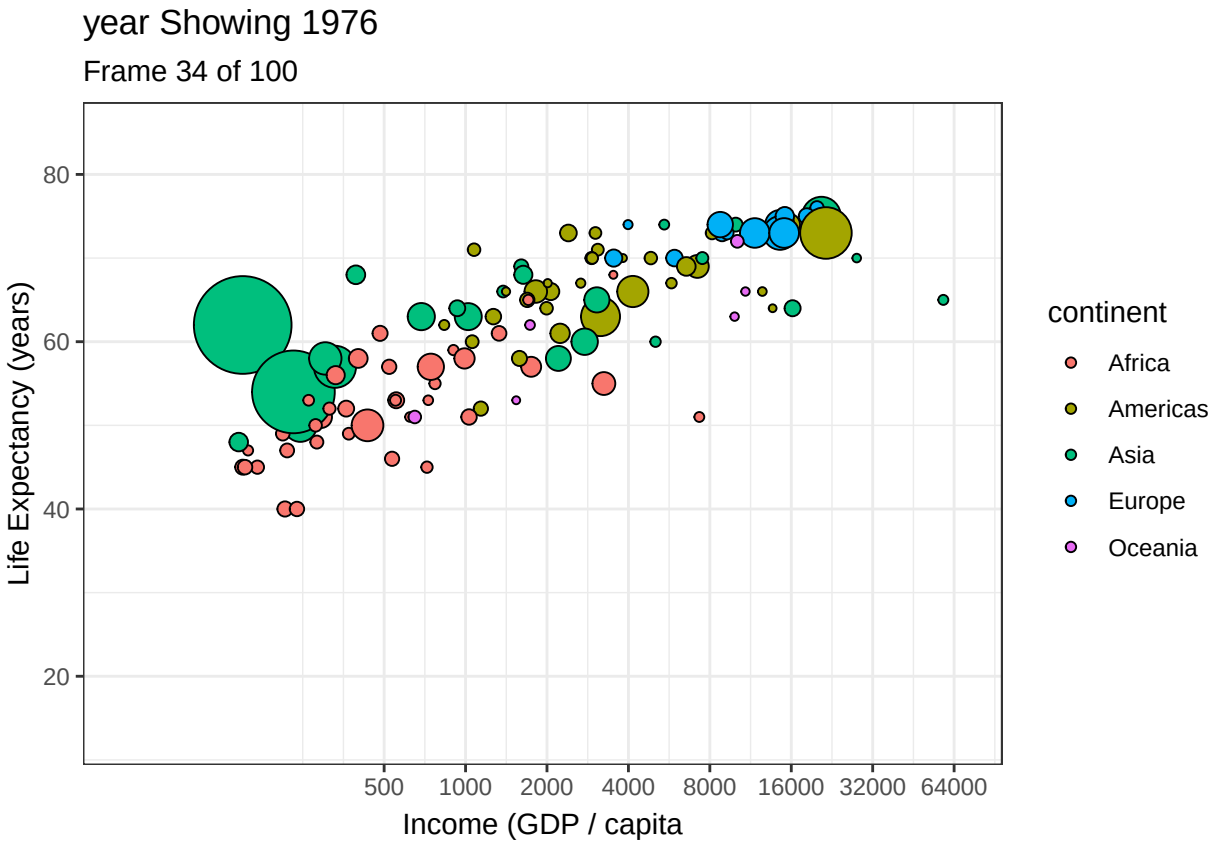

year Showing 1974
Frame 29 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

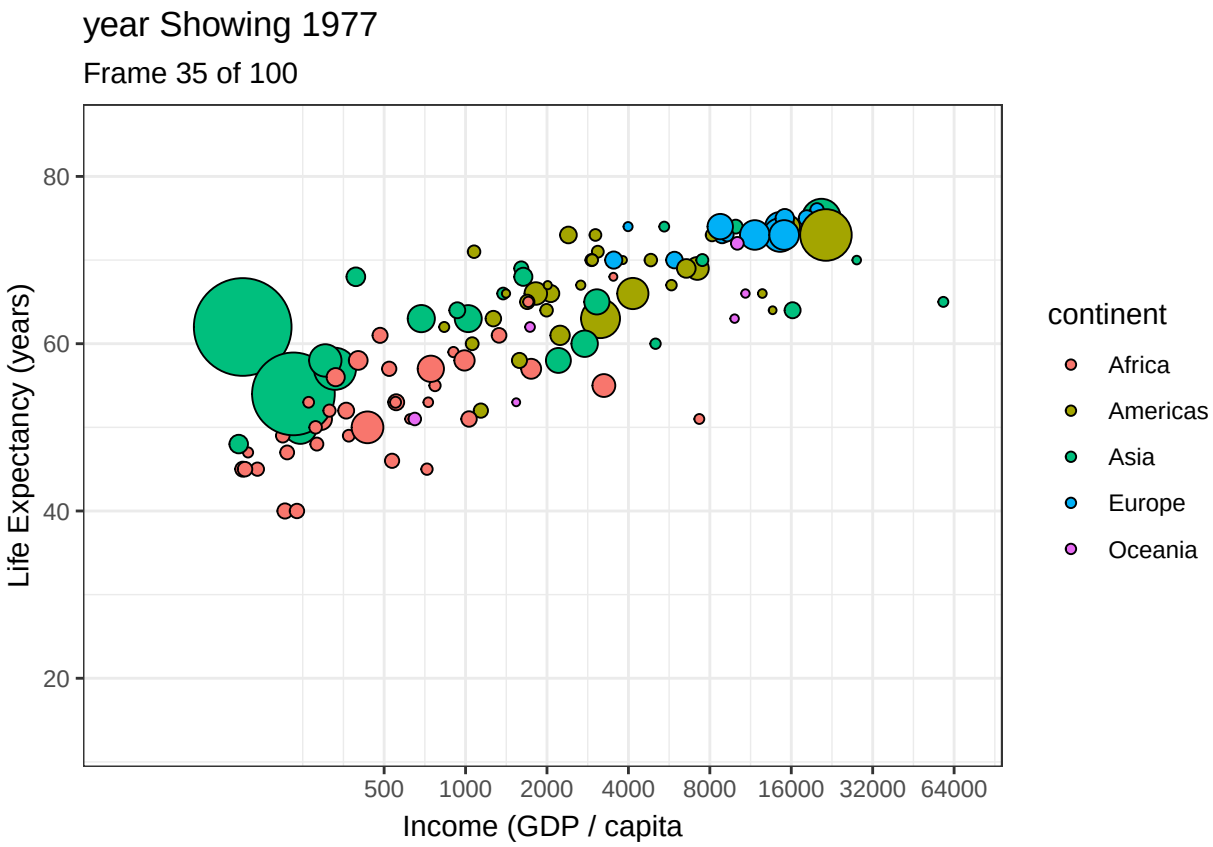

year Showing 1974
Frame 30 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

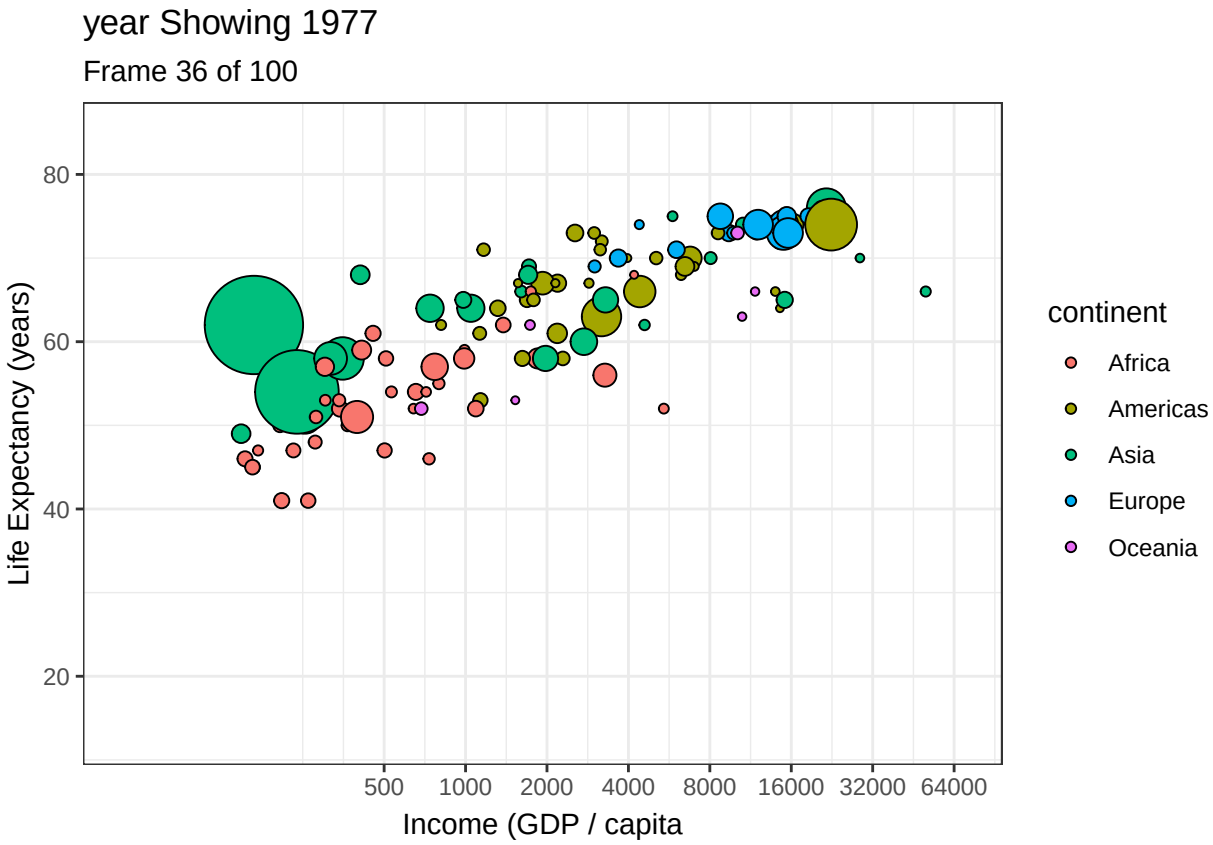

year Showing 1975
Frame 31 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

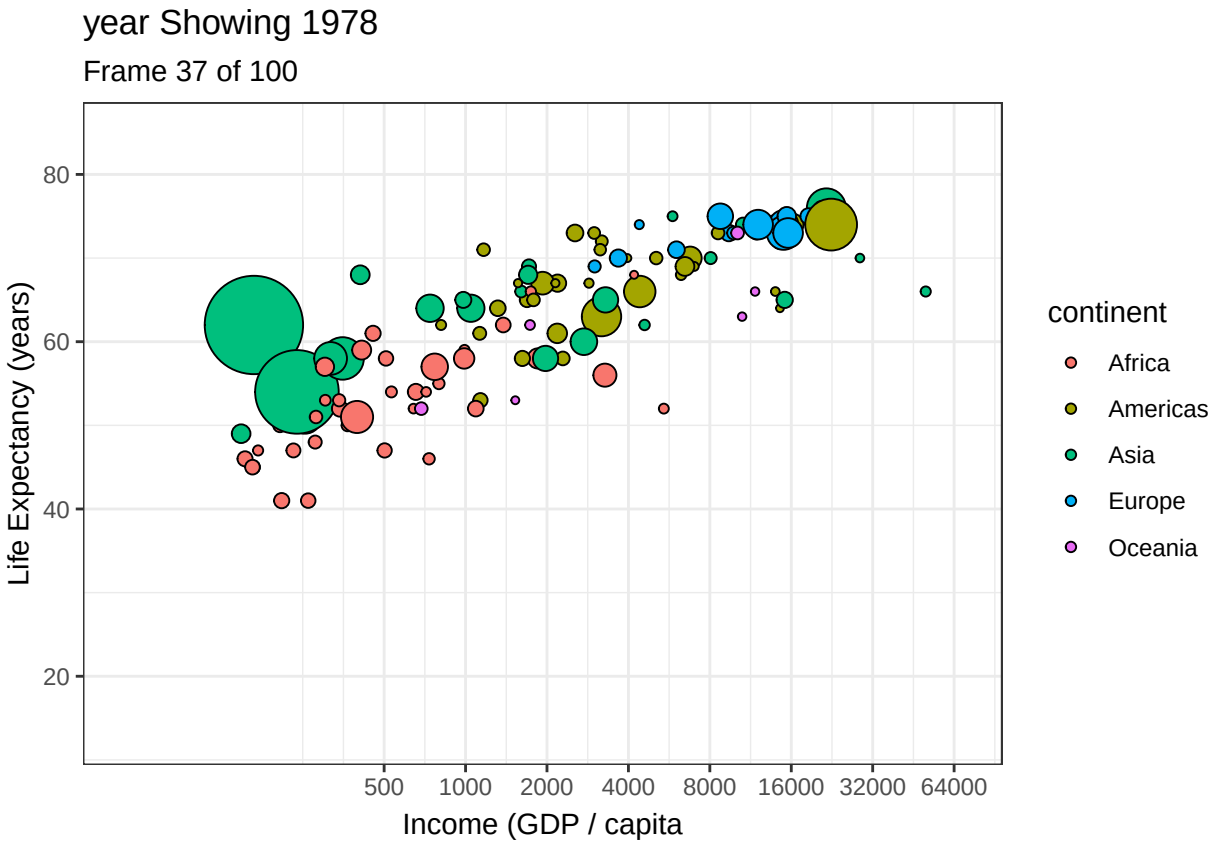

year Showing 1975
Frame 32 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

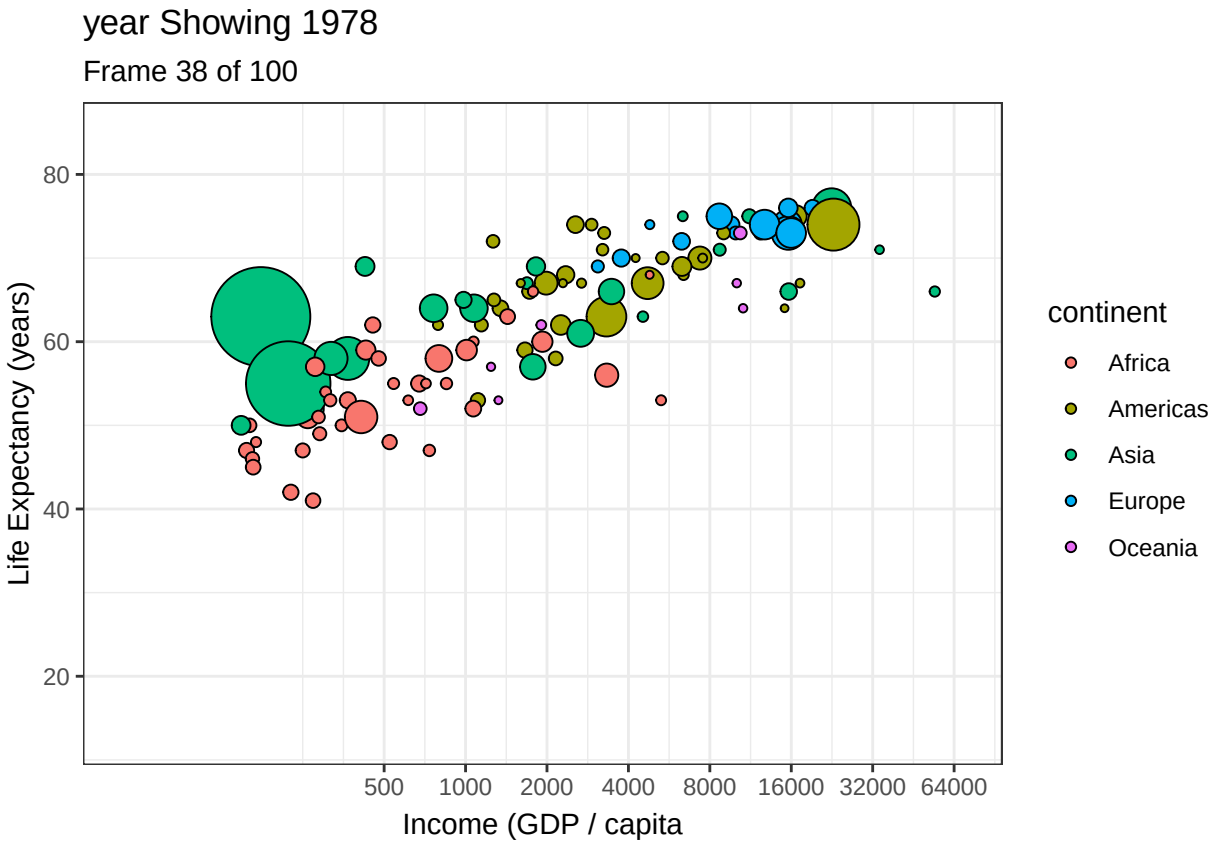

year Showing 1976
Frame 33 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

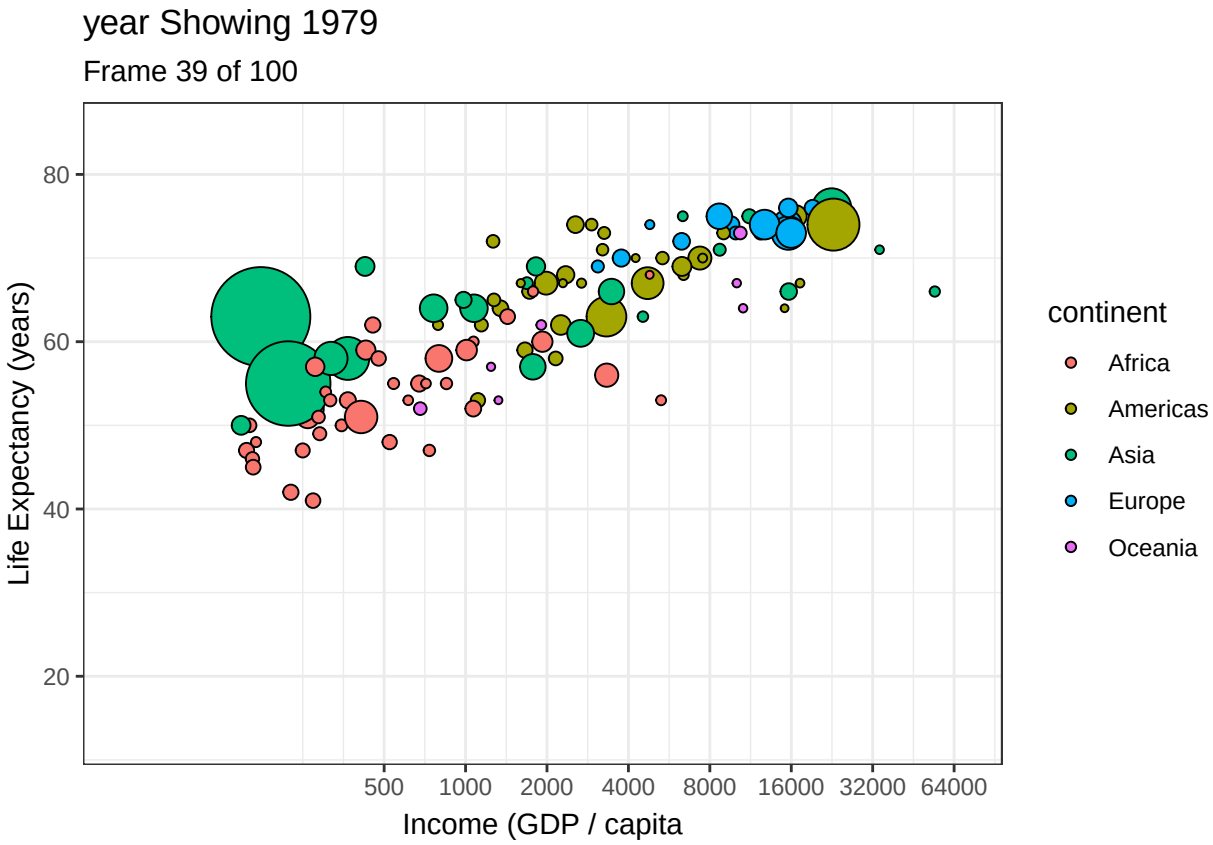

year Showing 1976
Frame 34 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

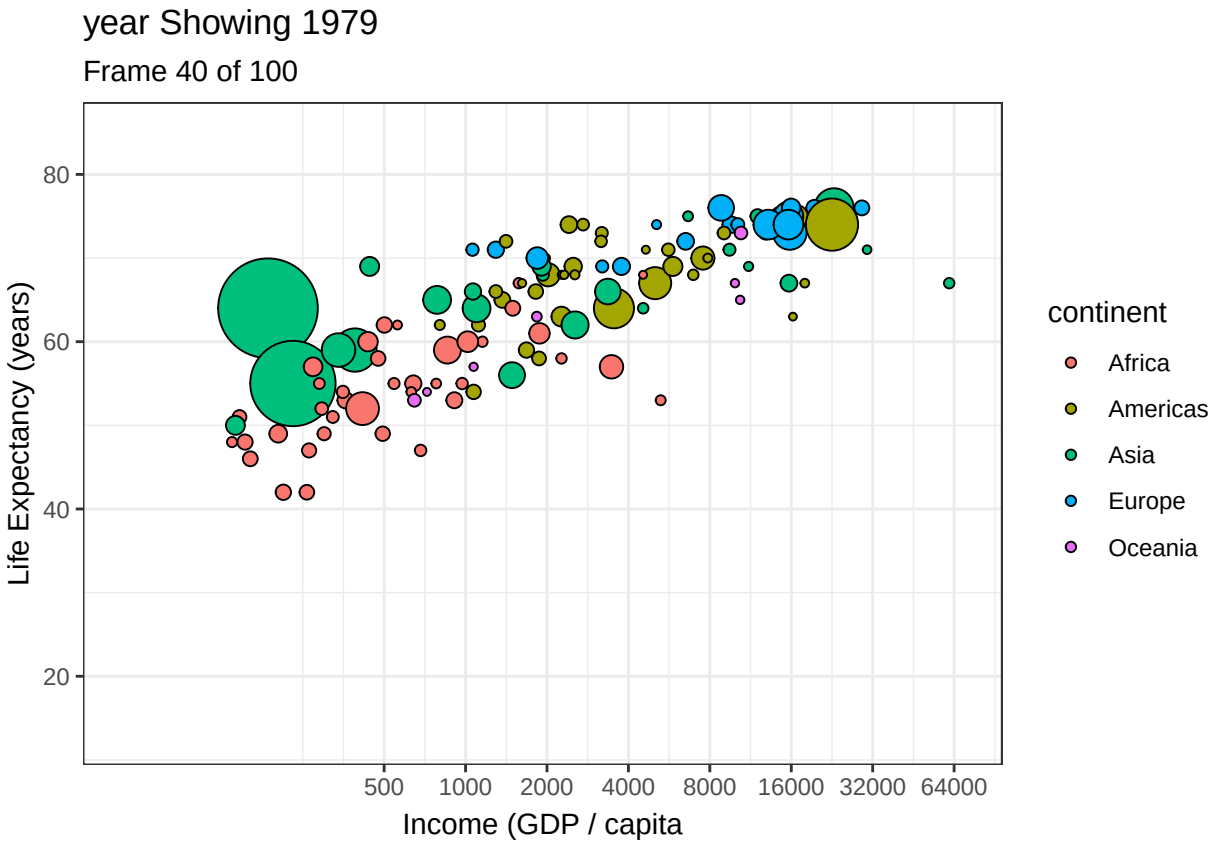

year Showing 1977
Frame 35 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

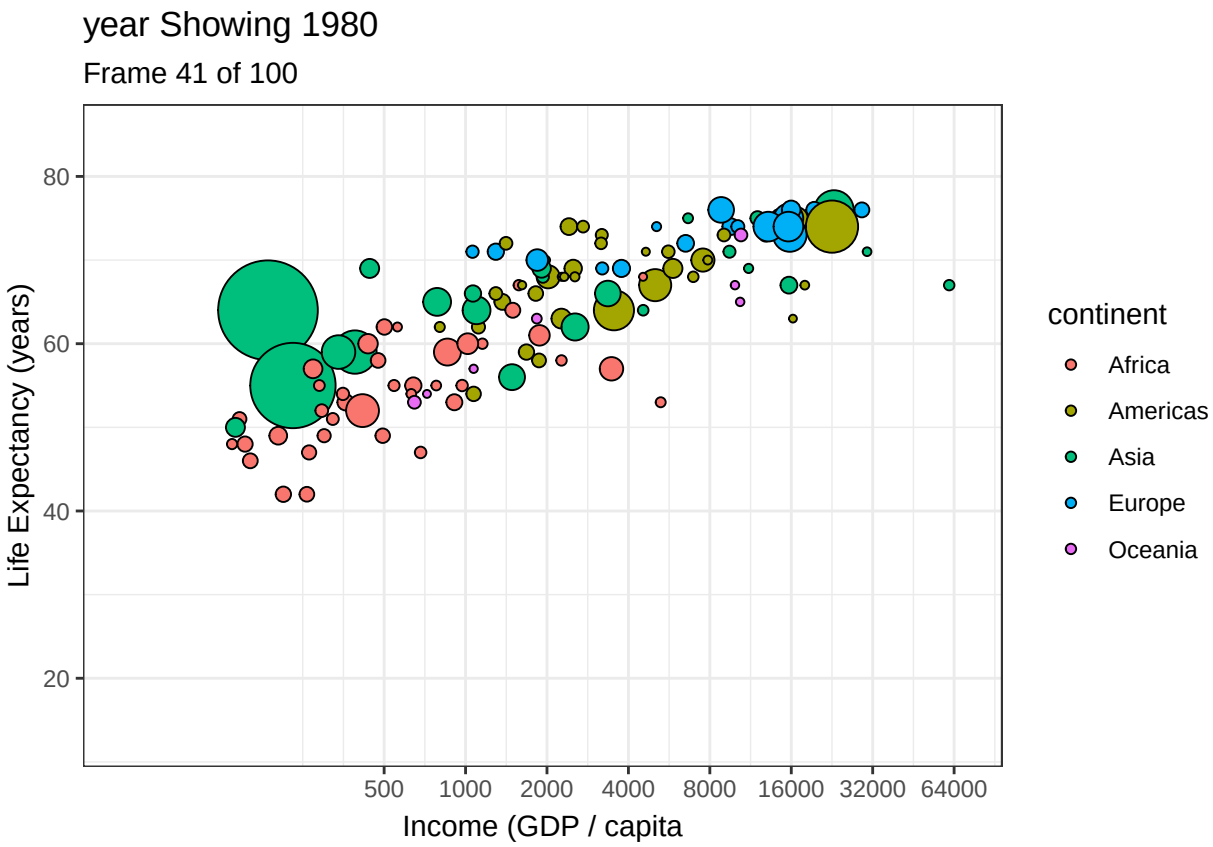

year Showing 1977
Frame 36 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

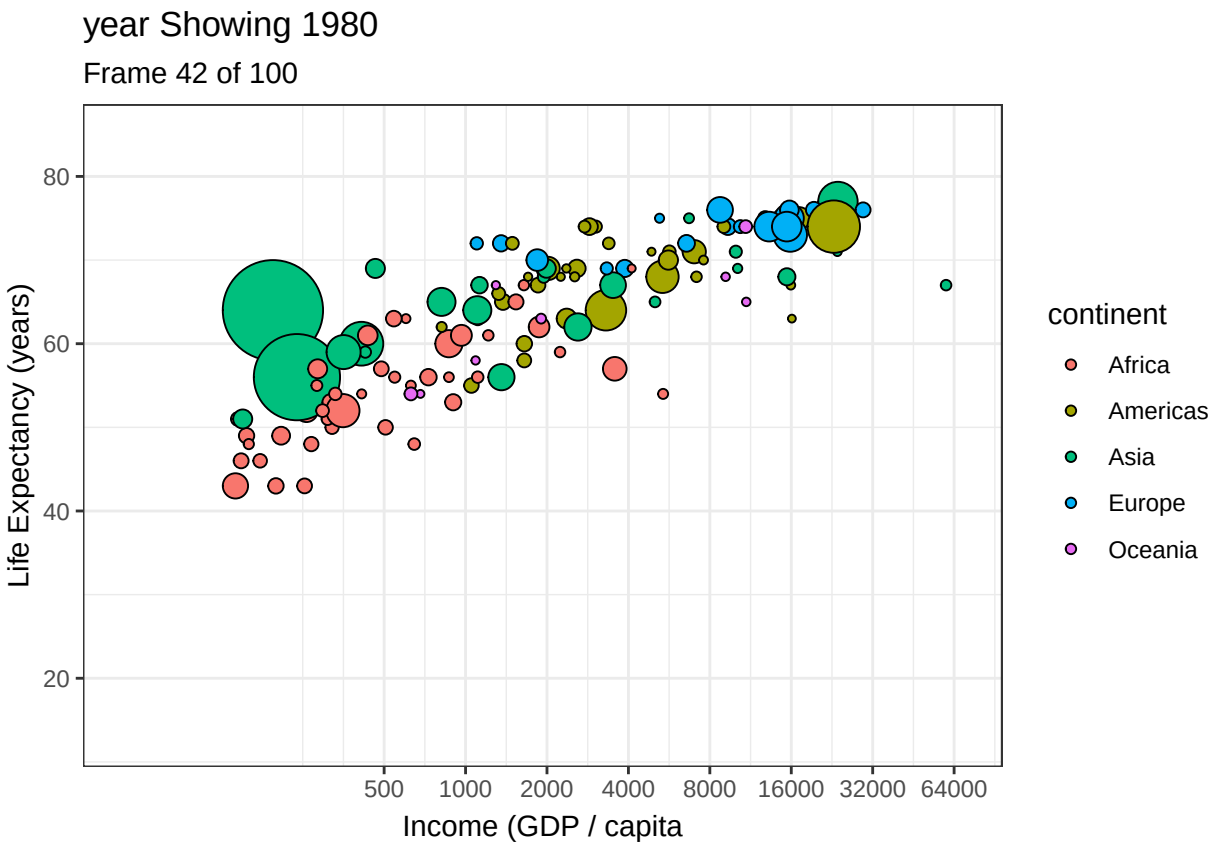

year Showing 1978
Frame 37 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

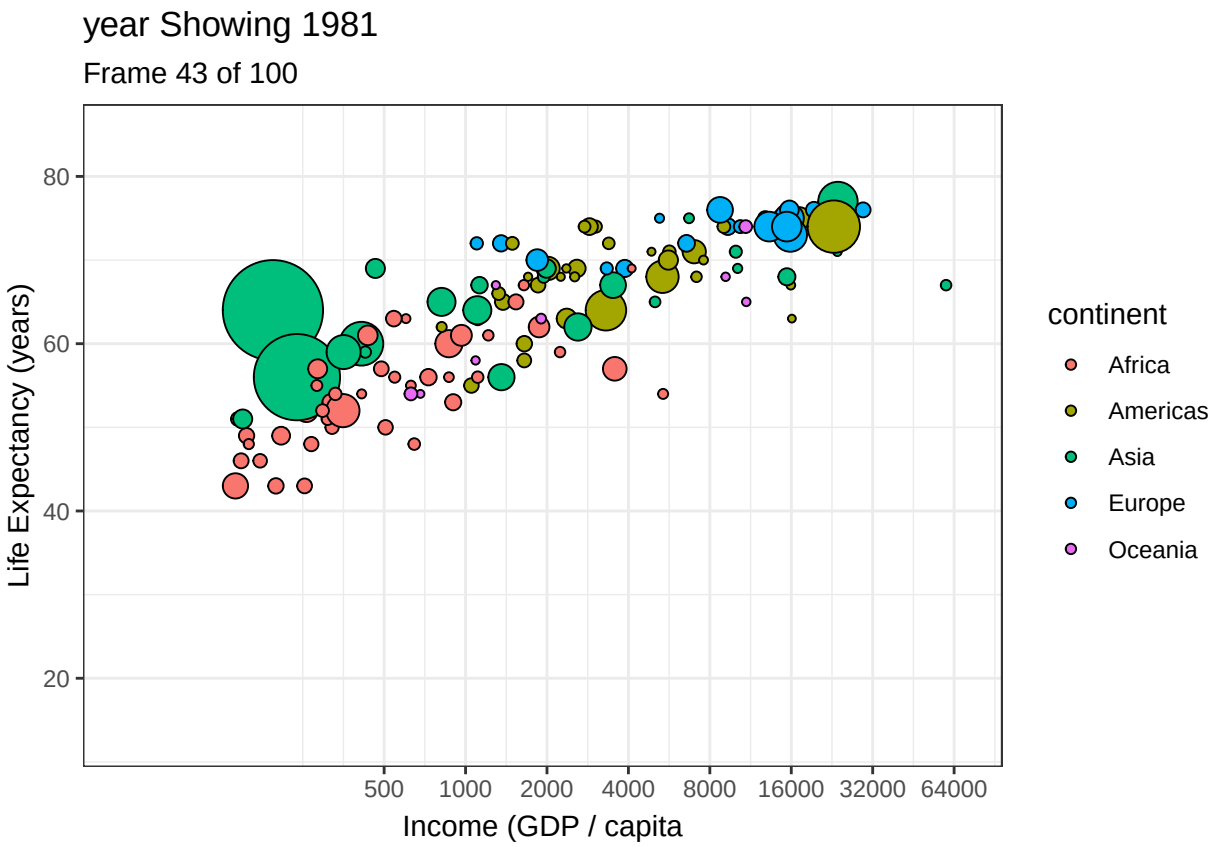

year Showing 1978
Frame 38 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

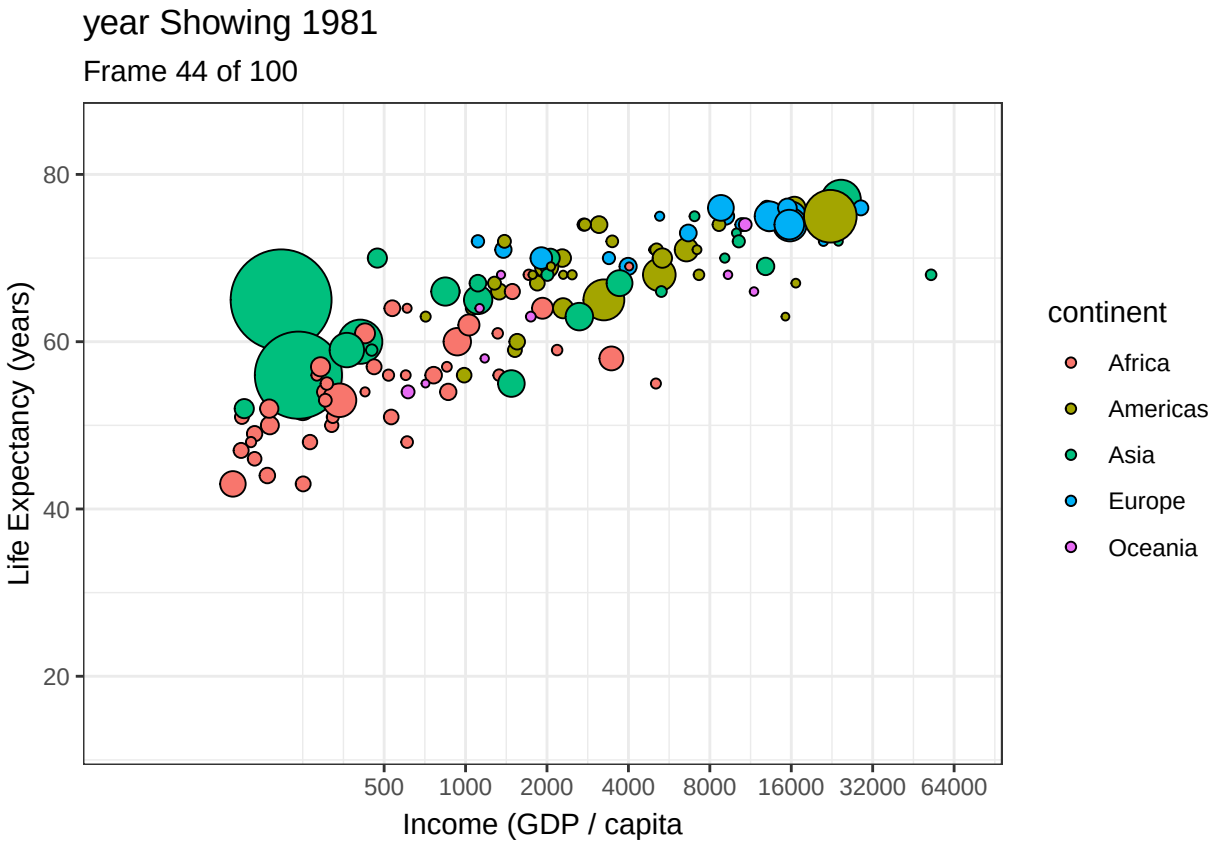

year Showing 1979
Frame 39 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

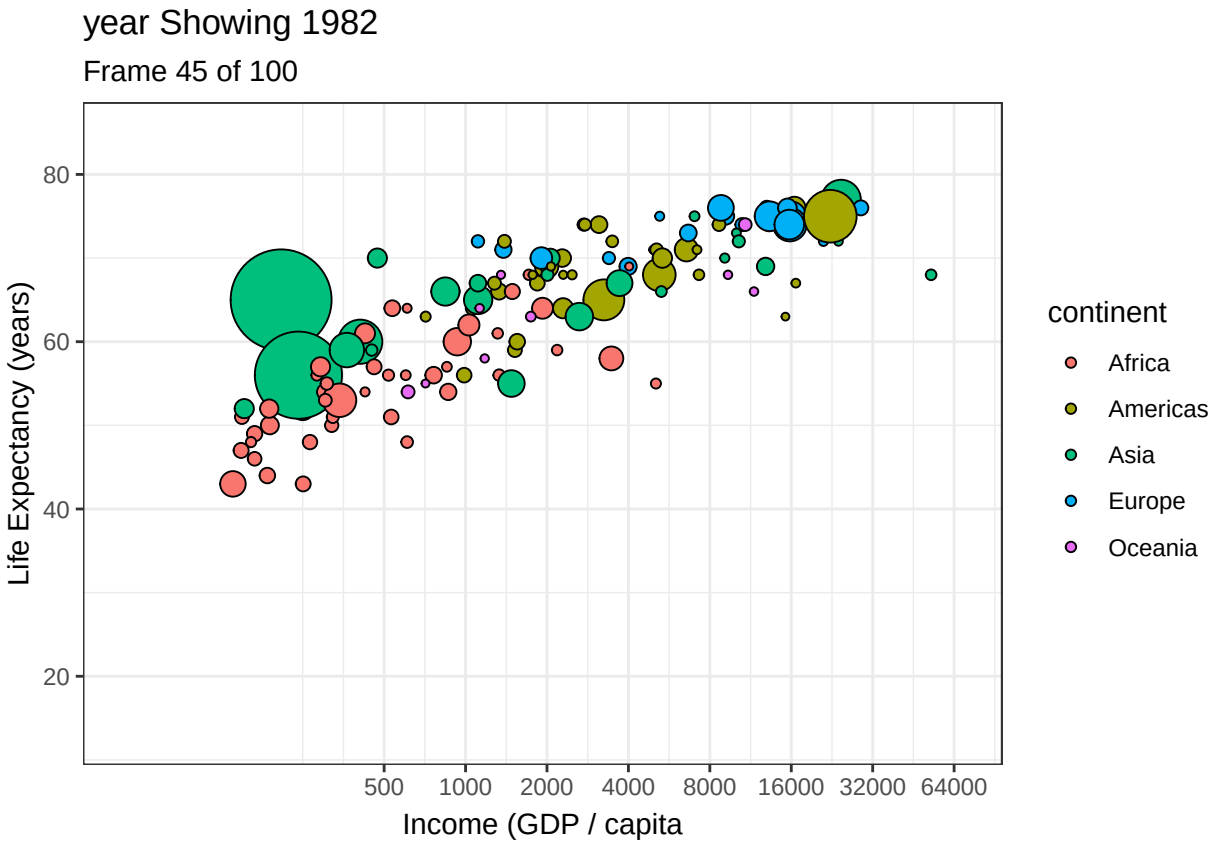

year Showing 1979
Frame 40 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

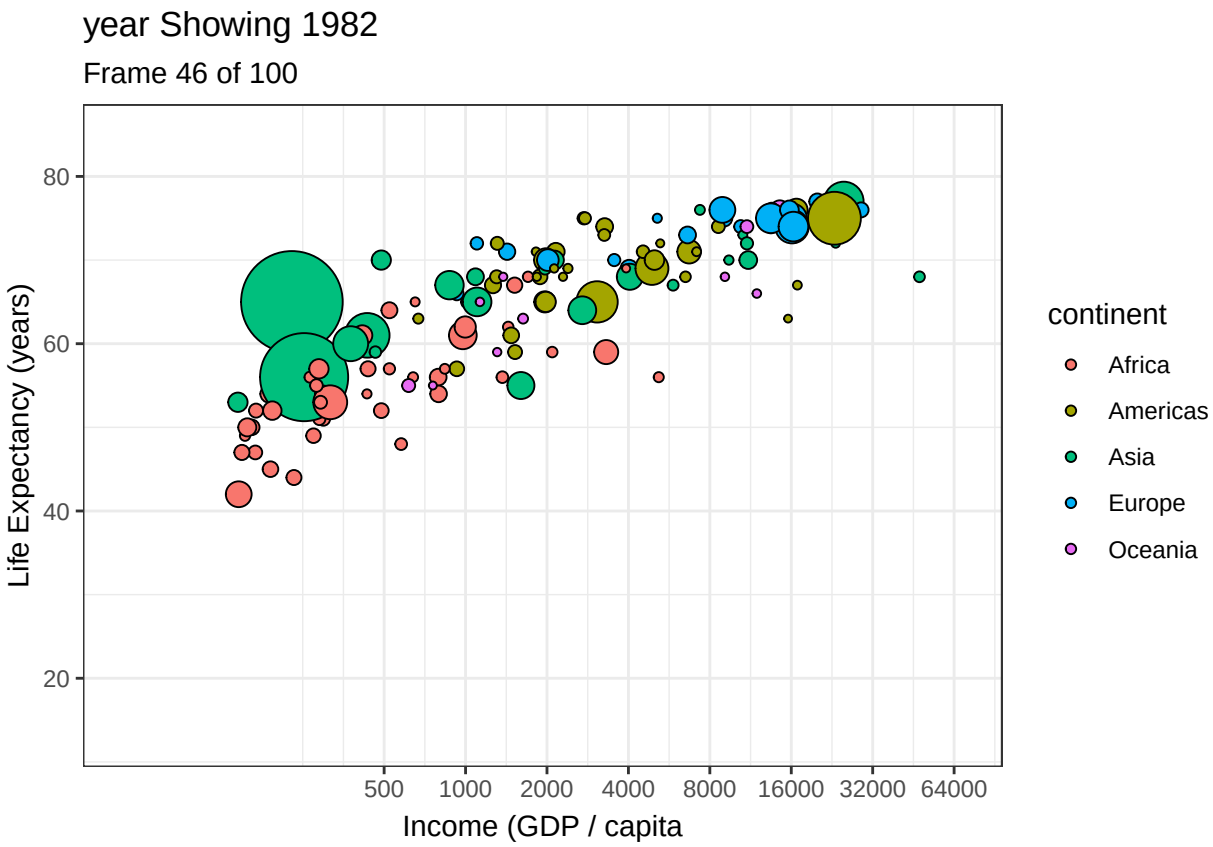

year Showing 1980
Frame 41 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

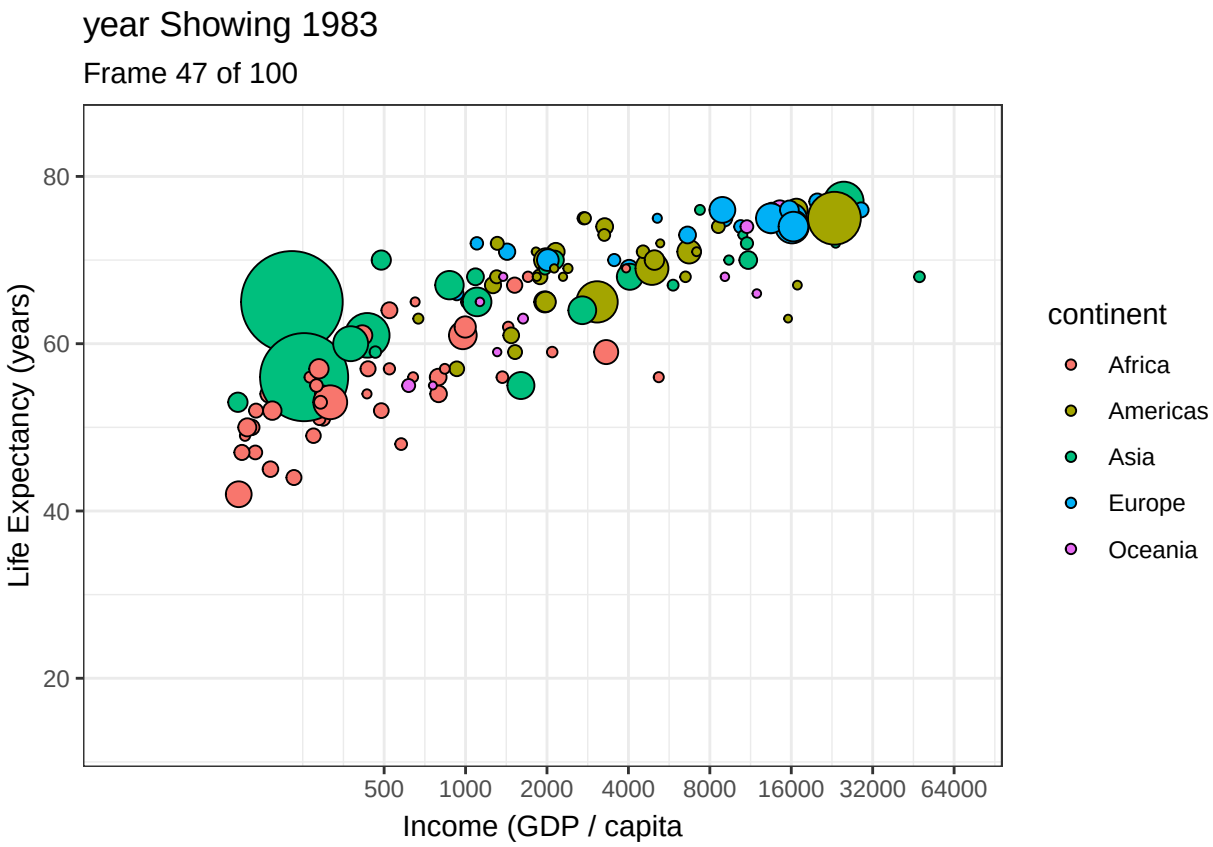

year Showing 1980
Frame 42 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

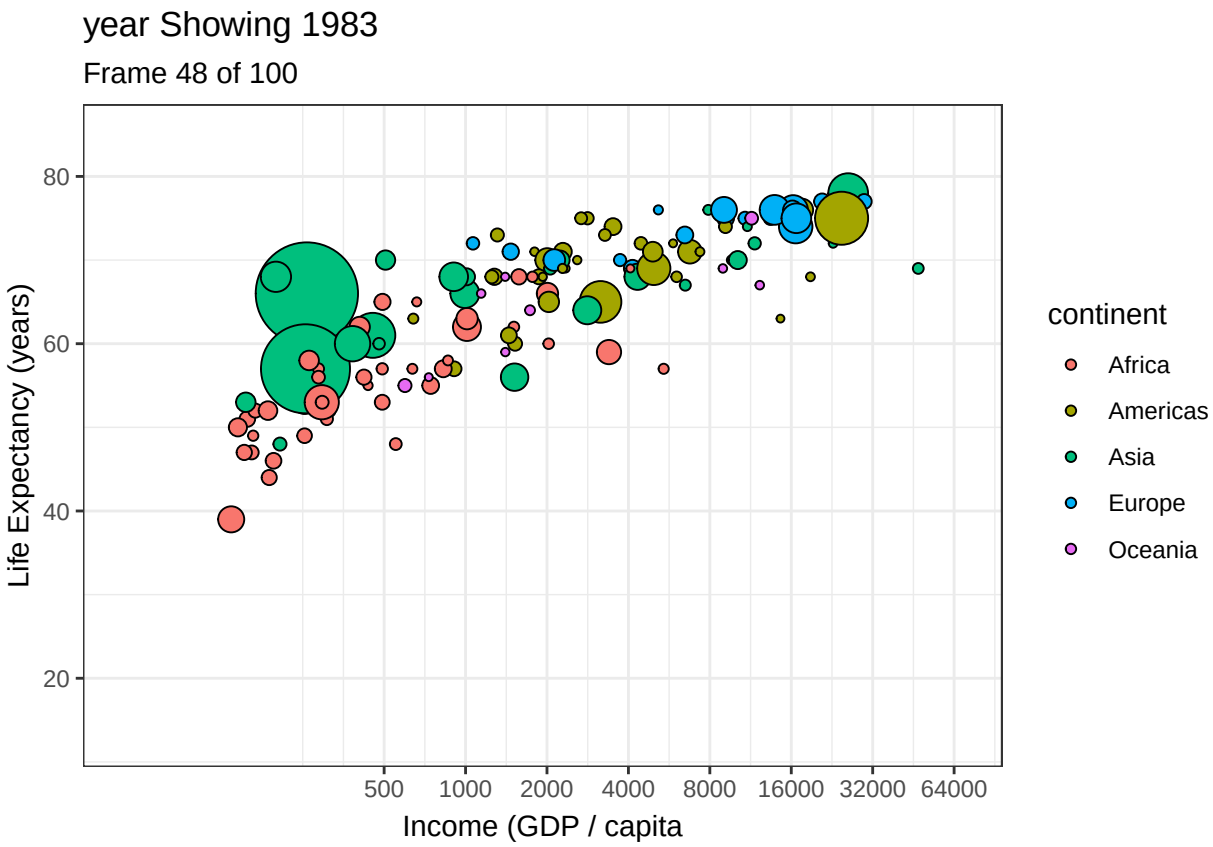

year Showing 1981
Frame 43 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

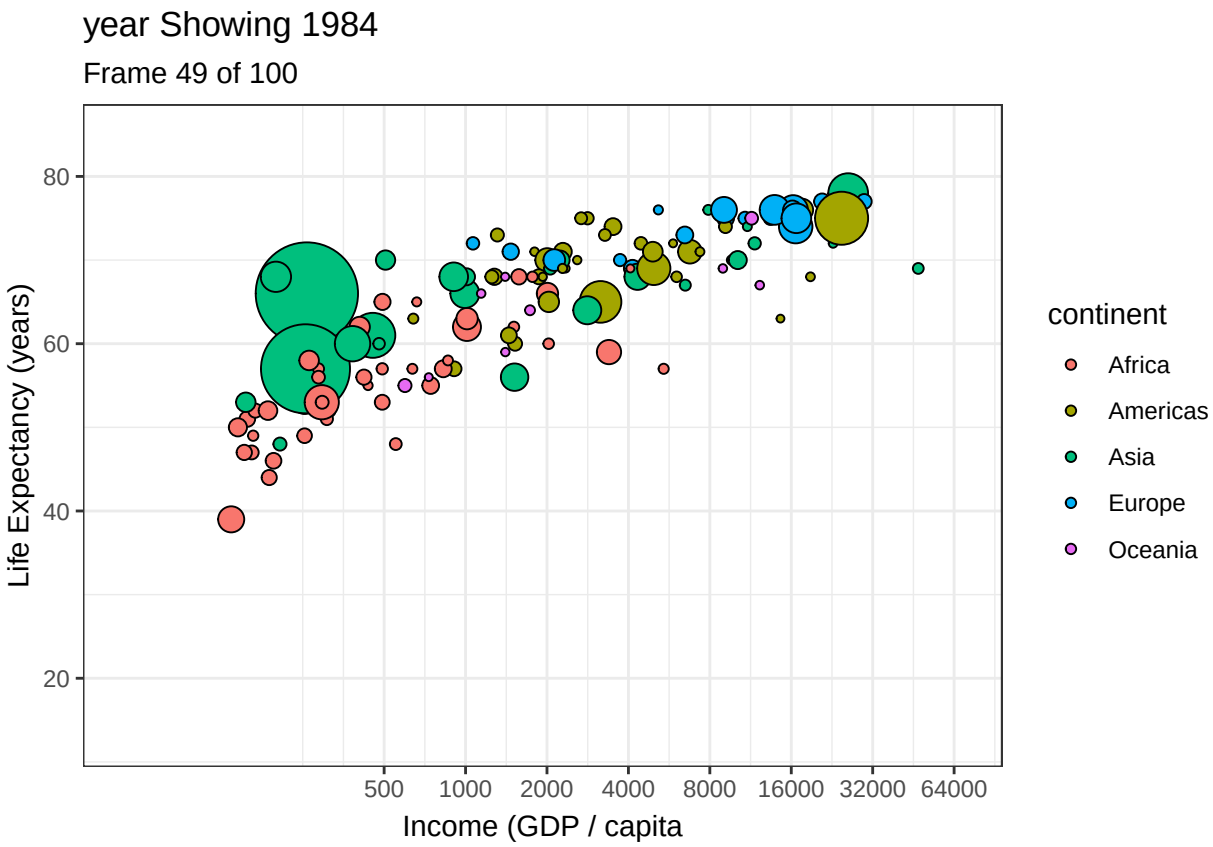

year Showing 1981
Frame 44 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

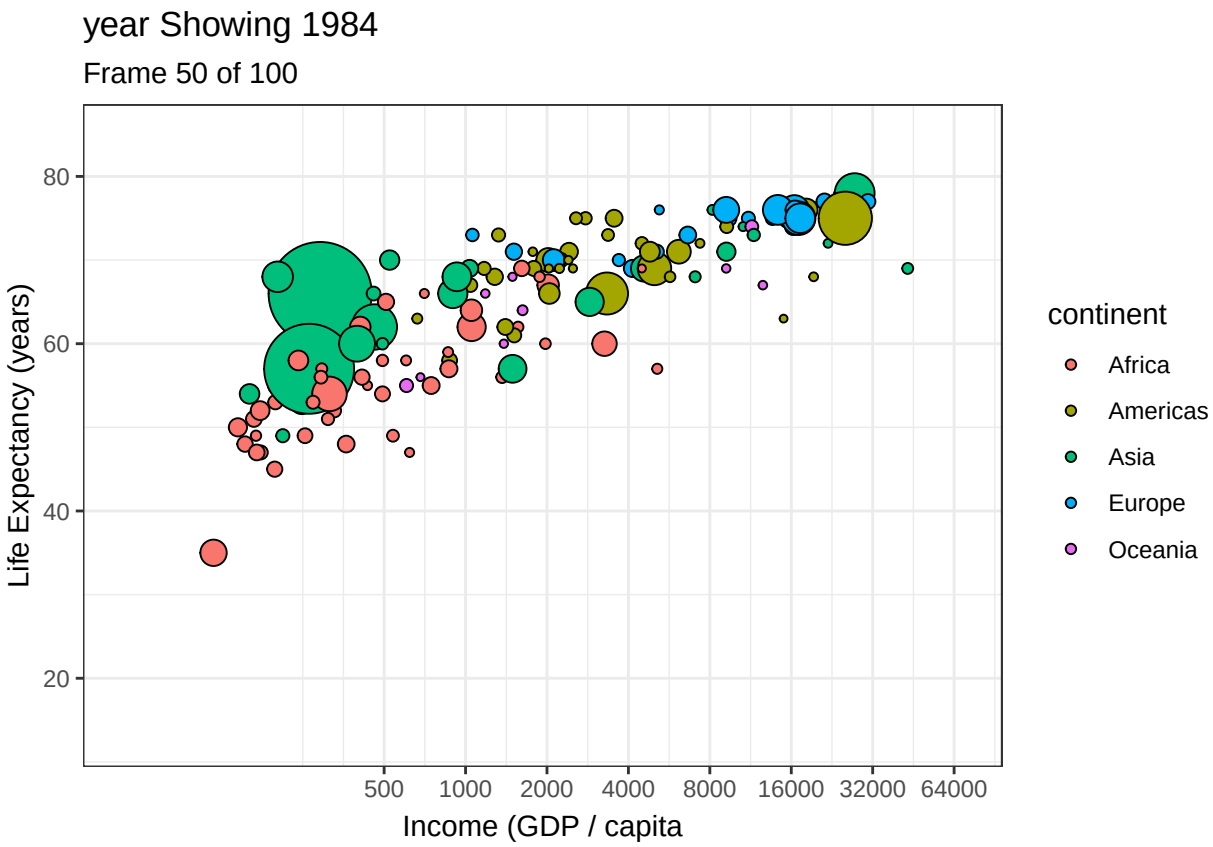

year Showing 1982
Frame 45 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

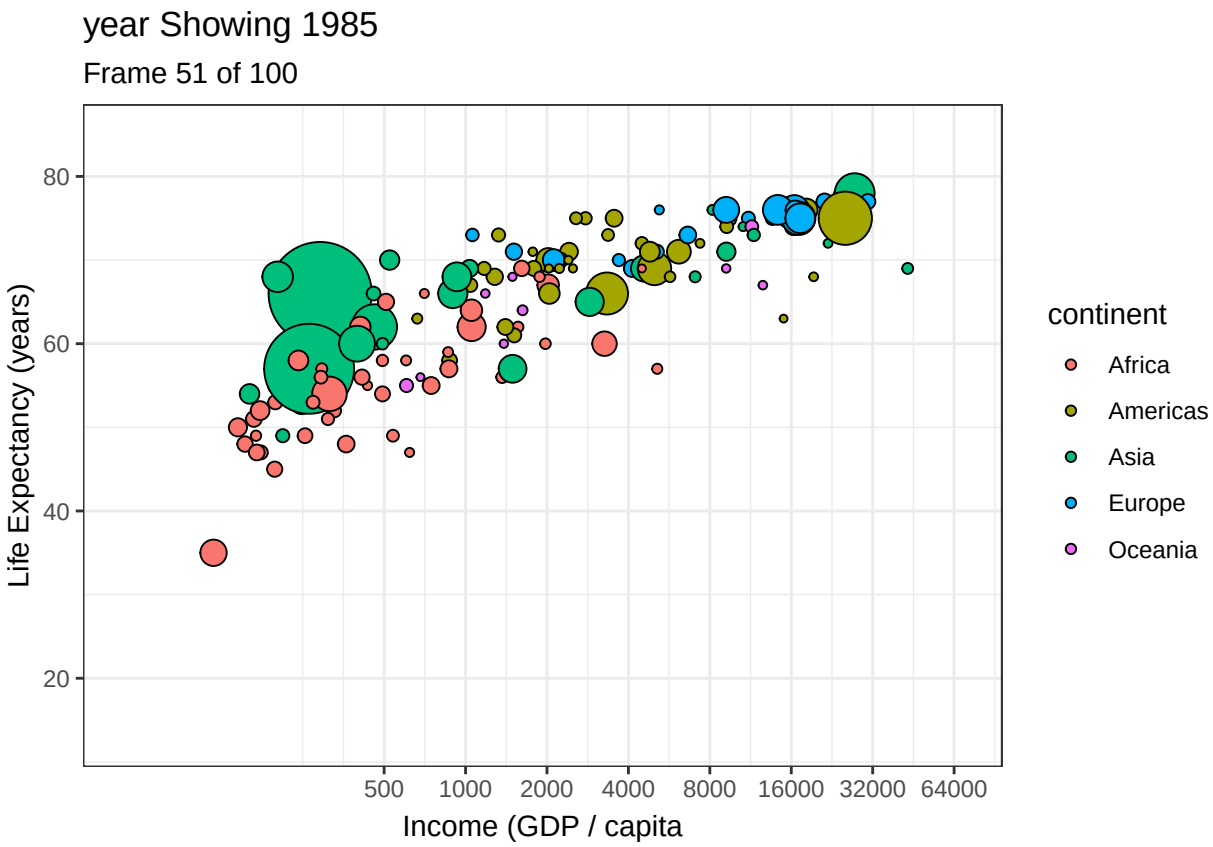

year Showing 1982
Frame 46 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

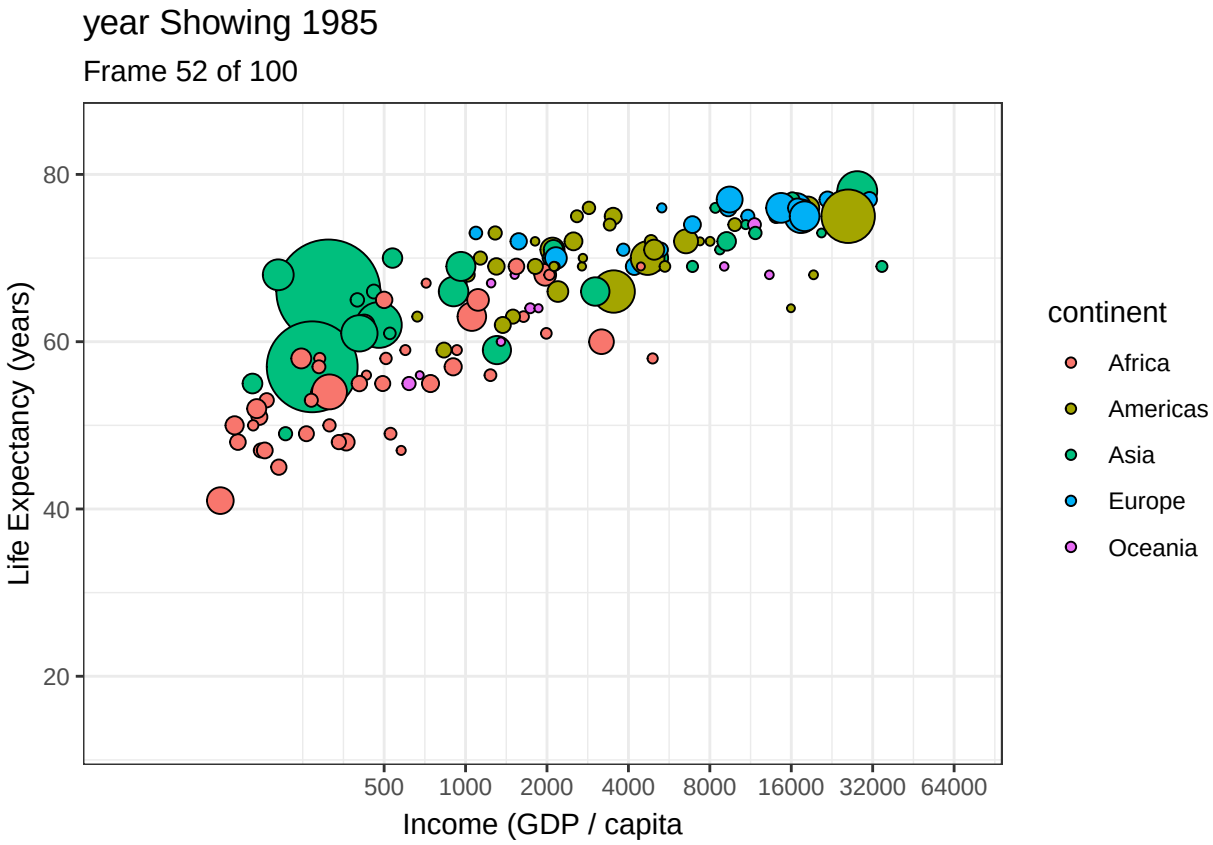

year Showing 1983
Frame 47 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

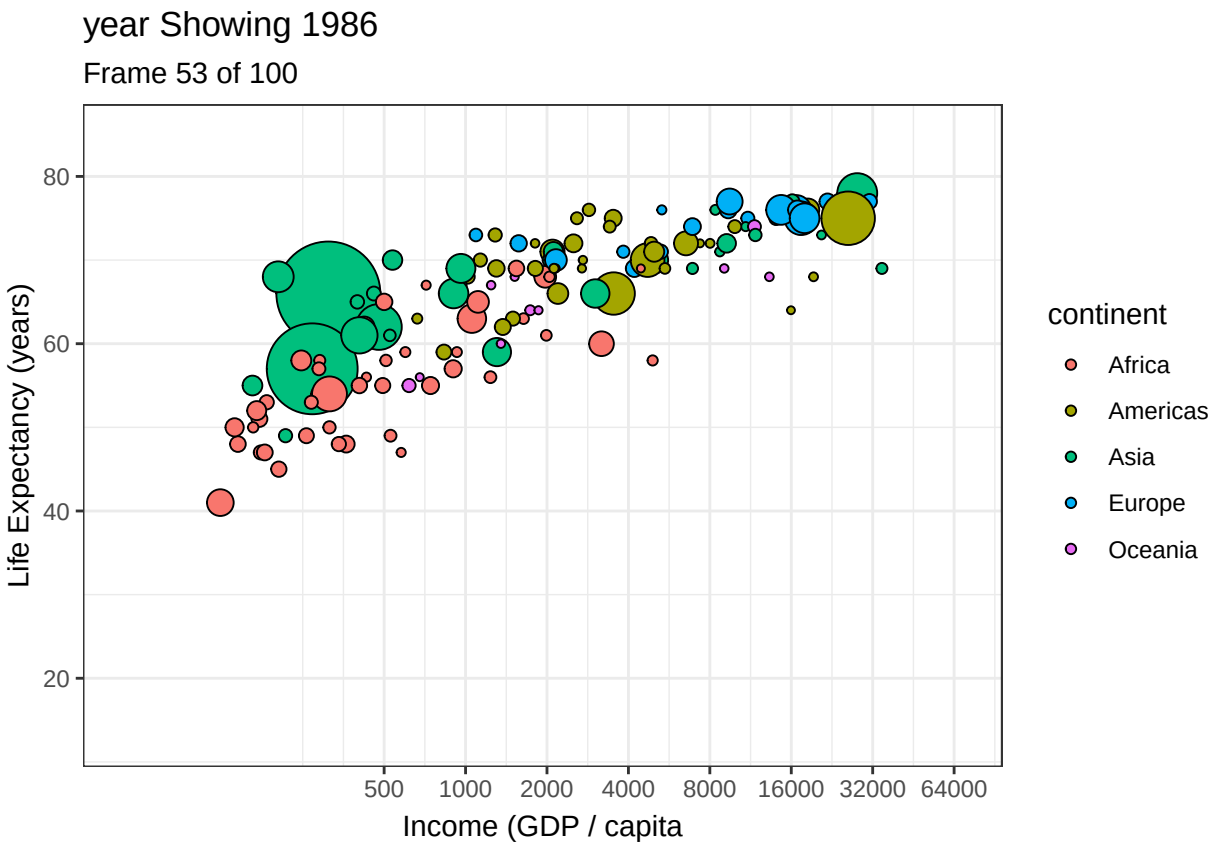

year Showing 1983
Frame 48 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

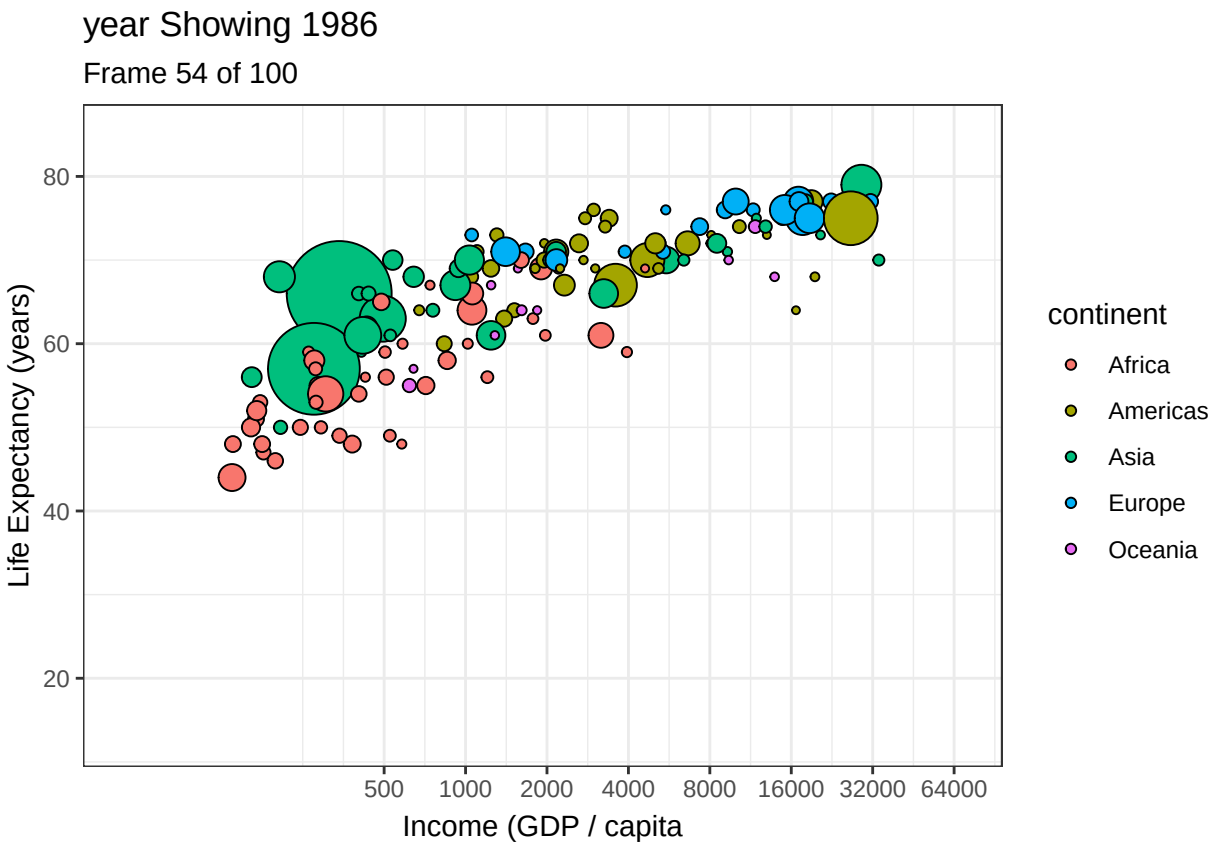

year Showing 1984
Frame 49 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

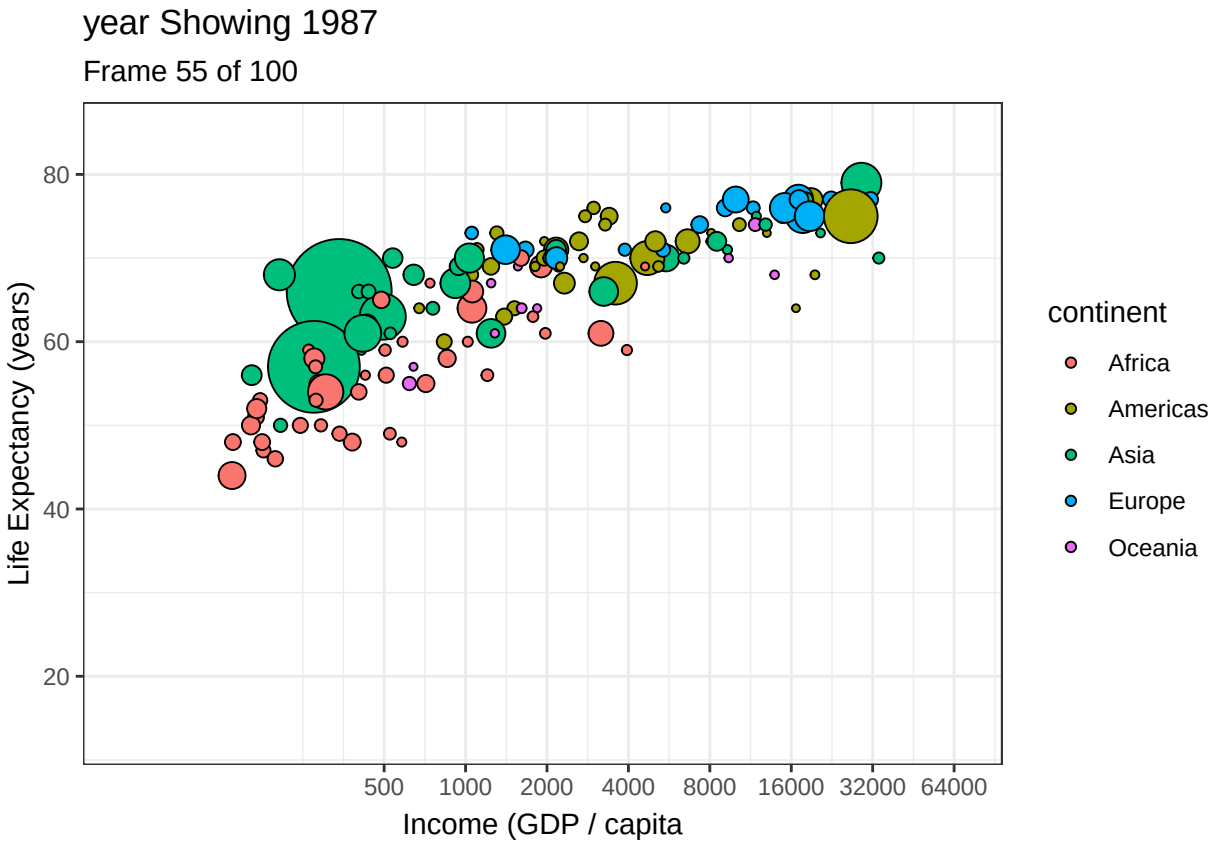

year Showing 1984
Frame 50 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

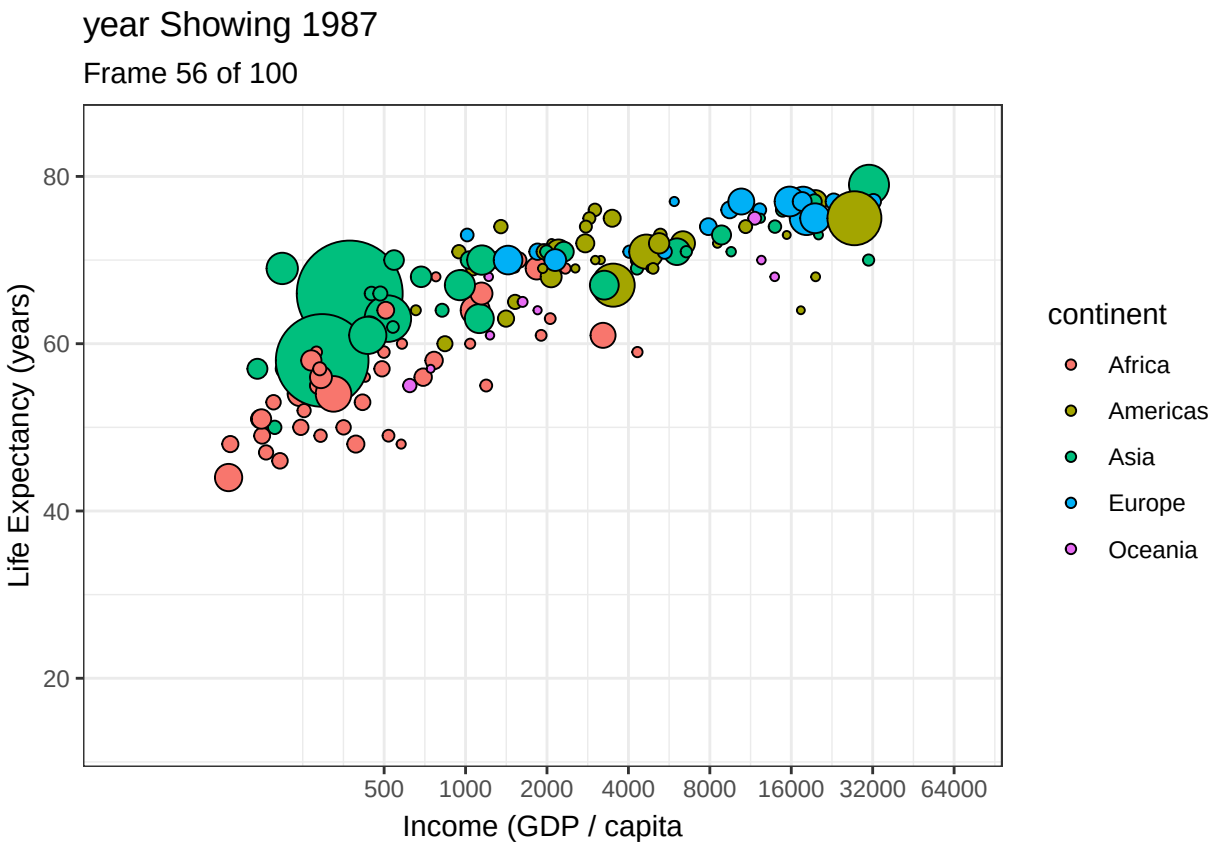

year Showing 1985
Frame 51 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

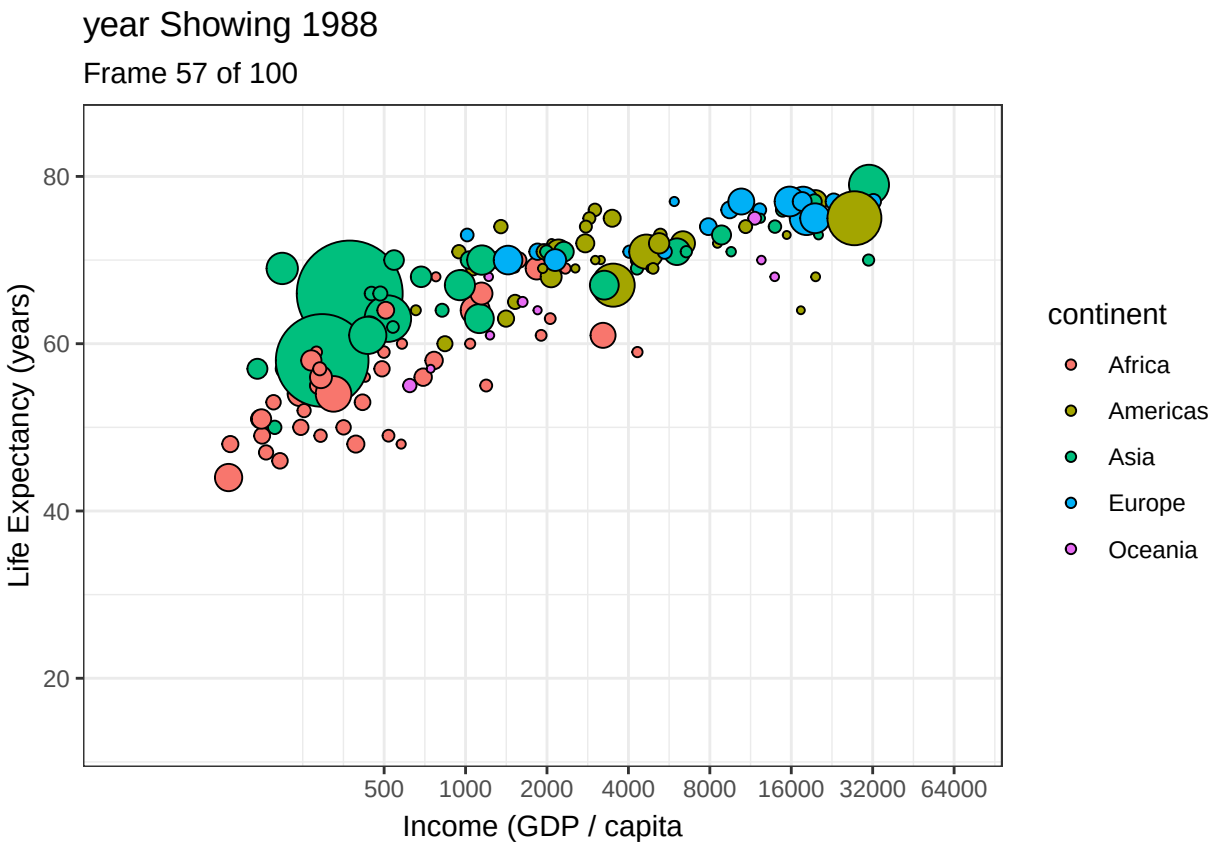

year Showing 1985
Frame 52 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

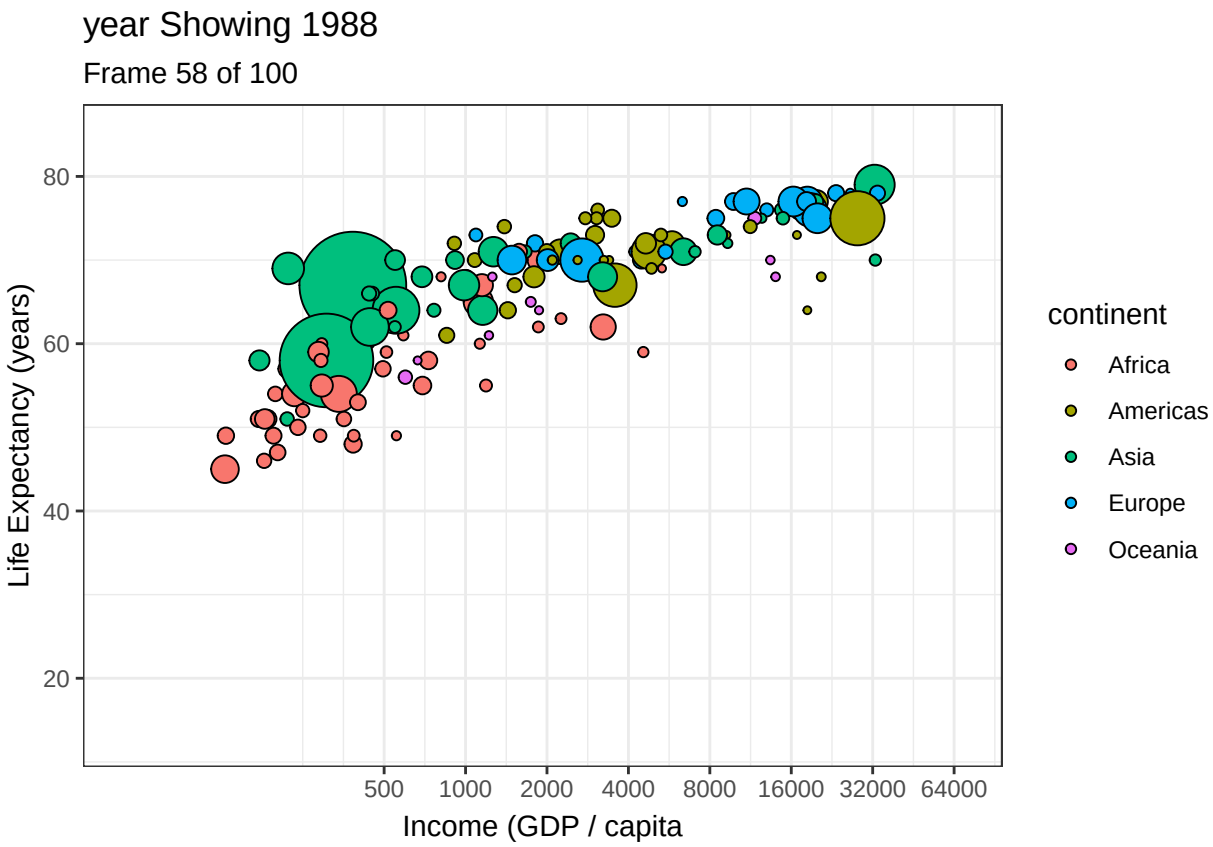

year Showing 1986
Frame 53 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

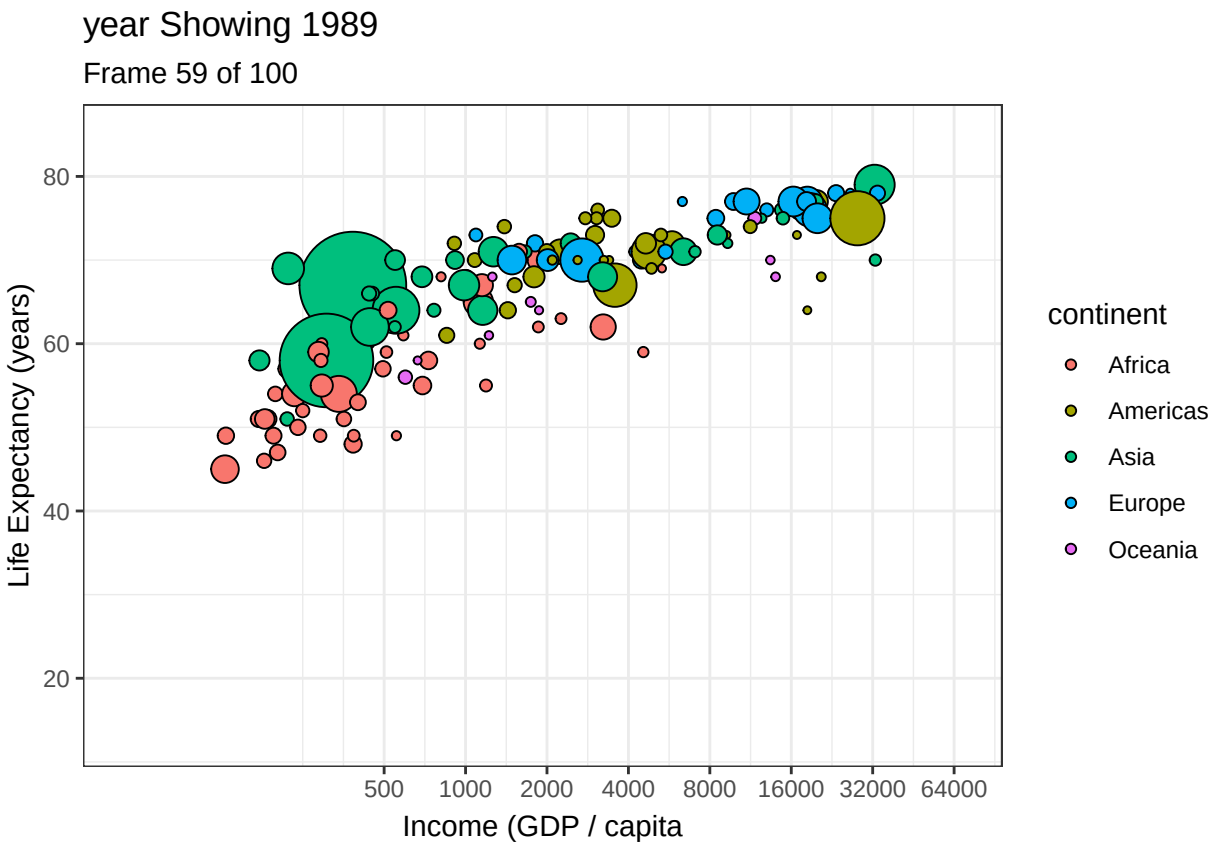

year Showing 1986
Frame 54 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

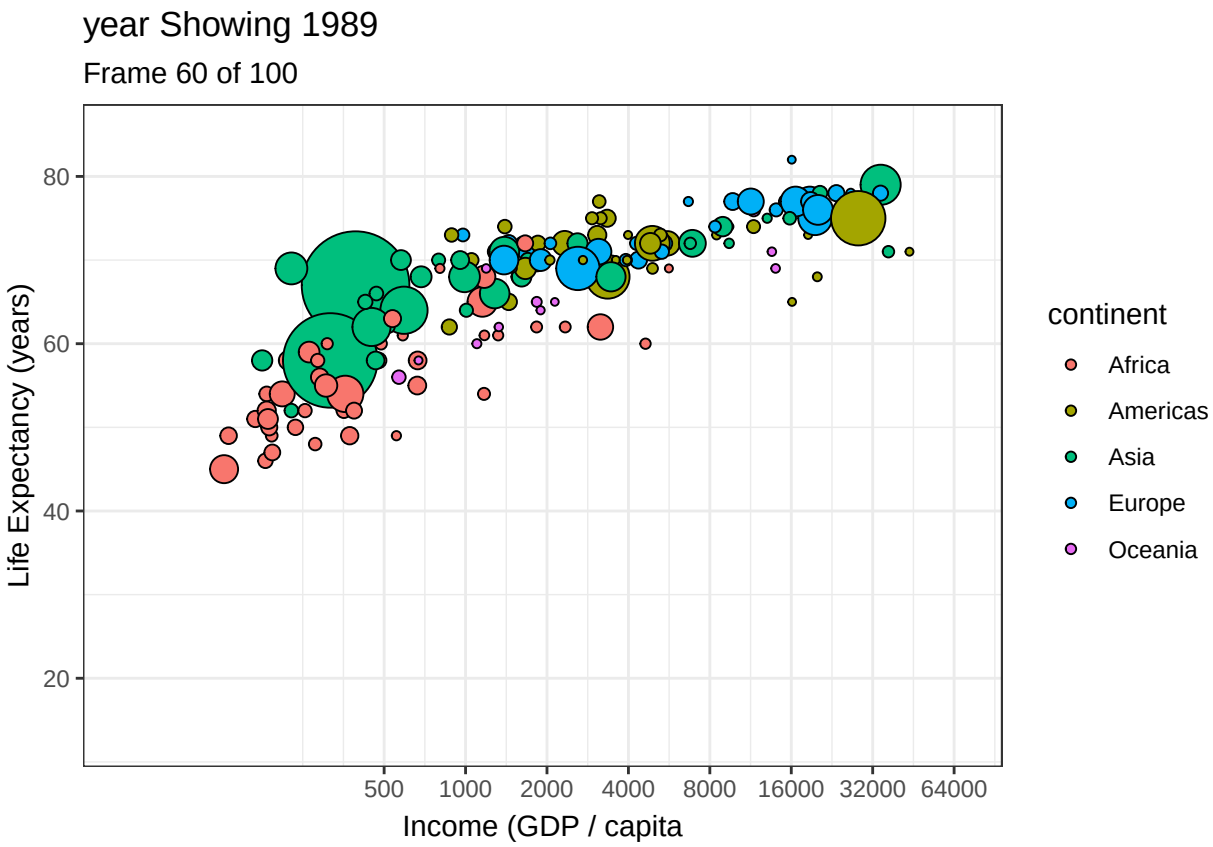

year Showing 1987
Frame 55 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

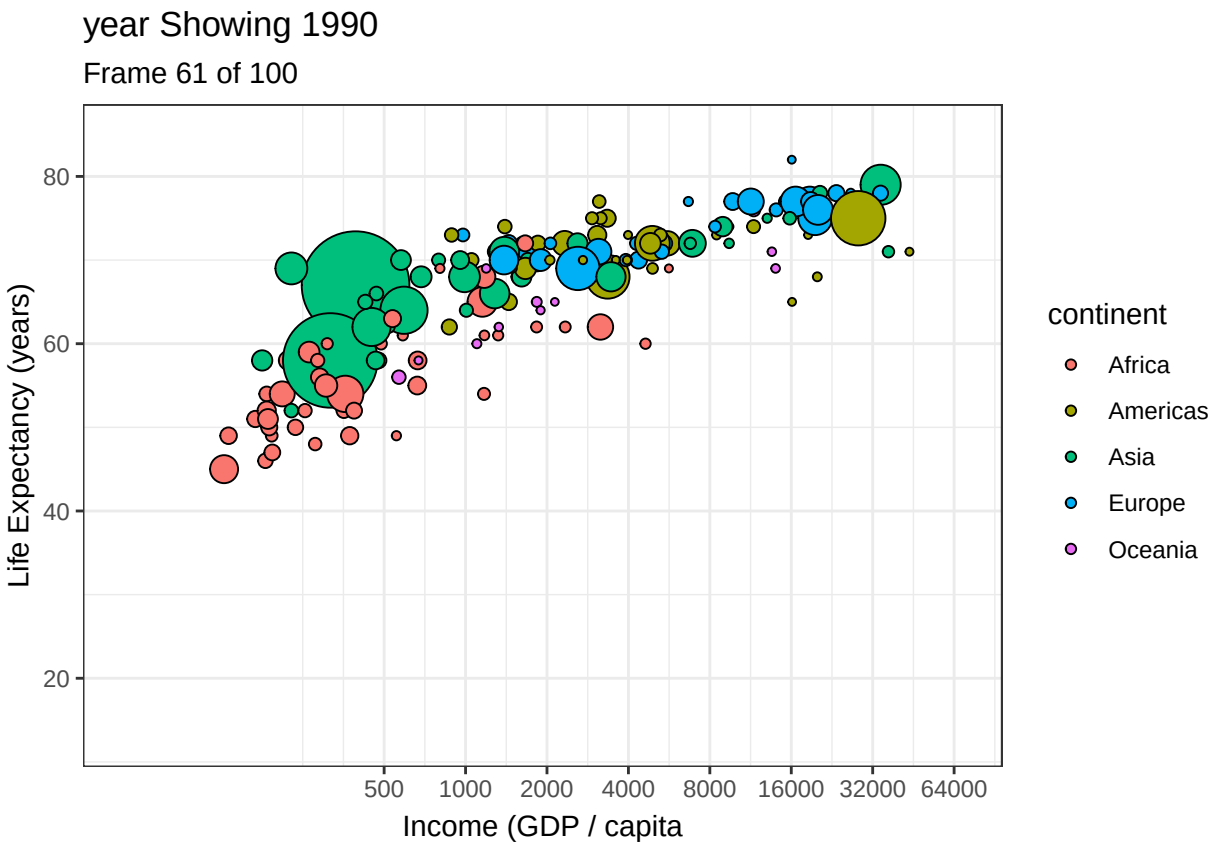

year Showing 1987
Frame 56 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

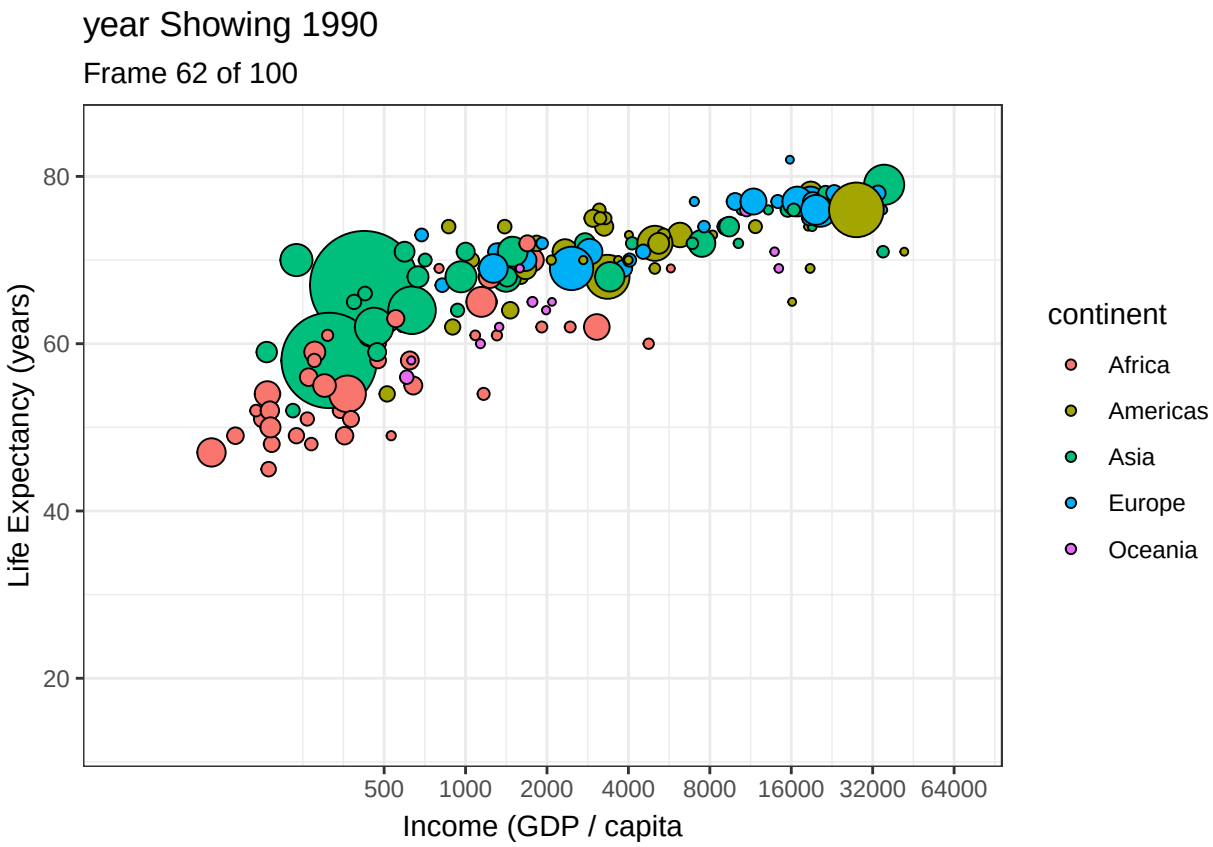

year Showing 1988
Frame 57 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

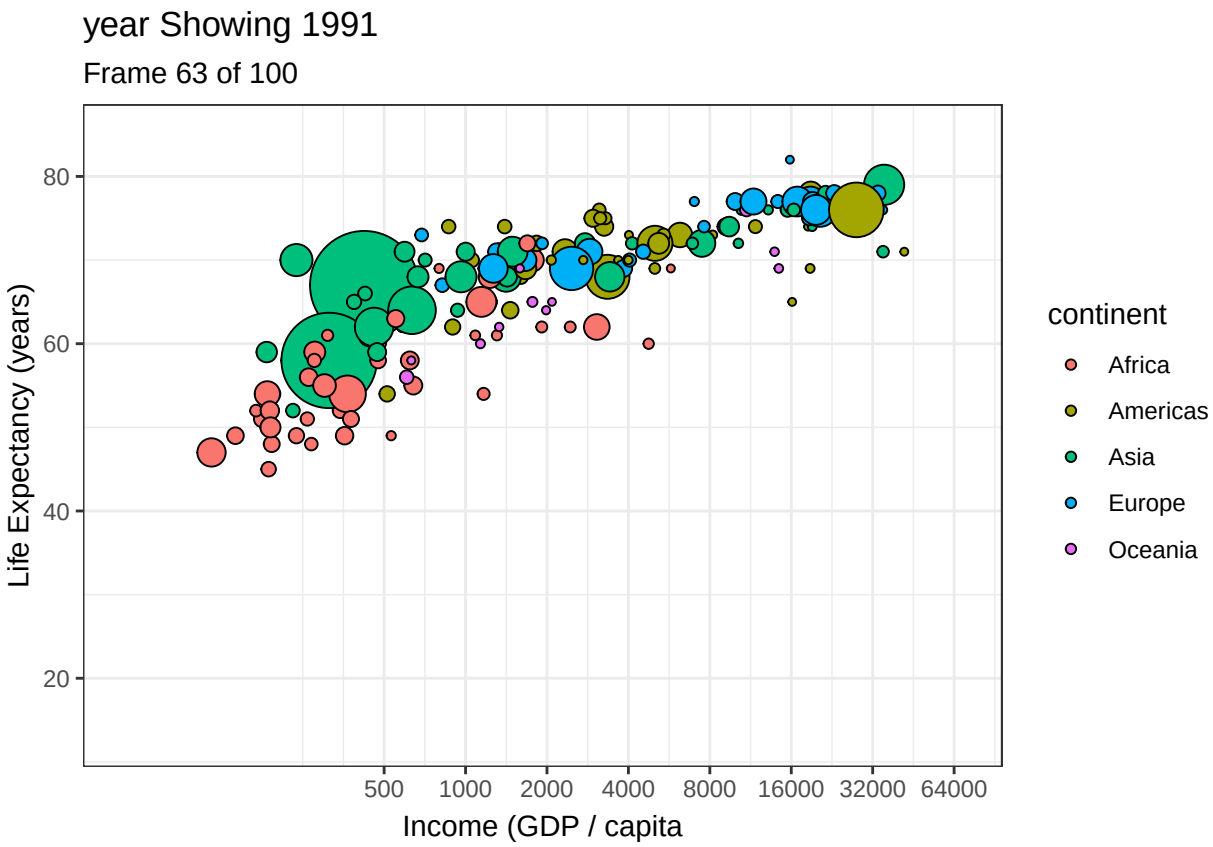

year Showing 1988
Frame 58 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

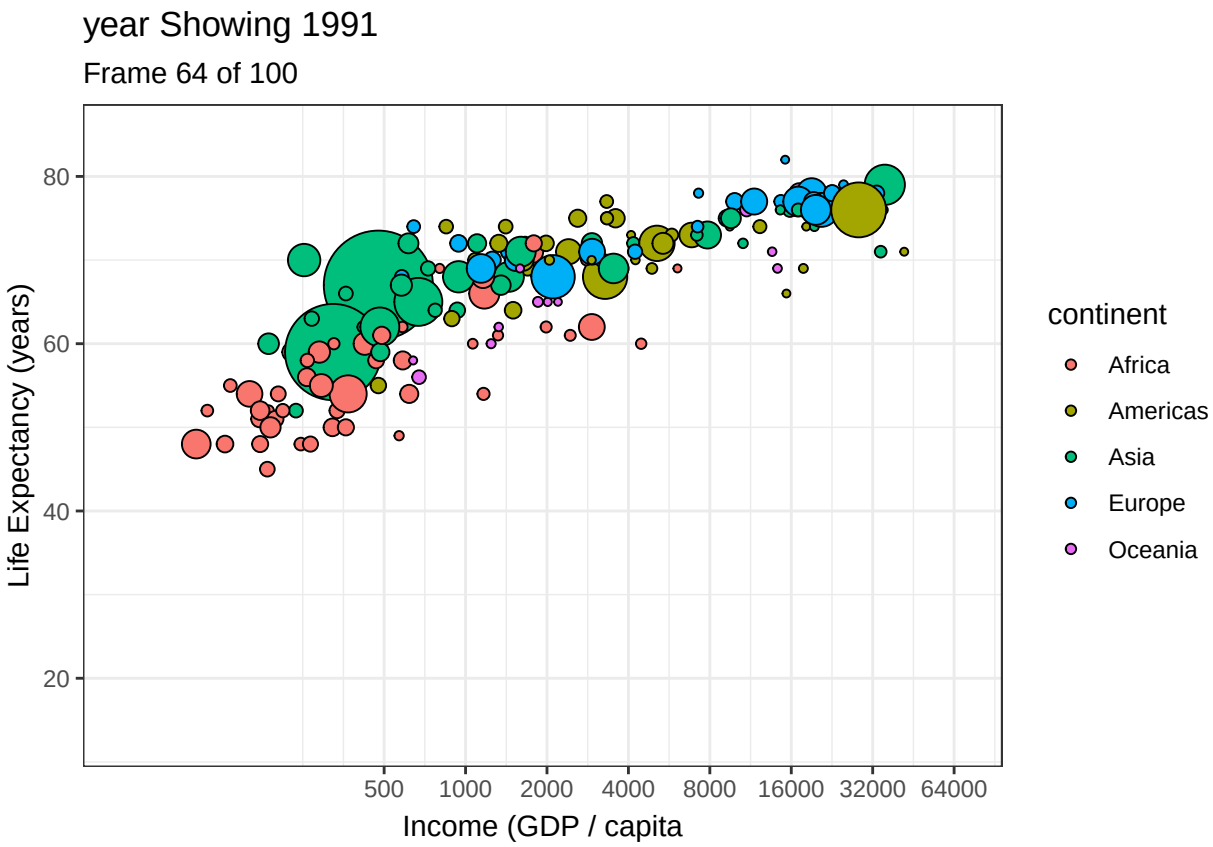

year Showing 1989
Frame 59 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

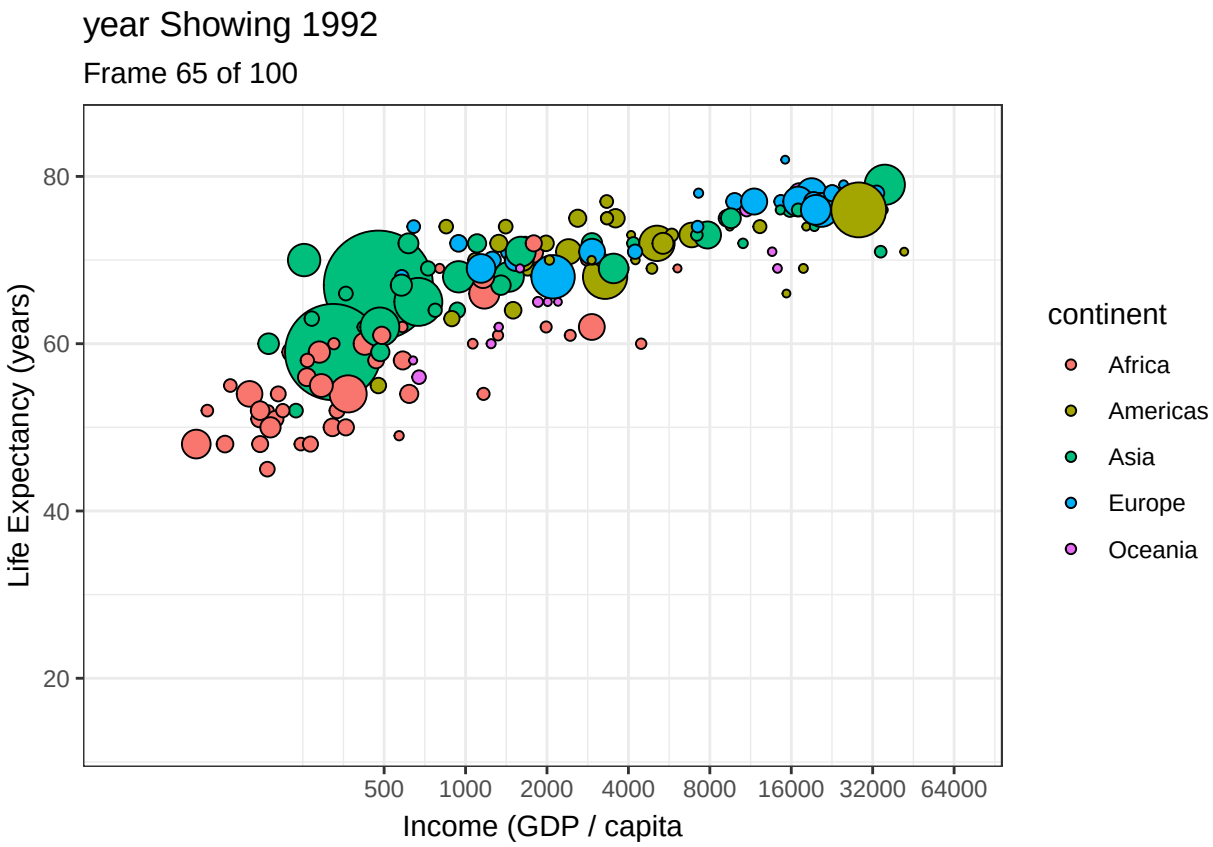

year Showing 1989
Frame 60 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

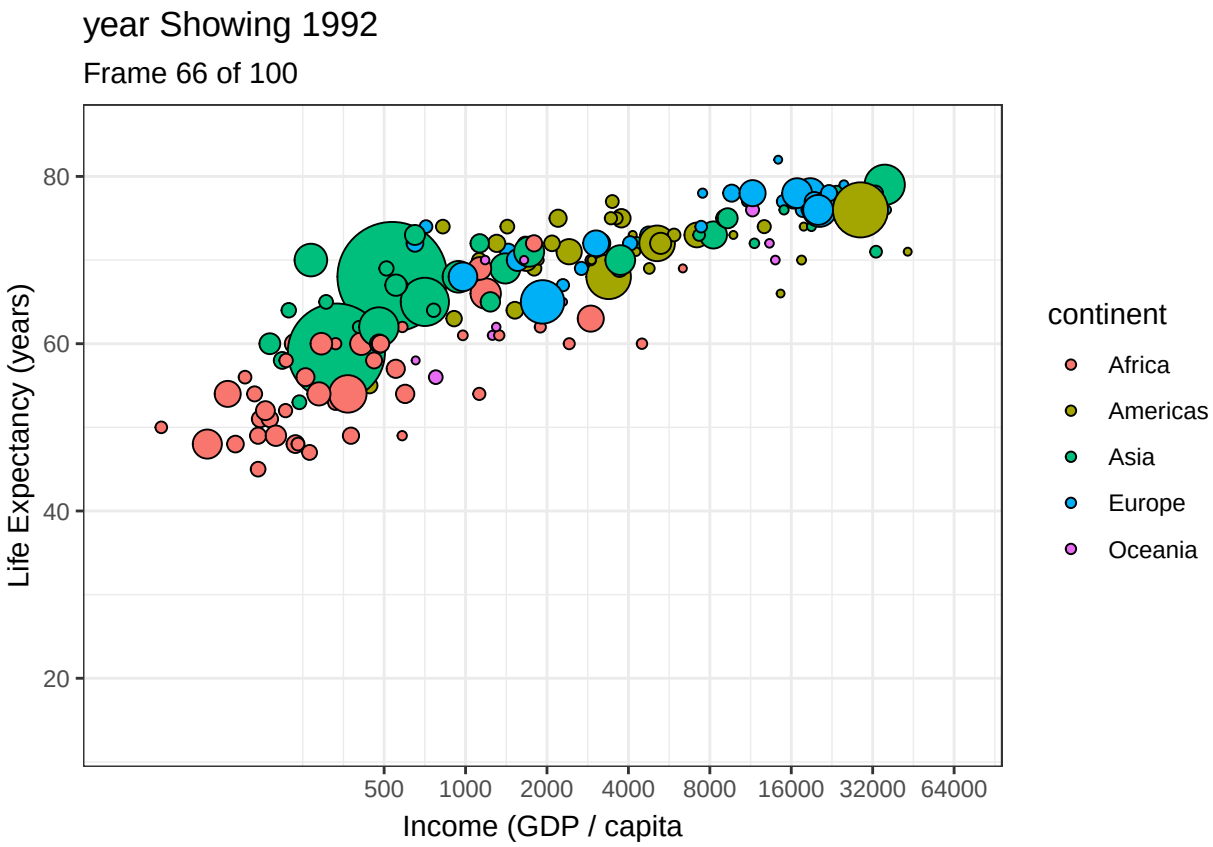

year Showing 1990
Frame 61 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

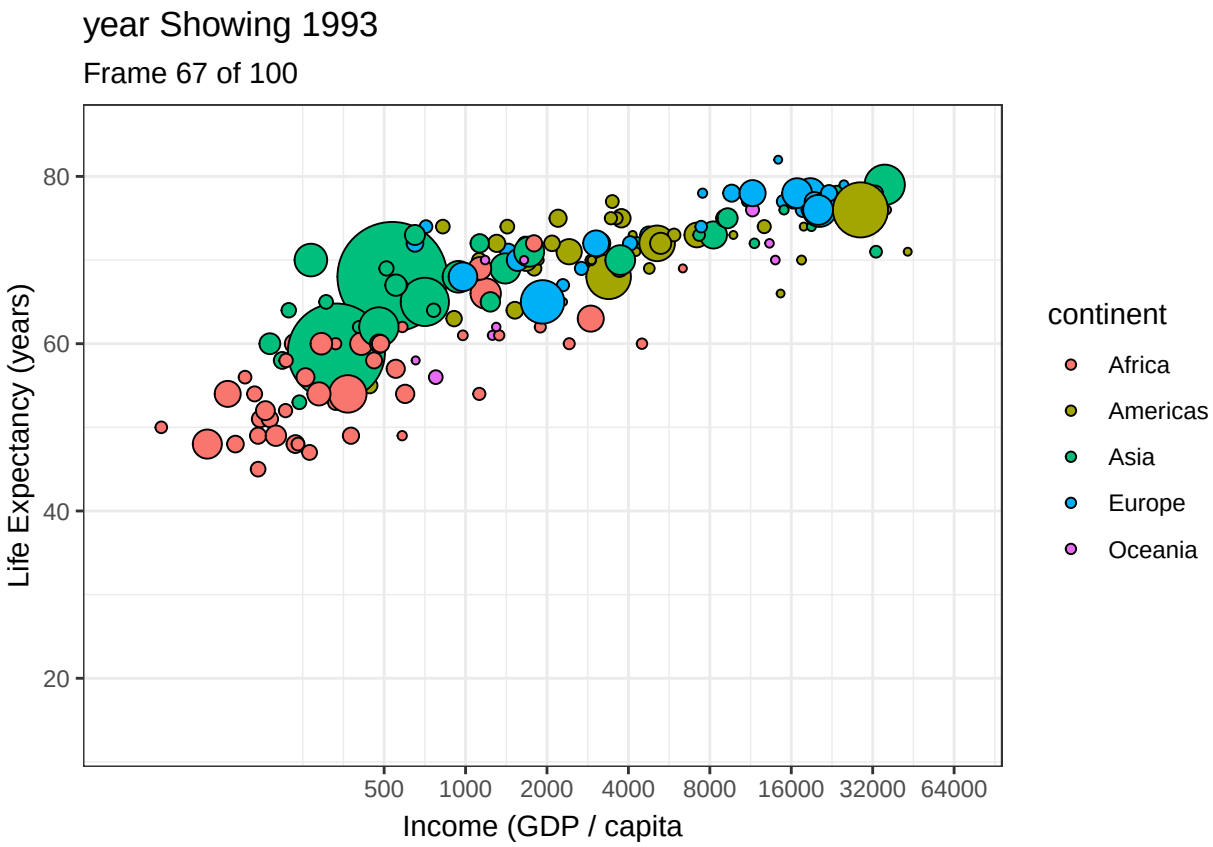

year Showing 1990
Frame 62 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

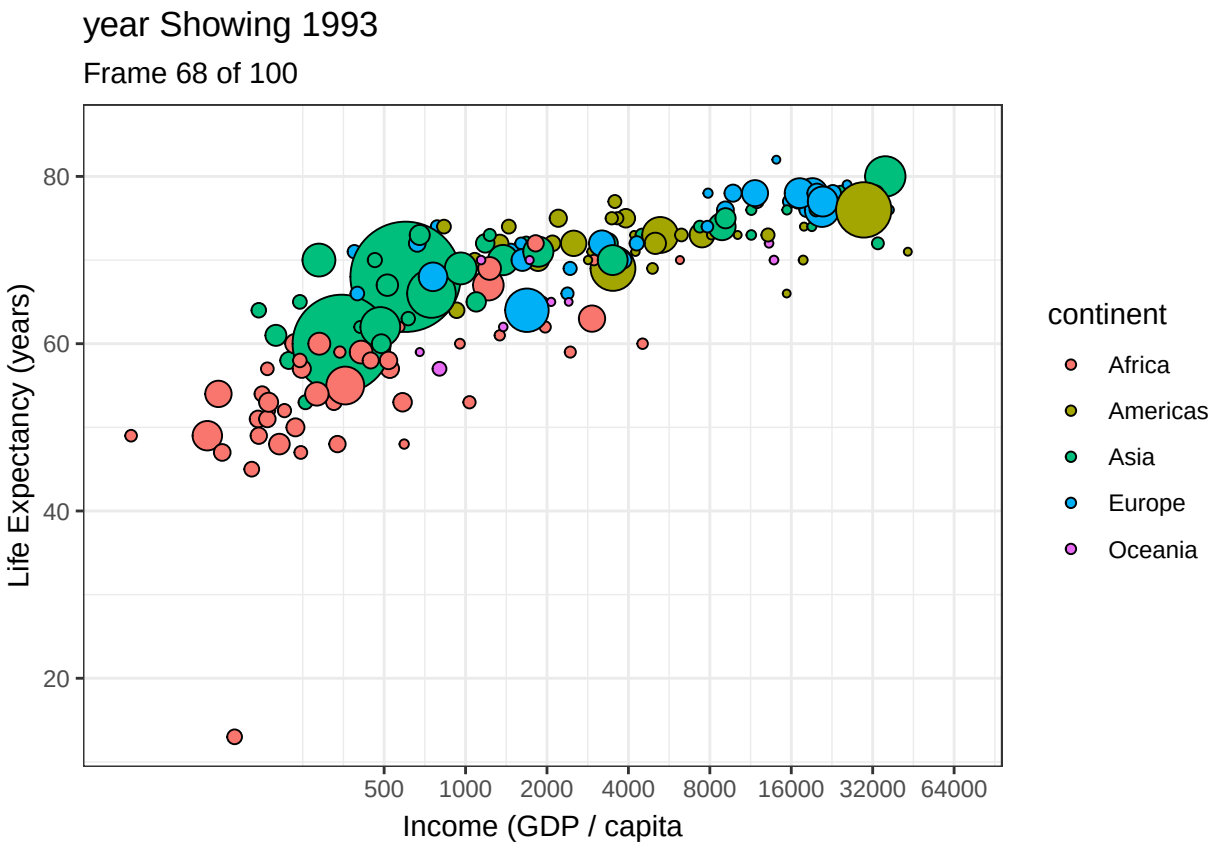

year Showing 1991
Frame 63 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

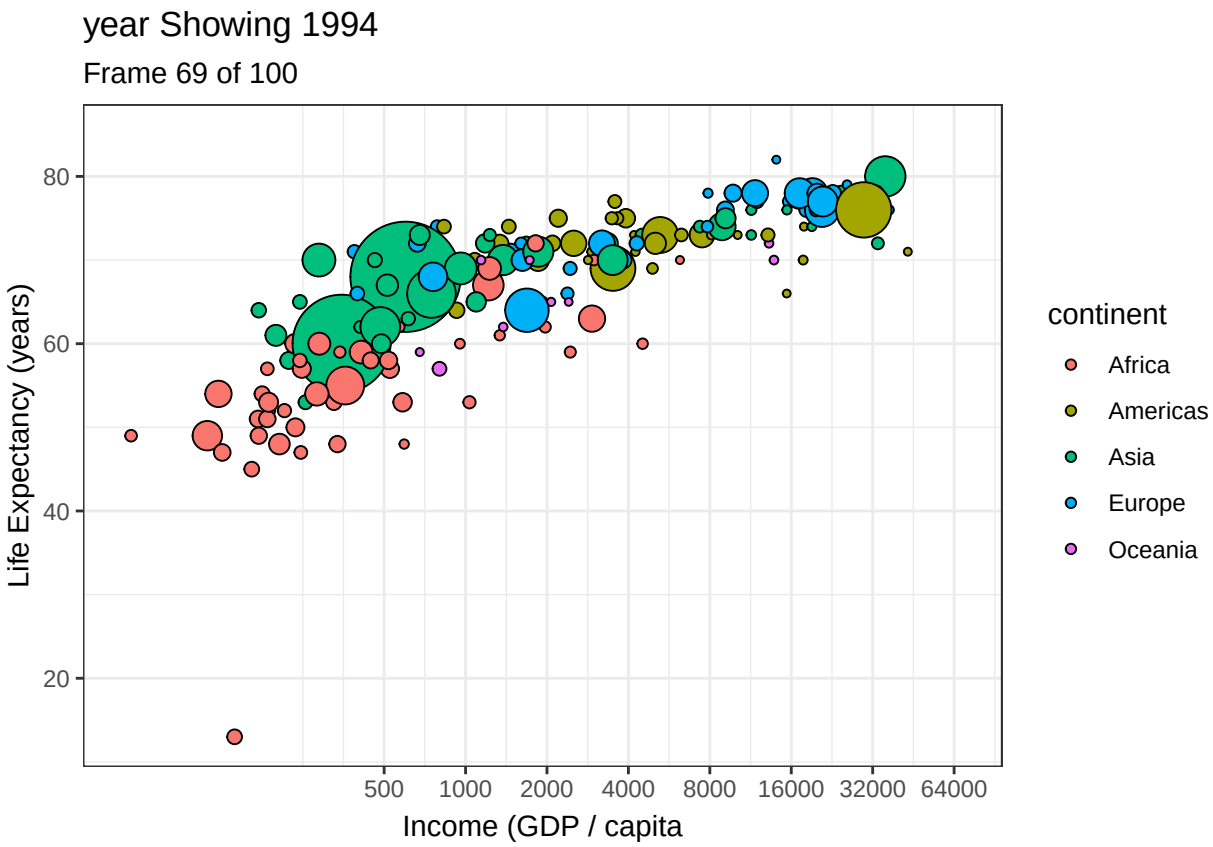

year Showing 1991
Frame 64 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

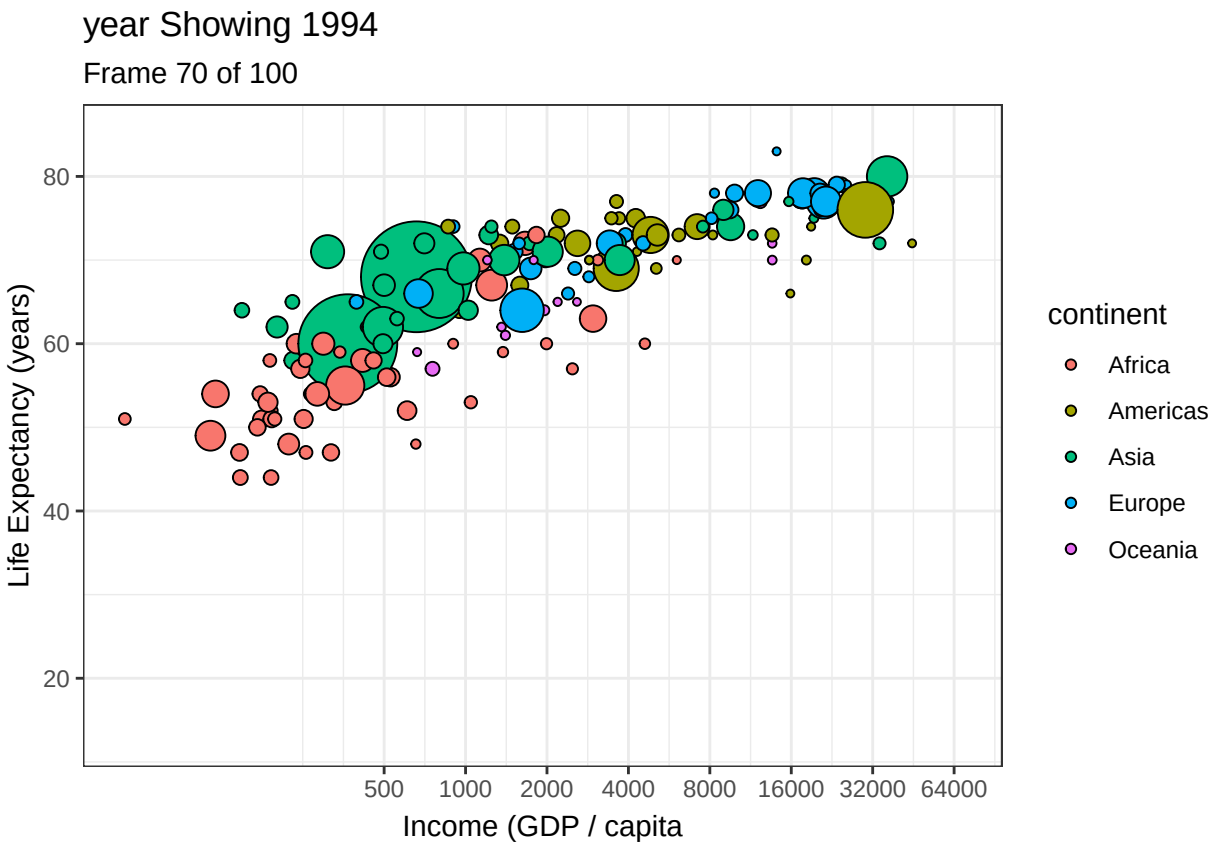

year Showing 1992
Frame 65 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

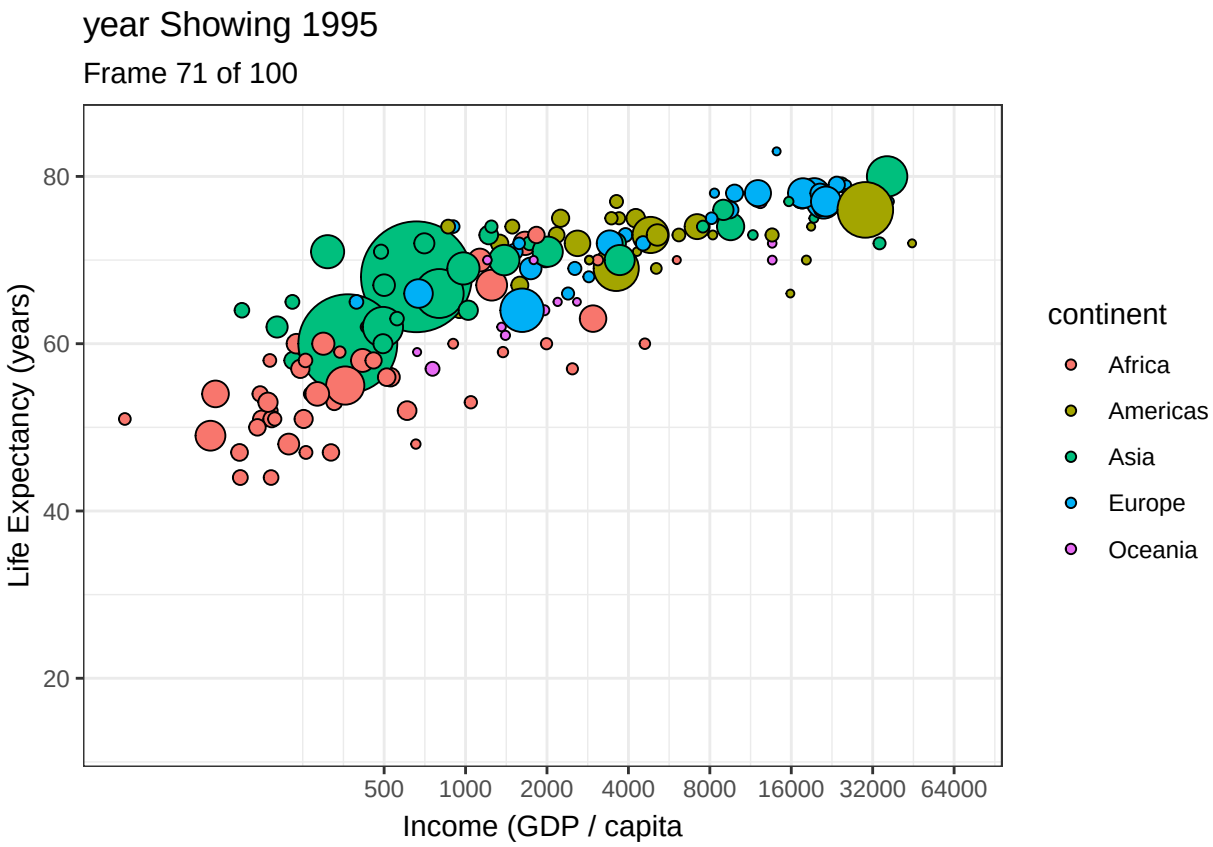

year Showing 1992
Frame 66 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

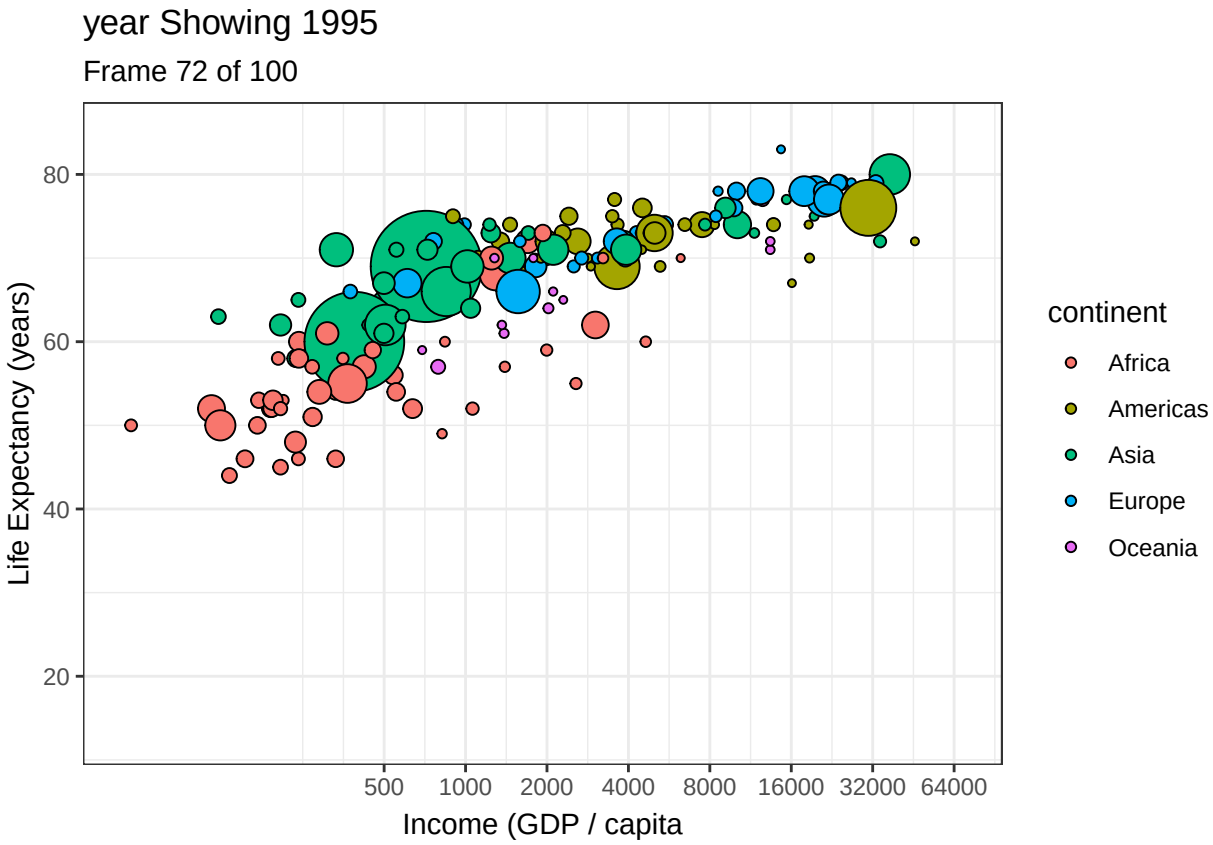

year Showing 1993
Frame 67 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

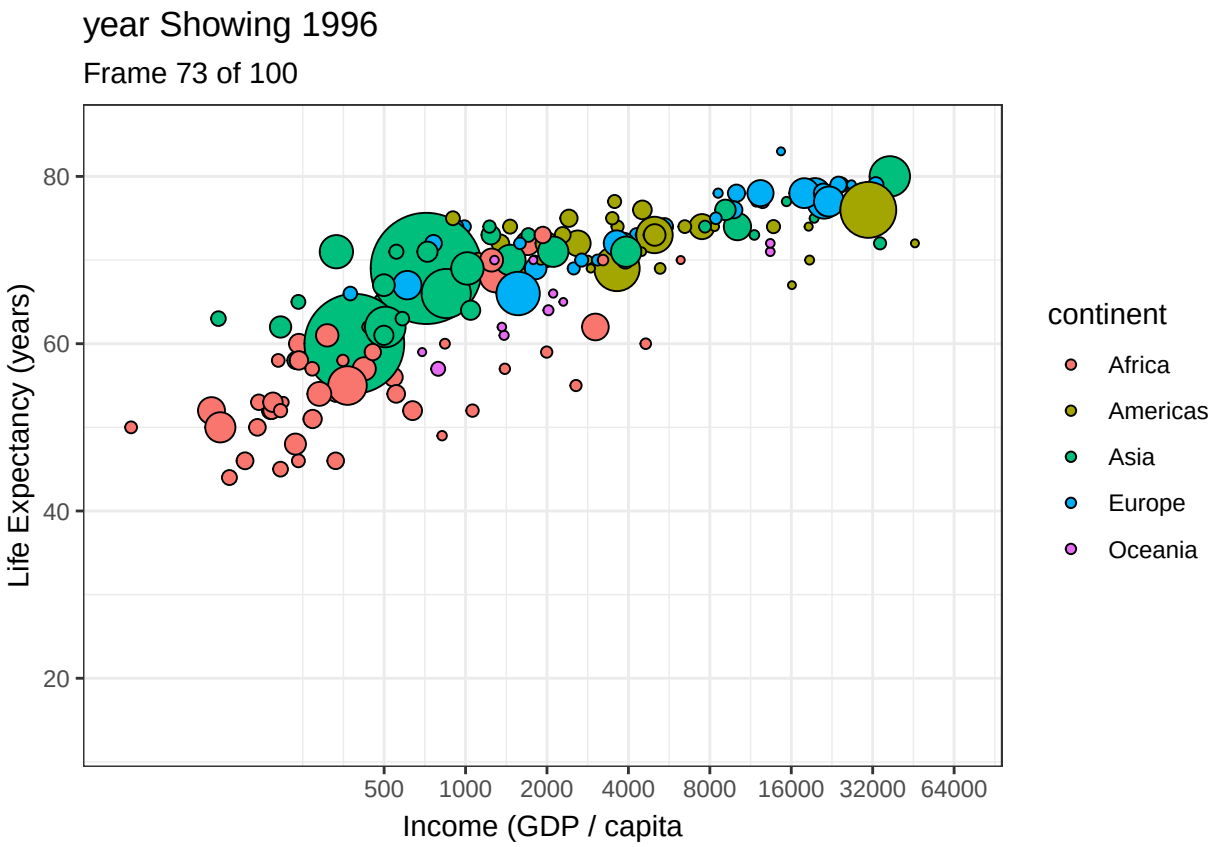

year Showing 1993
Frame 68 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

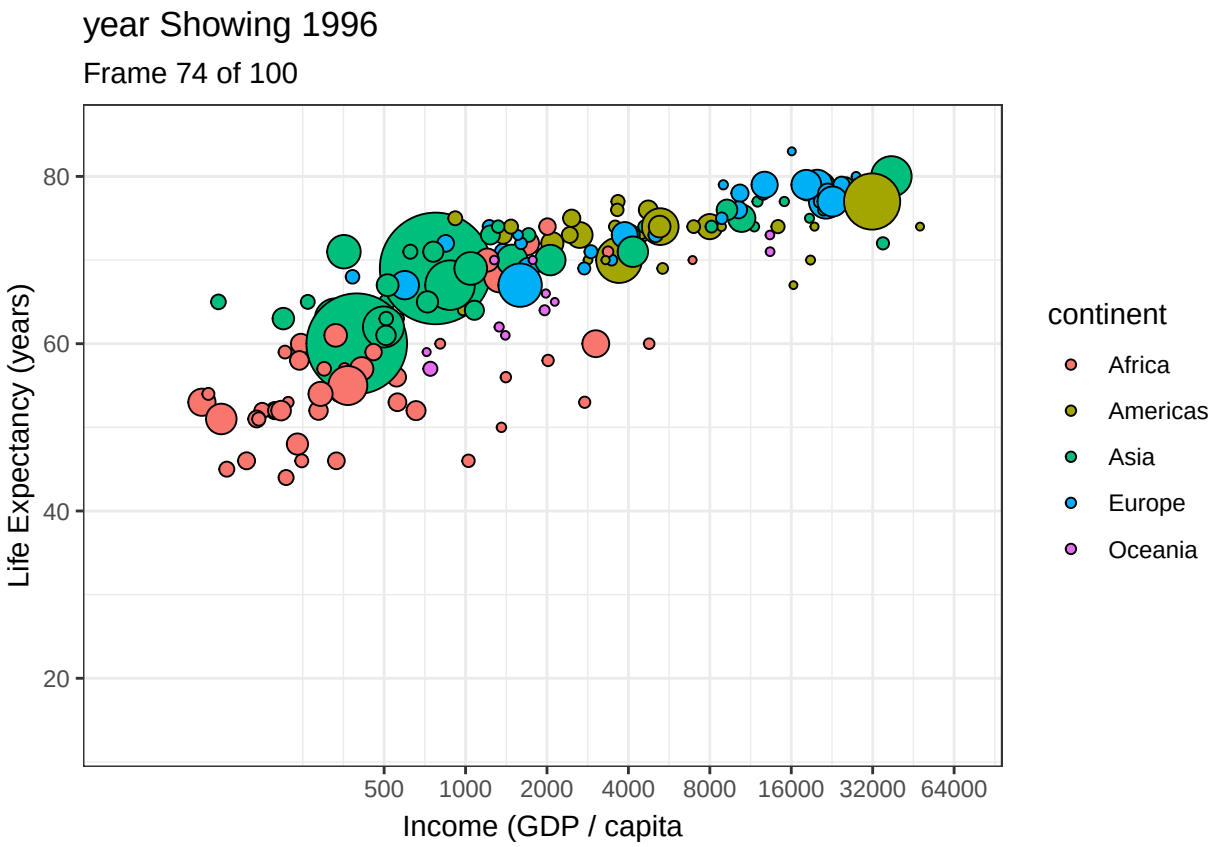

year Showing 1994
Frame 69 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

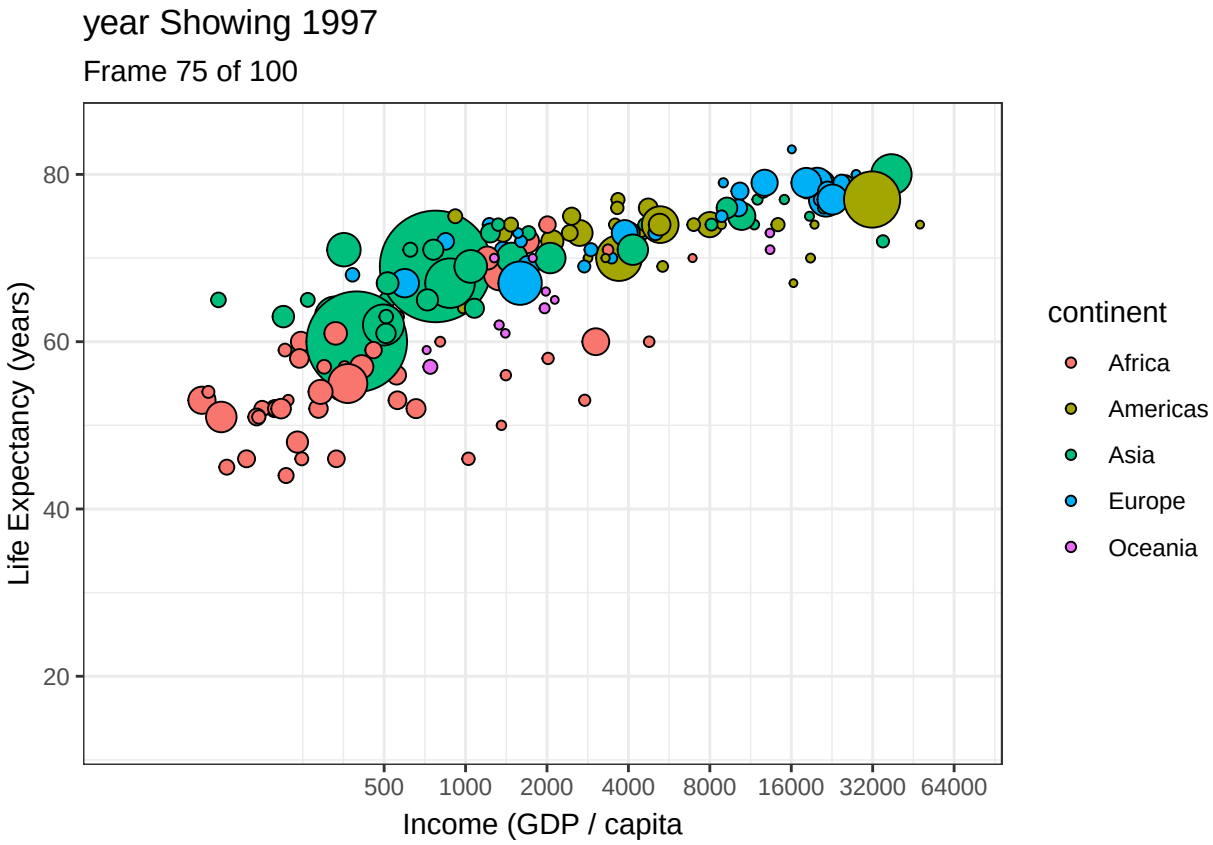

year Showing 1994
Frame 70 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

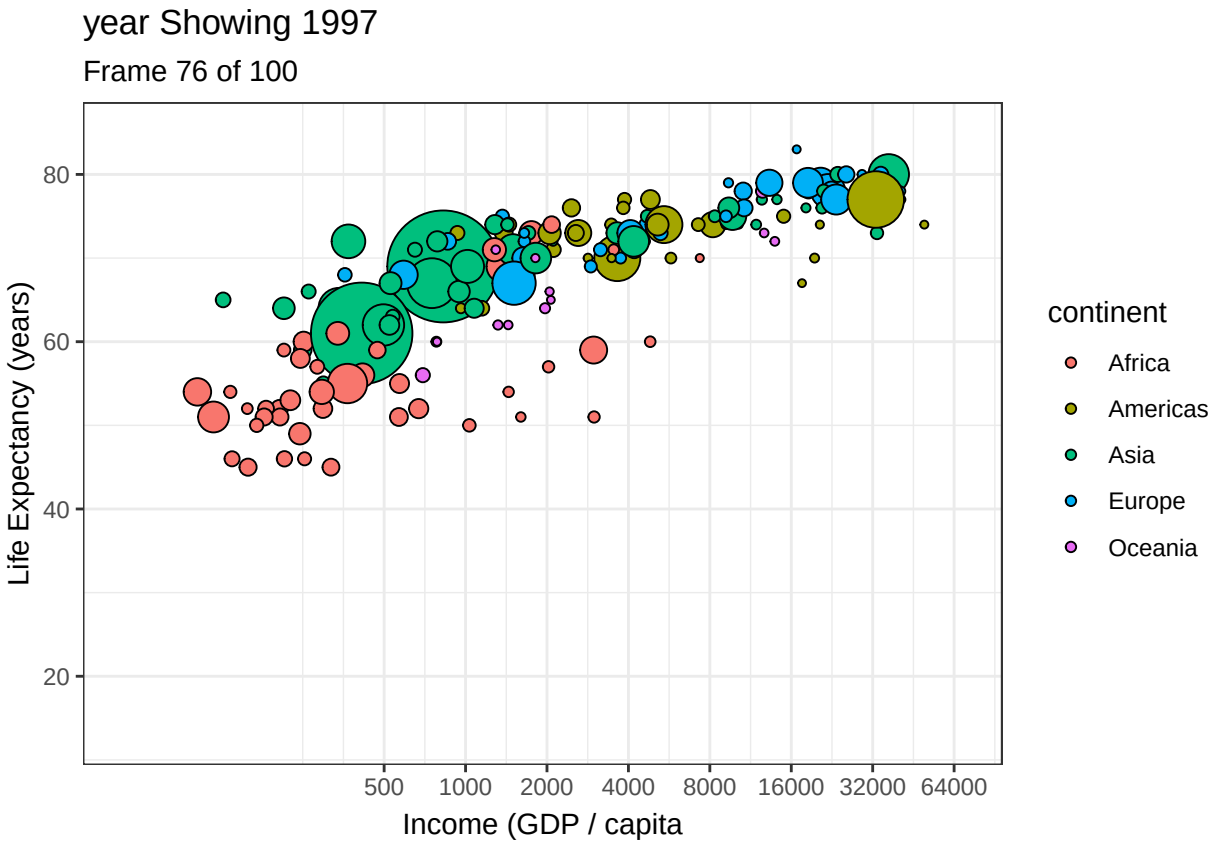

year Showing 1995
Frame 71 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

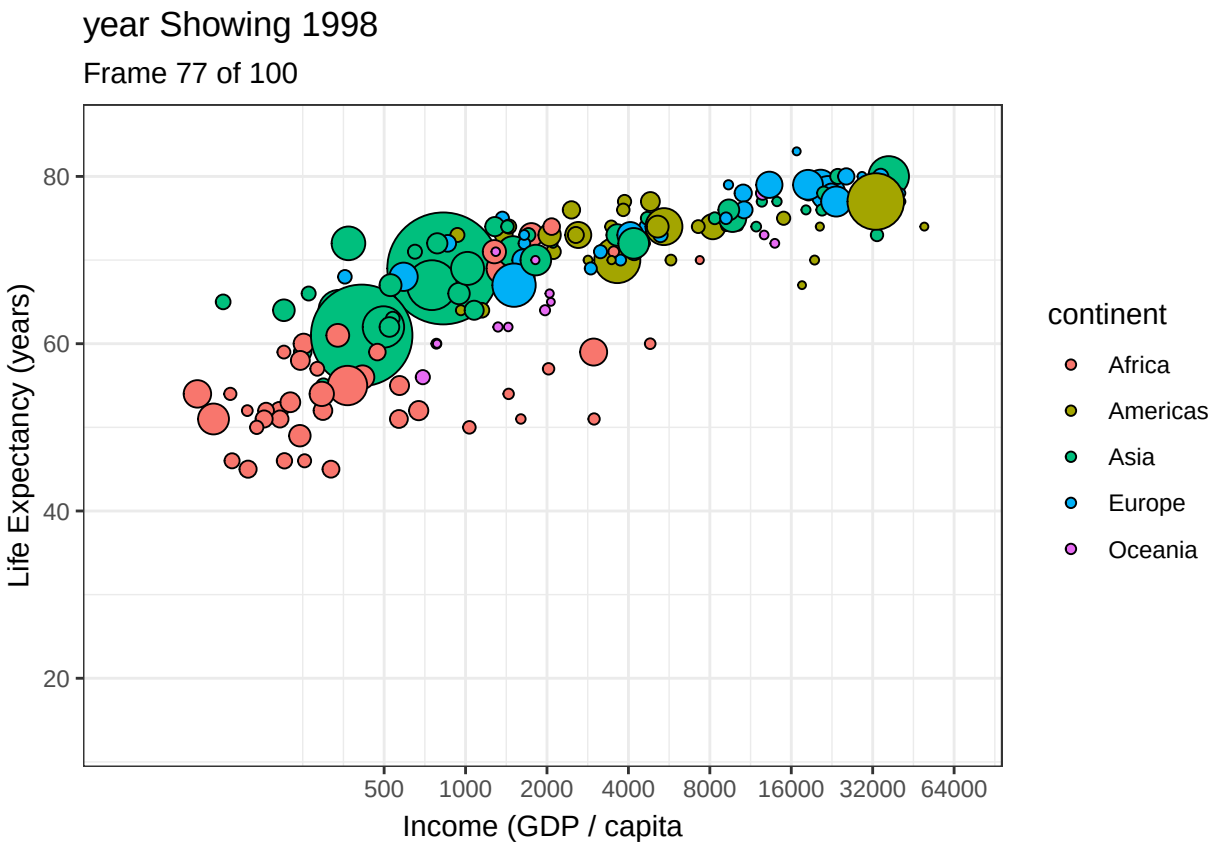

year Showing 1995
Frame 72 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

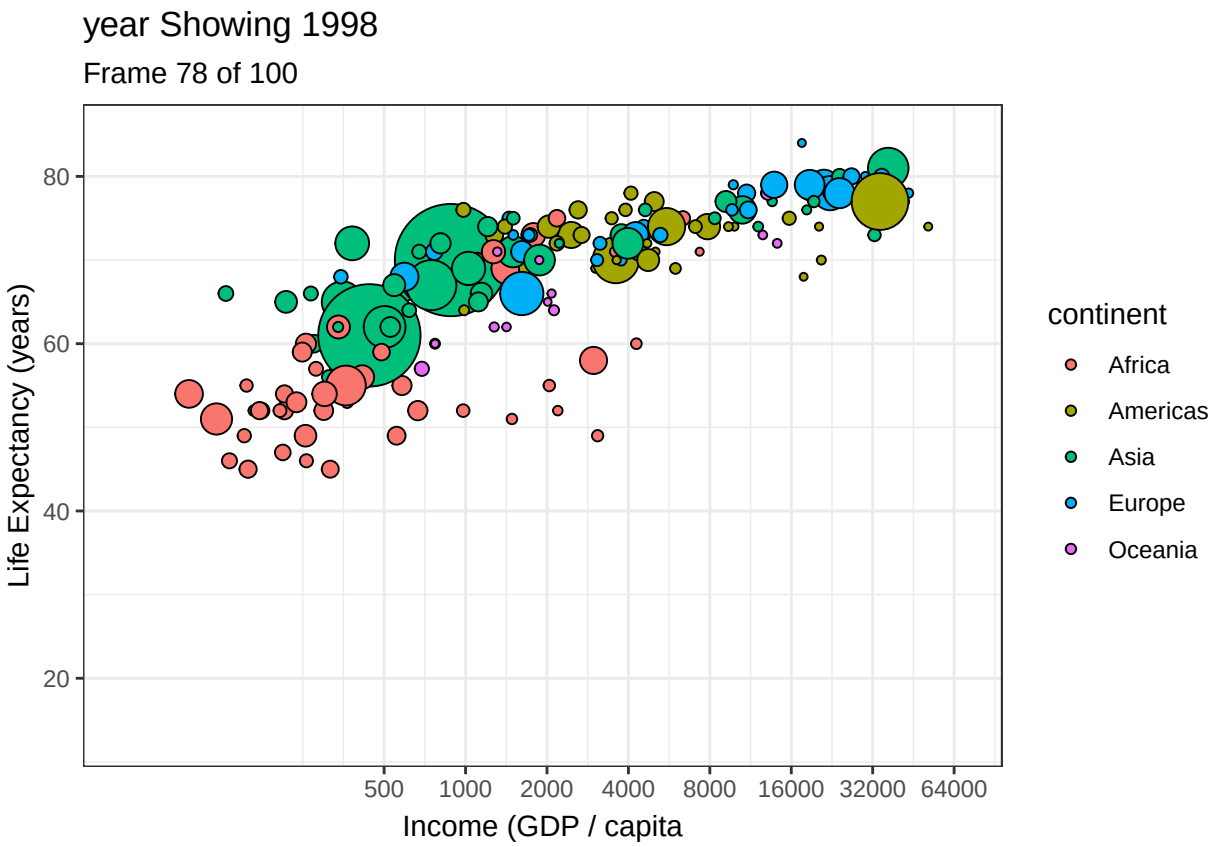

year Showing 1996
Frame 73 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

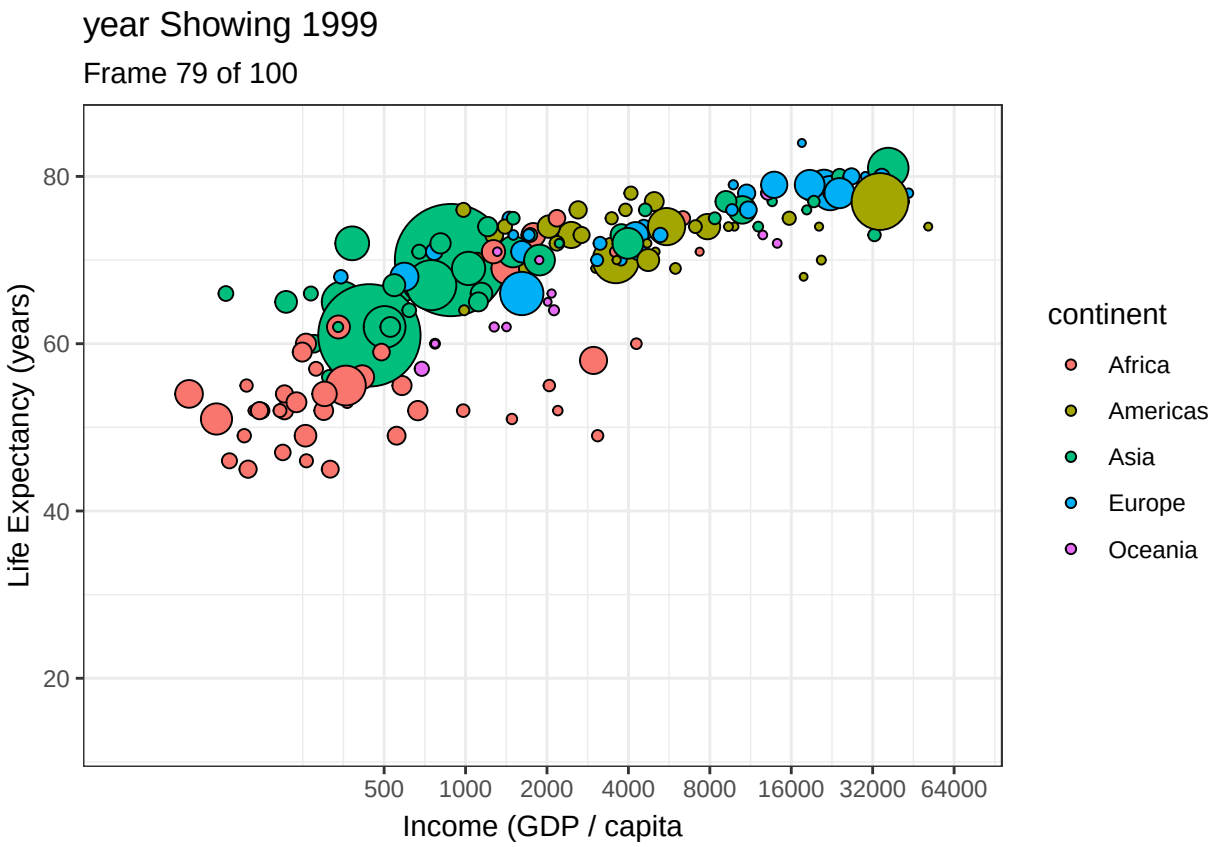

year Showing 1996
Frame 74 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

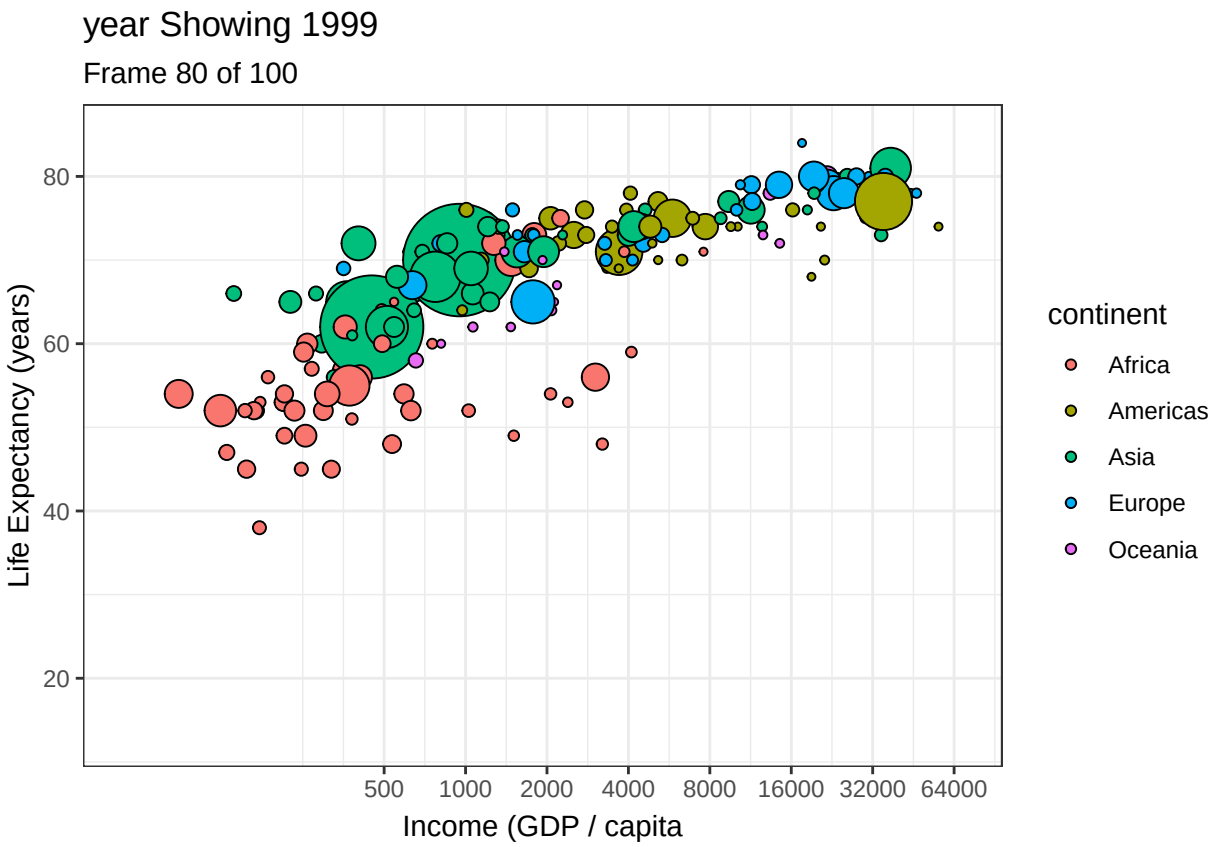

year Showing 1997
Frame 75 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

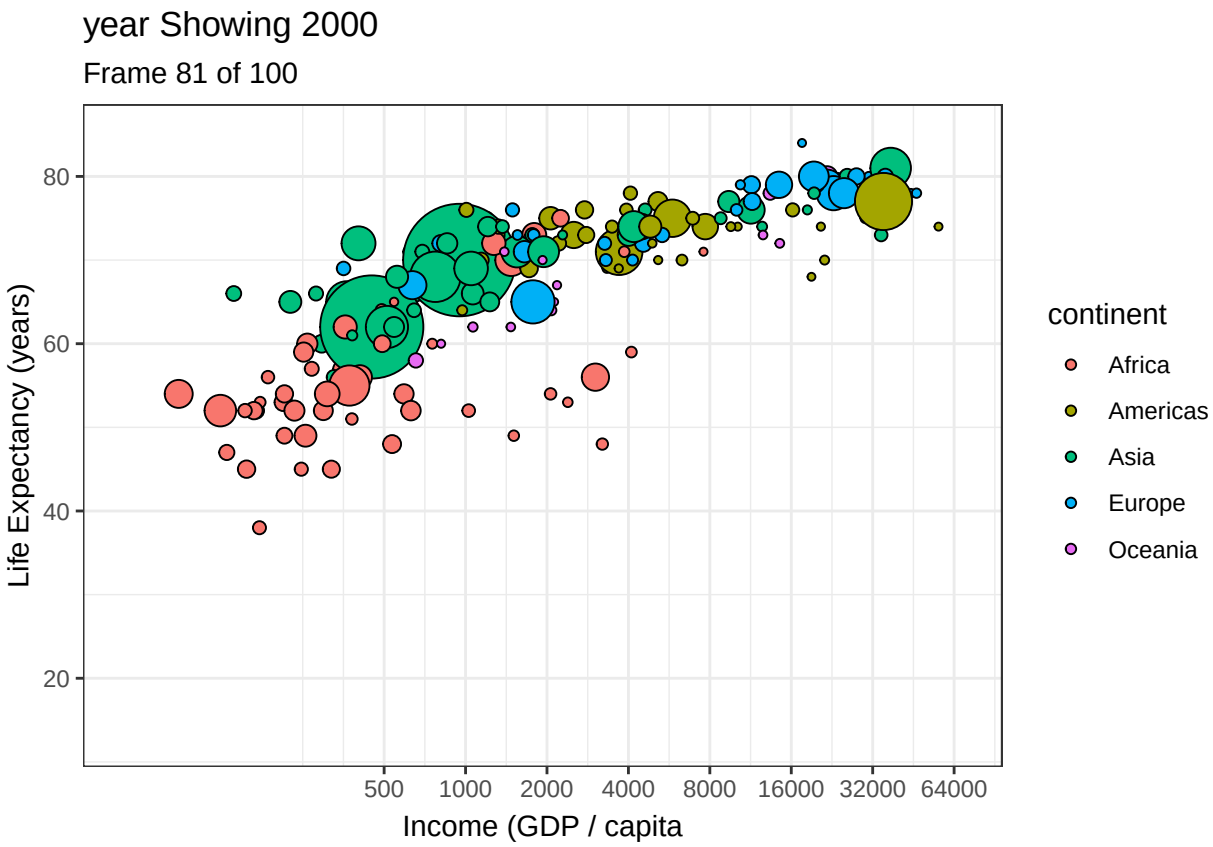

year Showing 1997
Frame 76 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

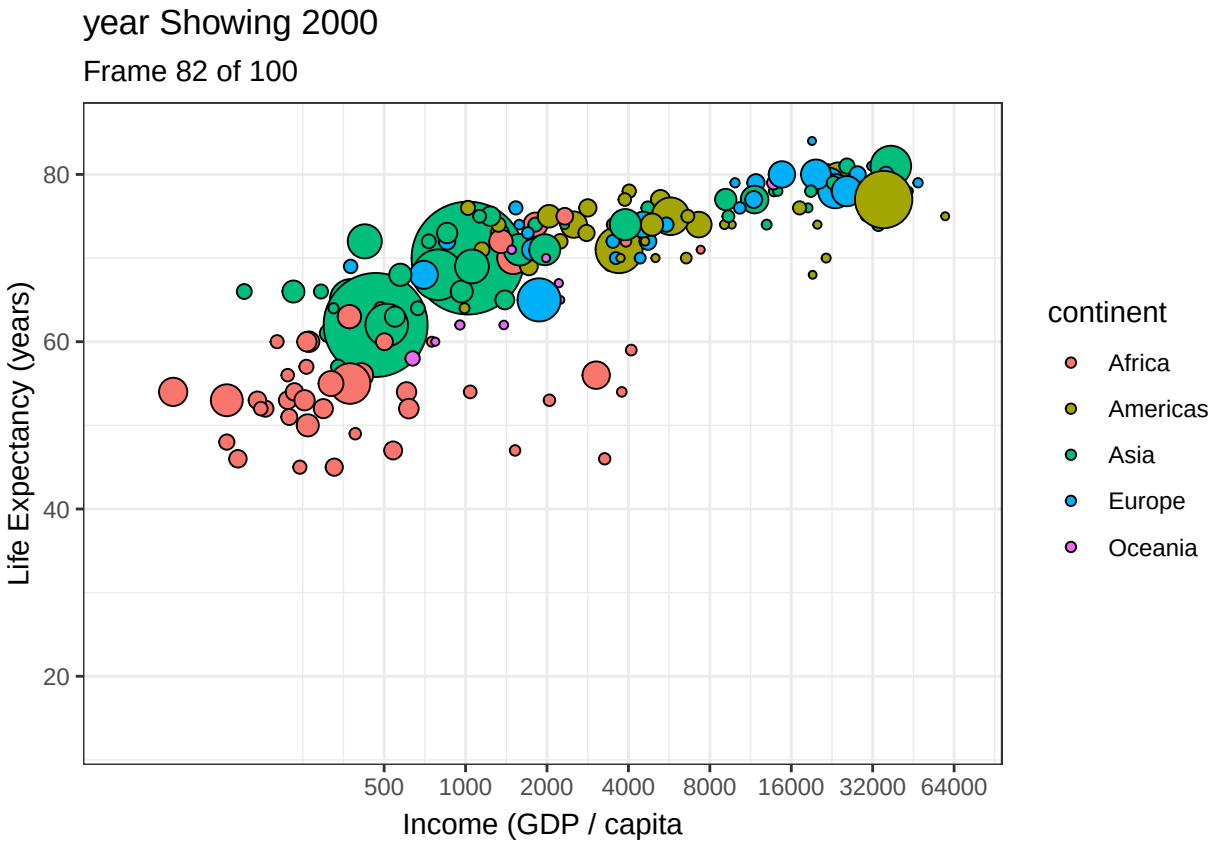

year Showing 1998
Frame 77 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

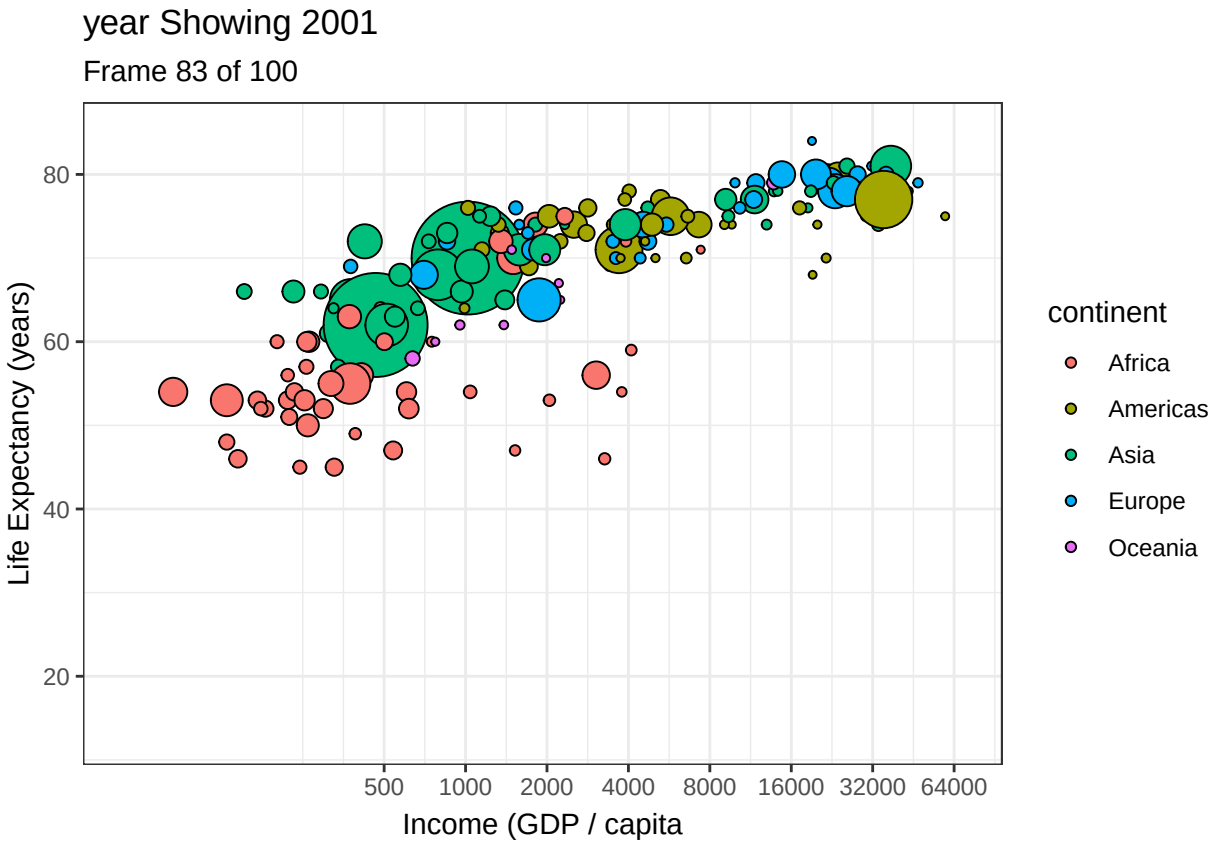

year Showing 1998
Frame 78 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

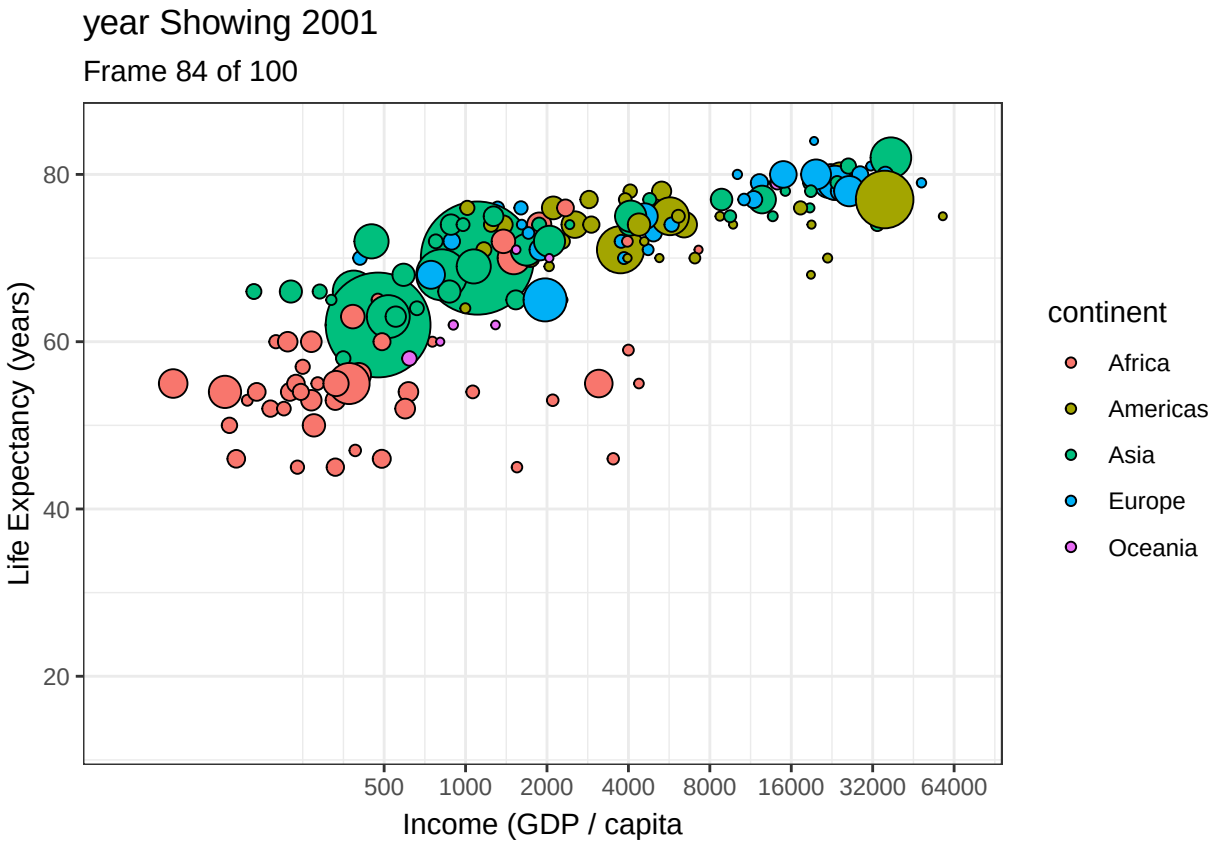

year Showing 1999
Frame 79 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

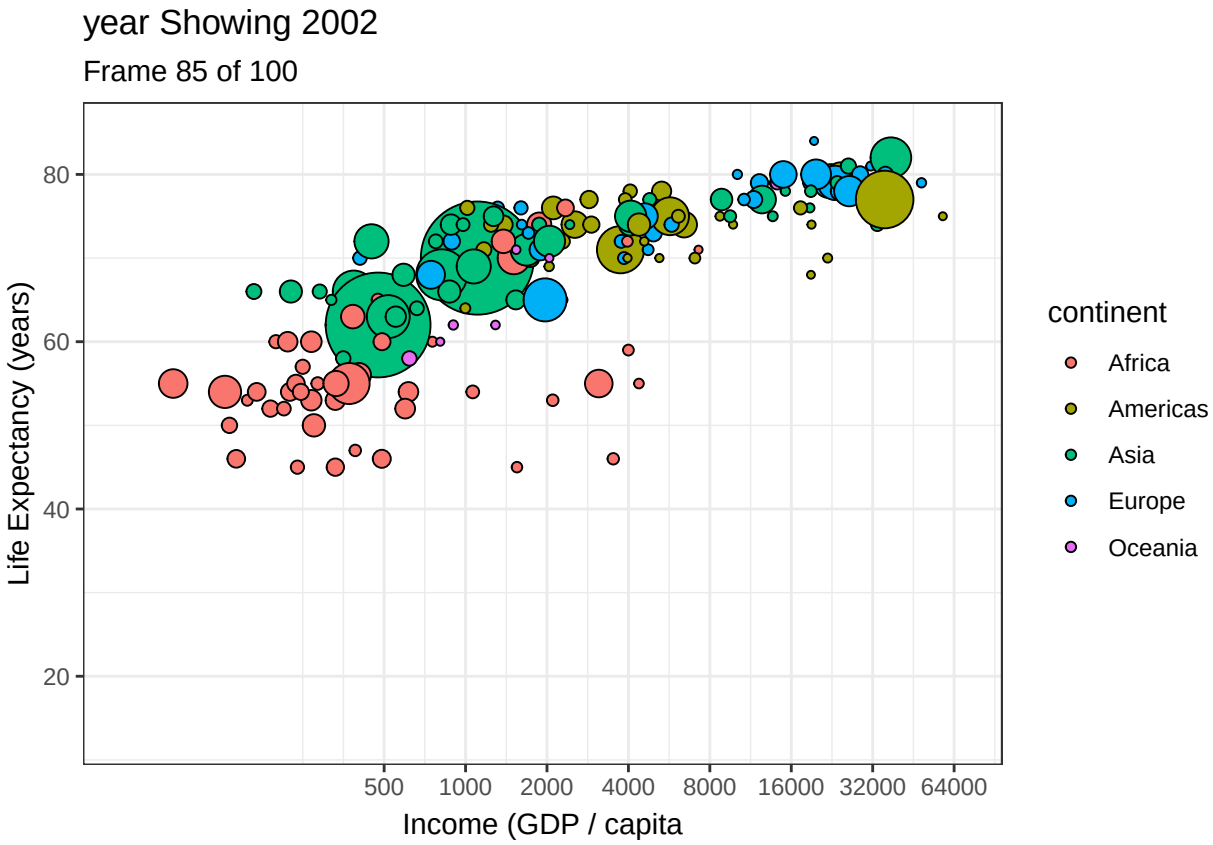

year Showing 1999
Frame 80 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

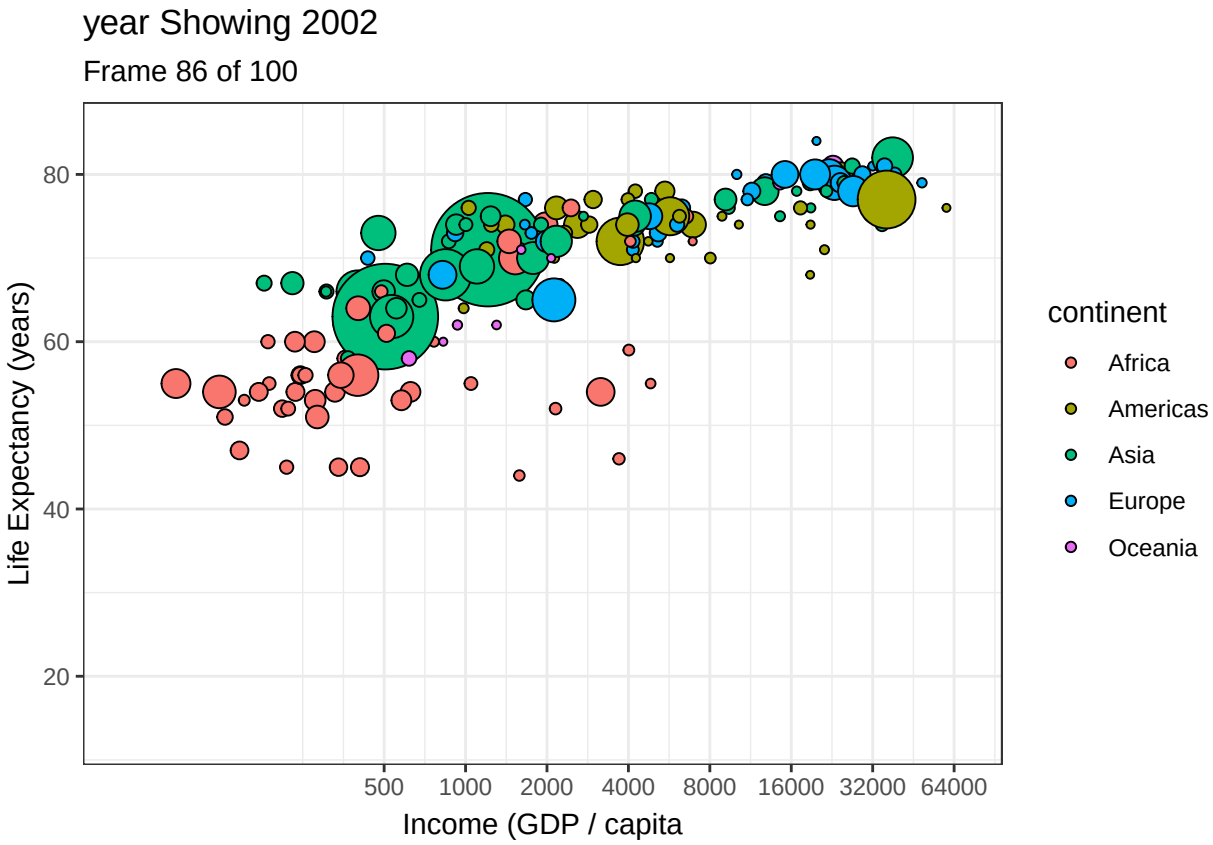

year Showing 2000
Frame 81 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

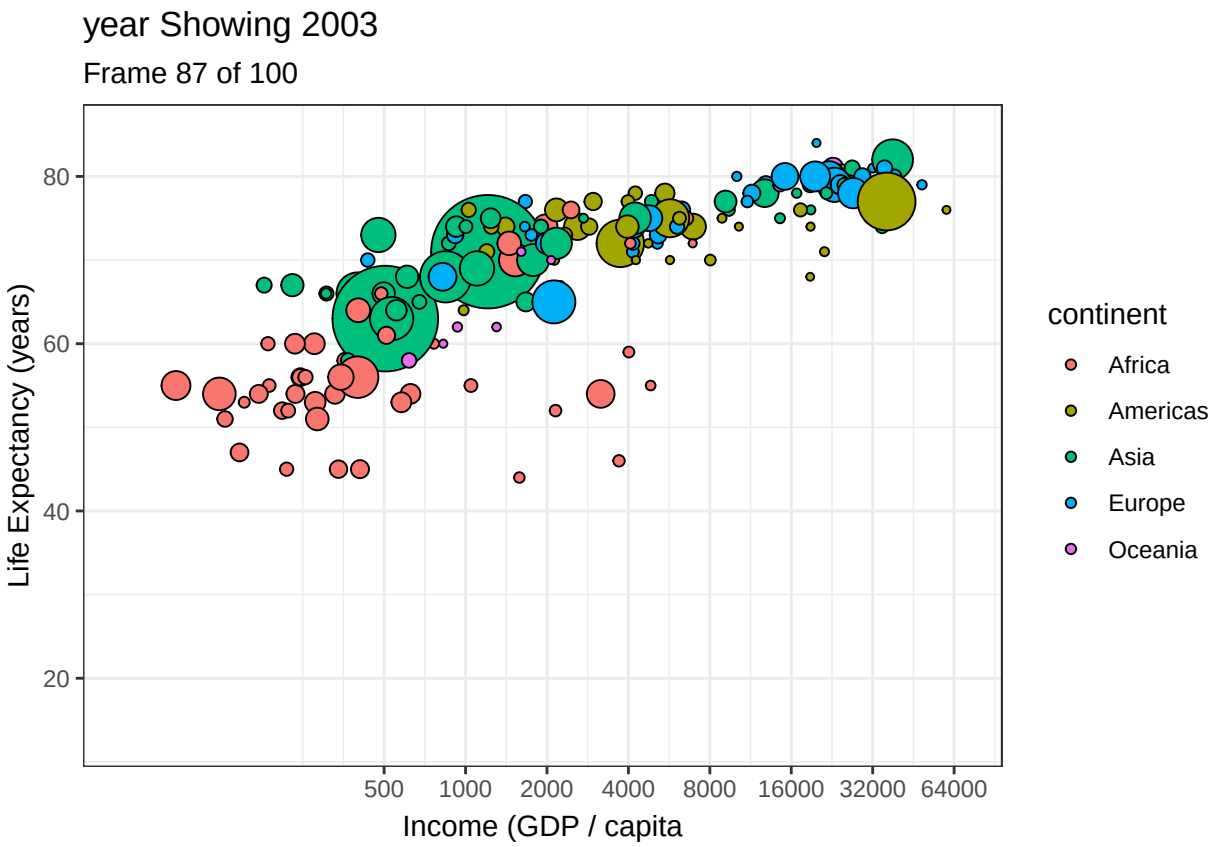

year Showing 2000
Frame 82 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

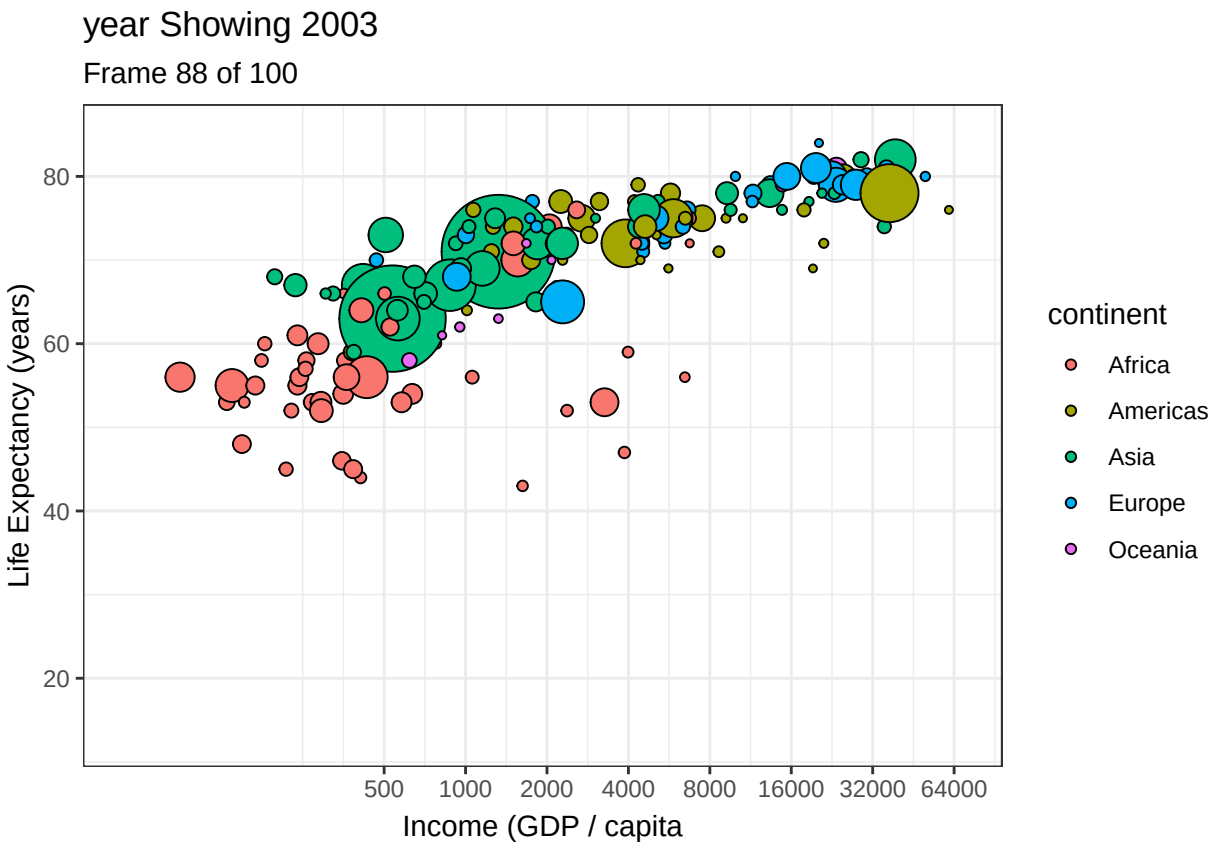

year Showing 2001
Frame 83 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

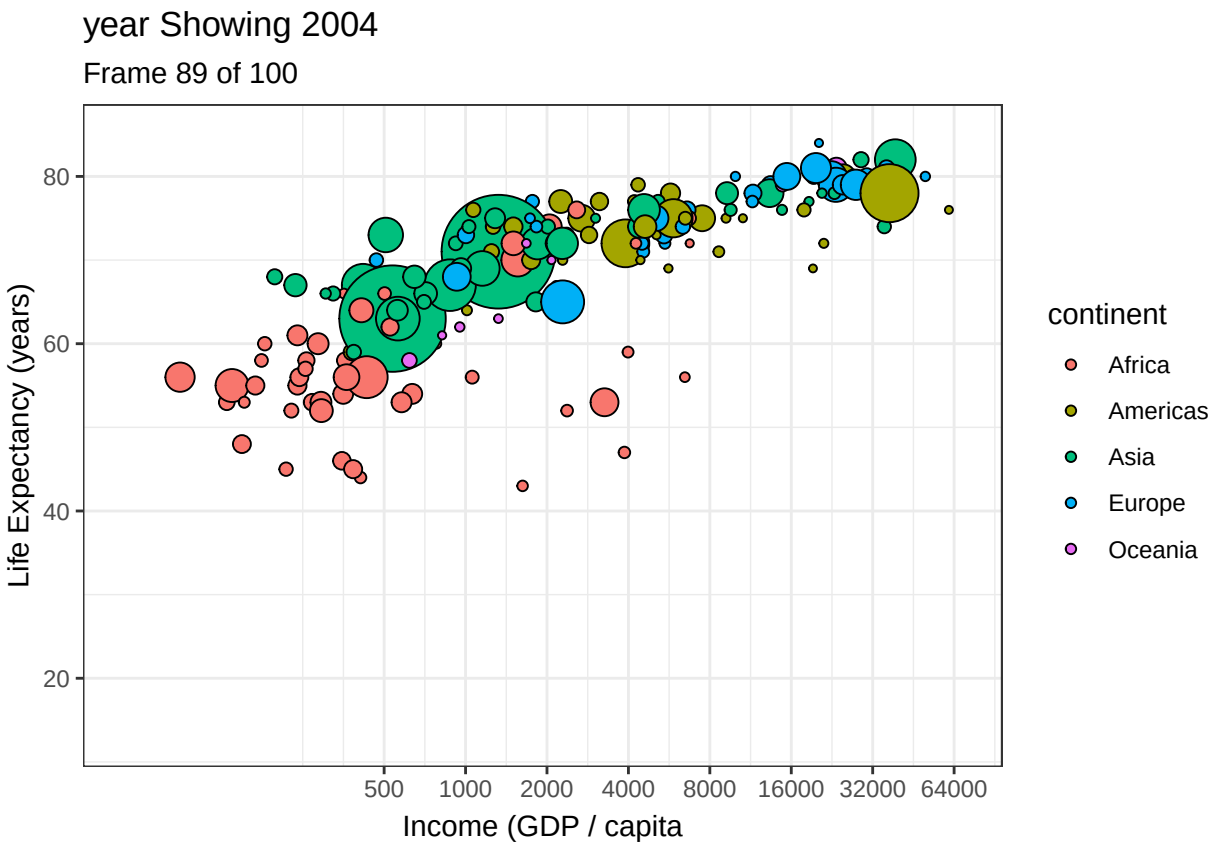

year Showing 2001
Frame 84 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

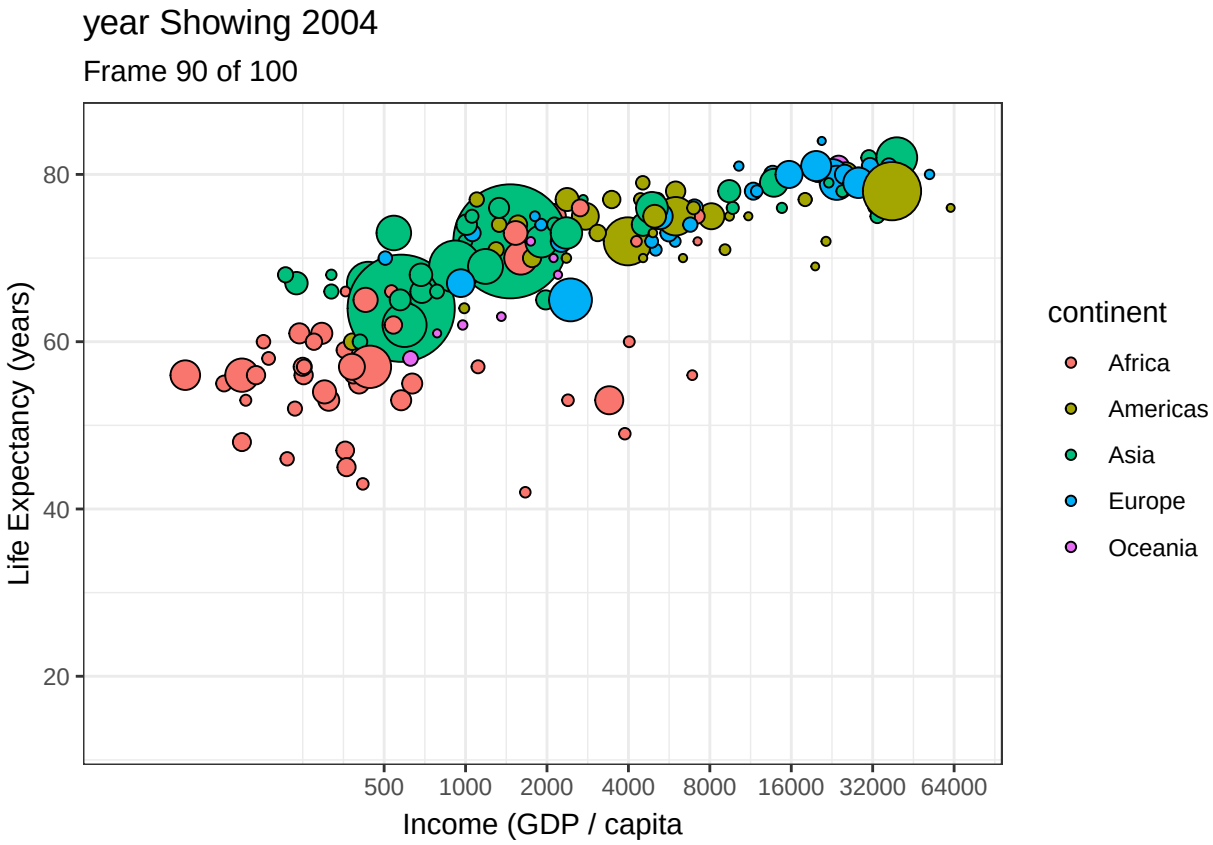

year Showing 2002
Frame 85 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

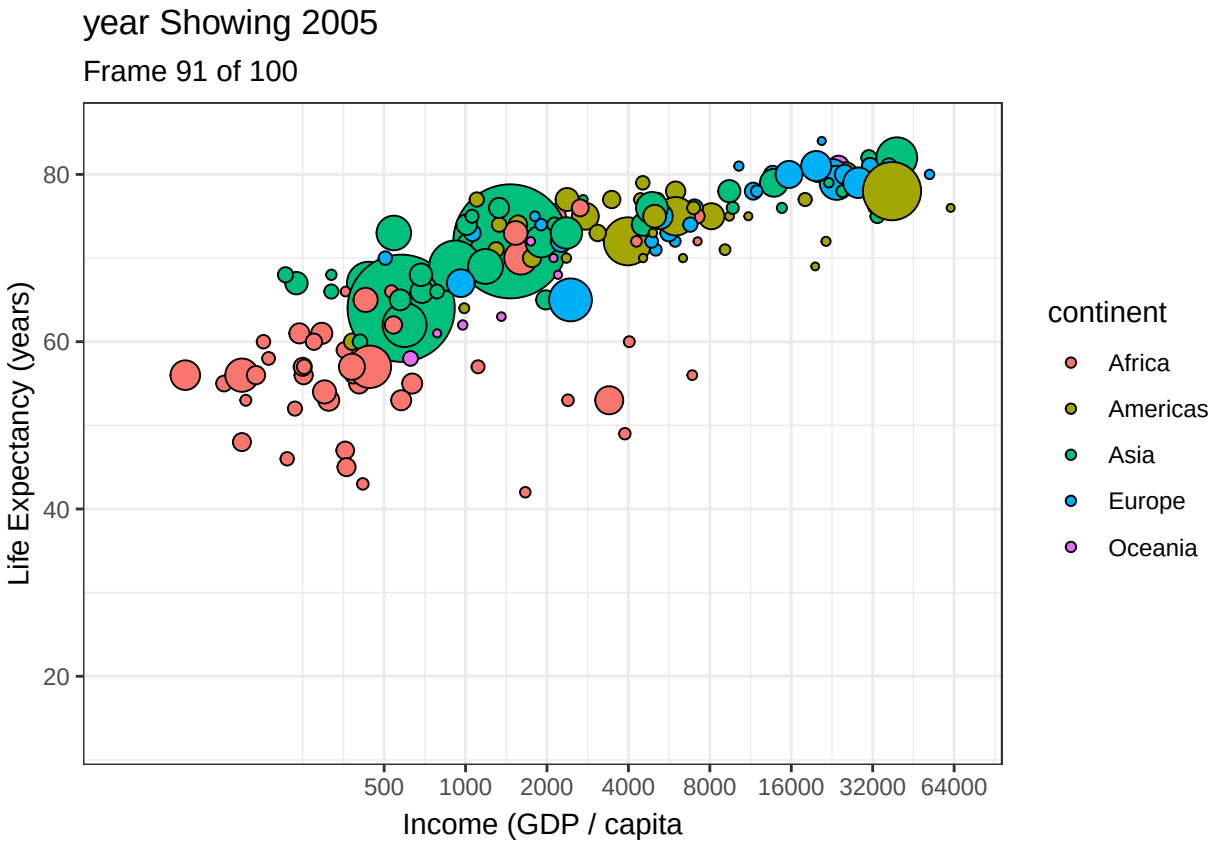

year Showing 2002
Frame 86 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

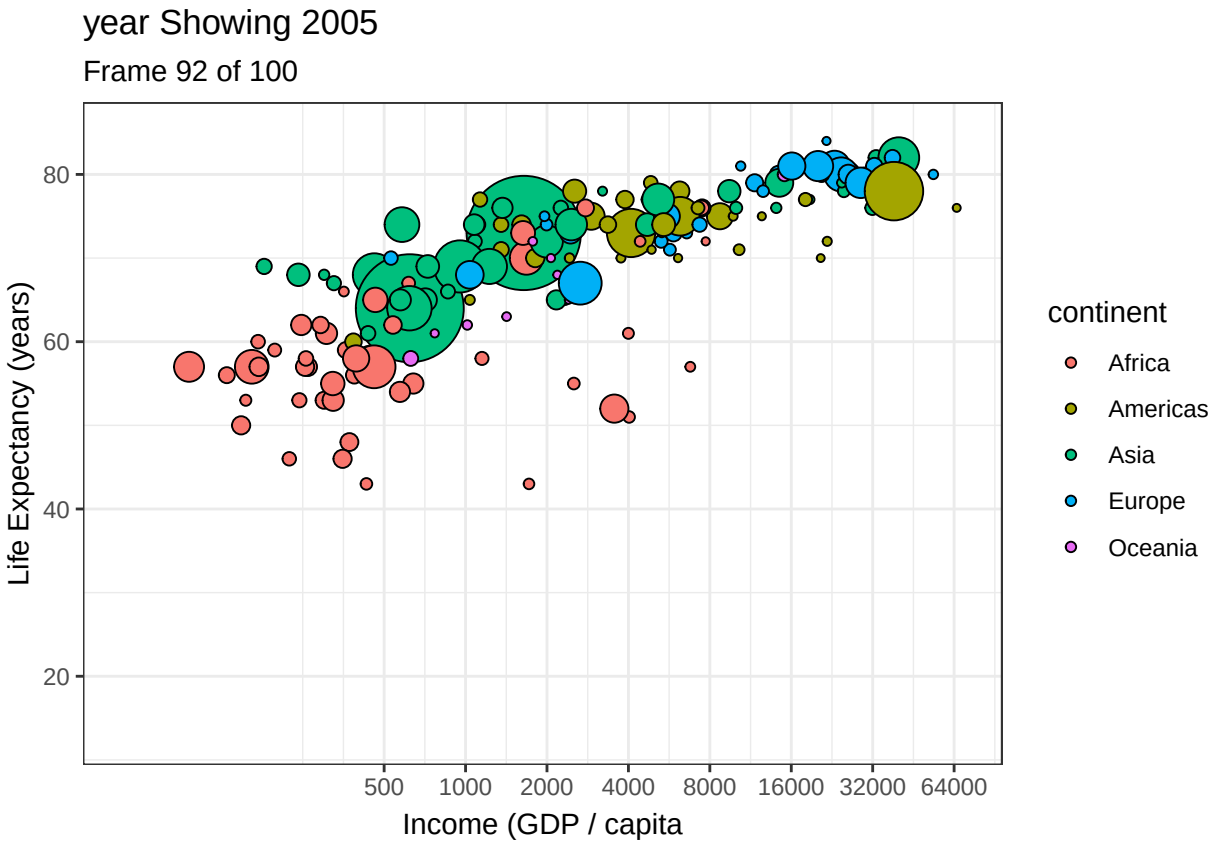

year Showing 2003
Frame 87 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

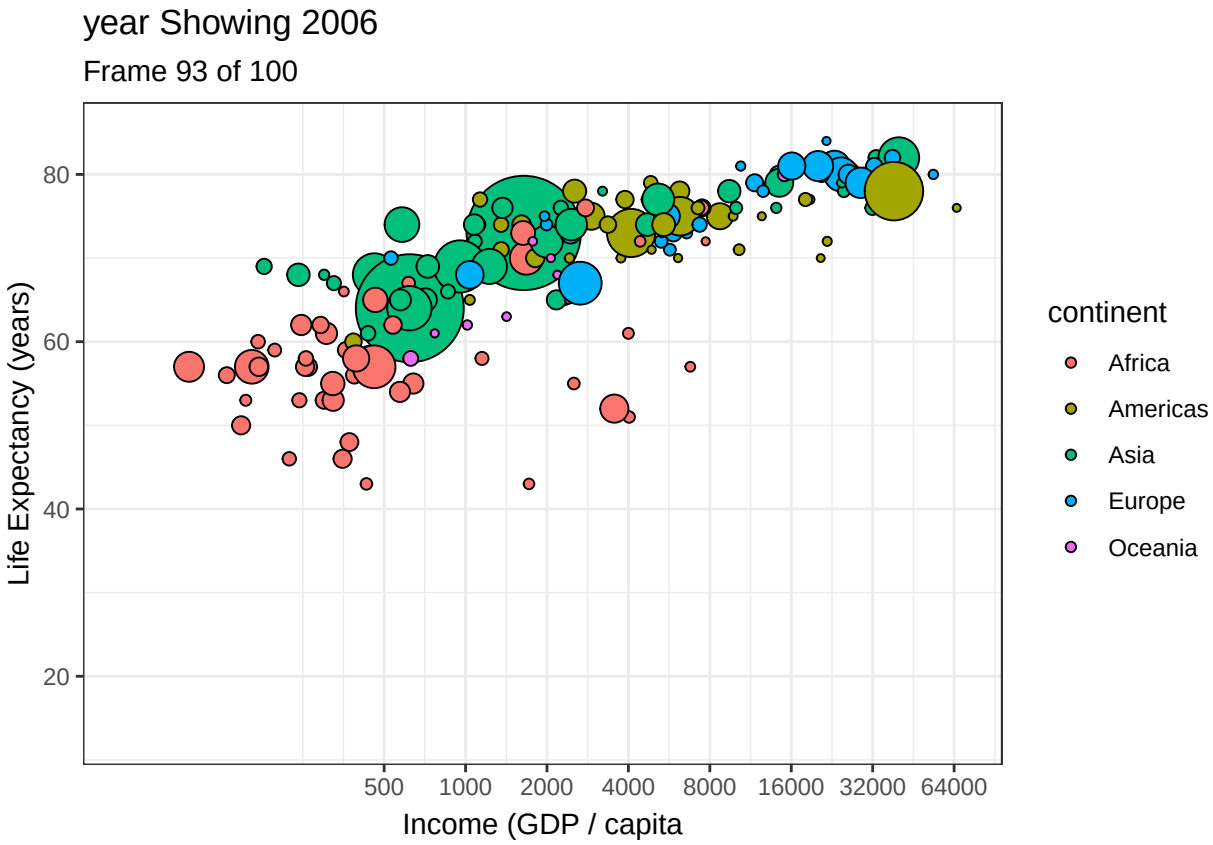

year Showing 2003
Frame 88 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

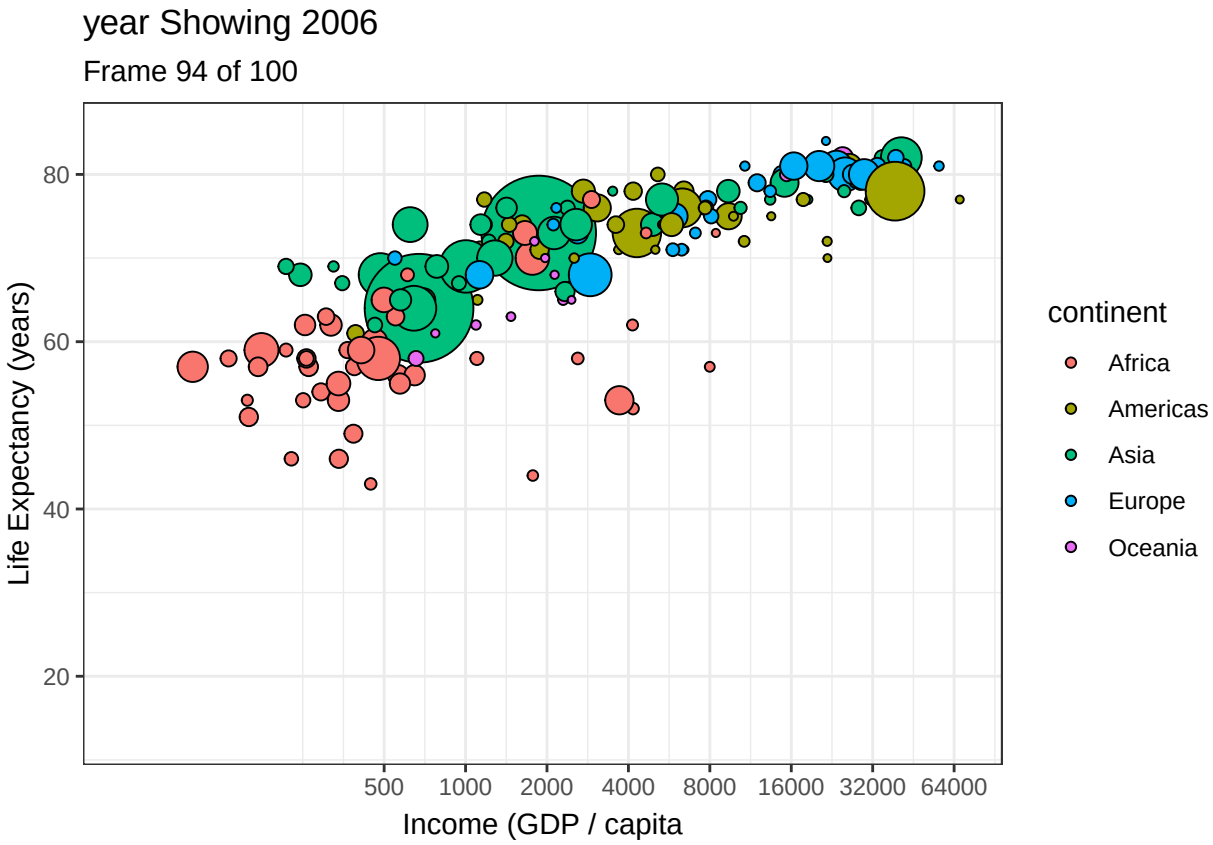

year Showing 2004
Frame 89 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

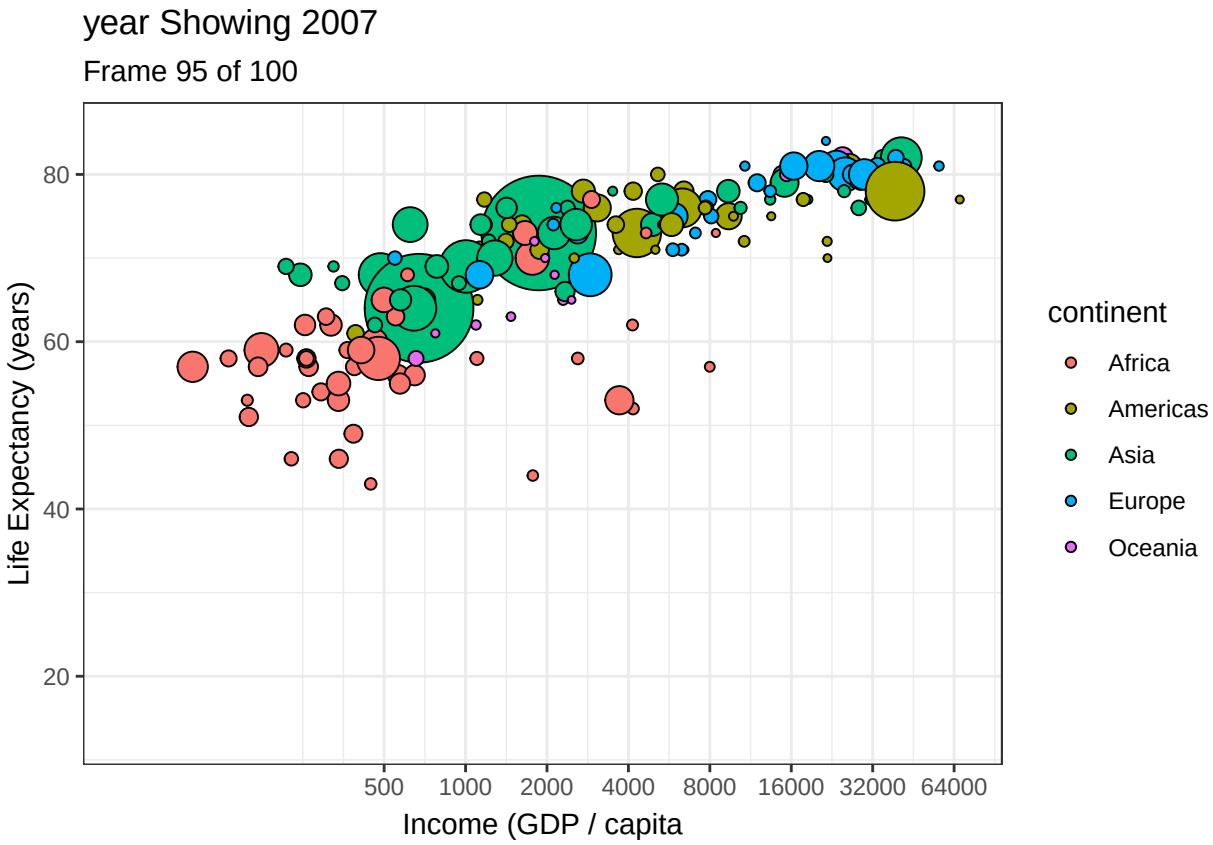

year Showing 2004
Frame 90 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

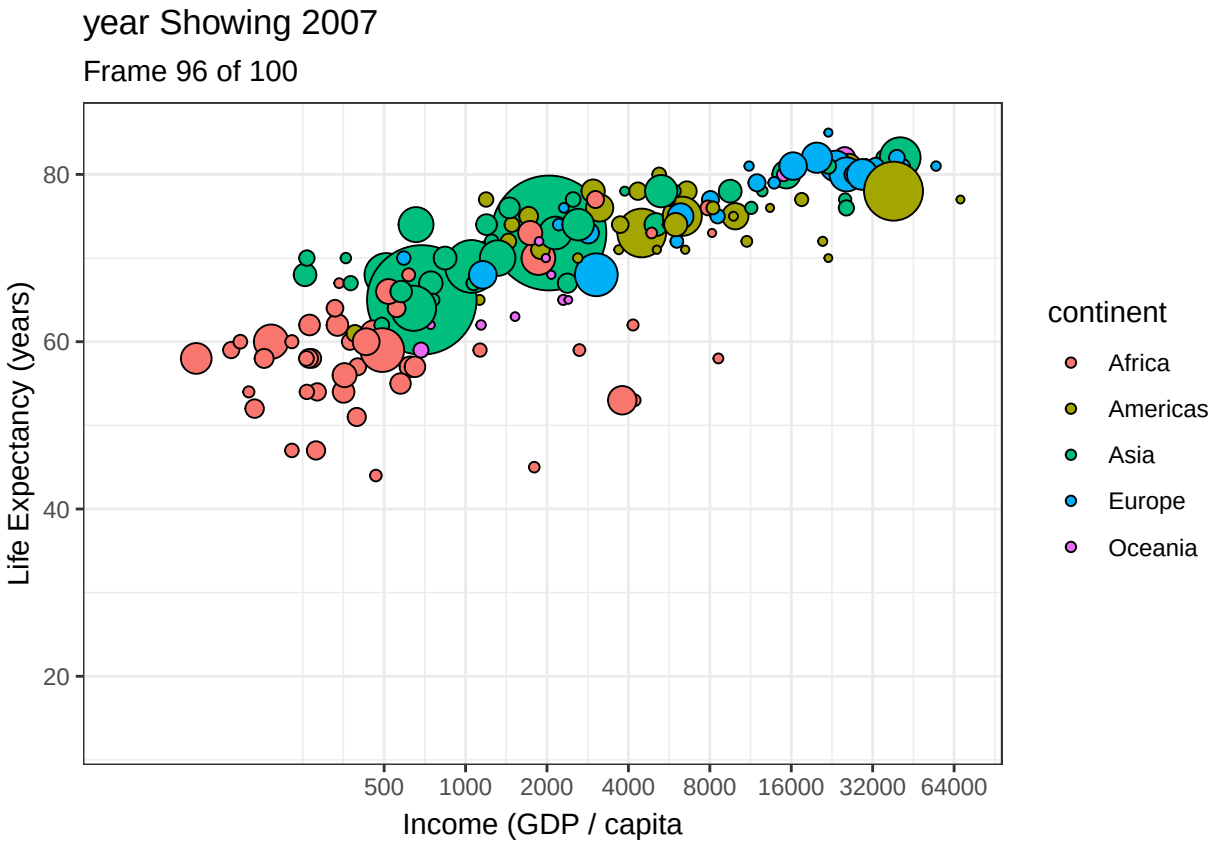

year Showing 2005
Frame 91 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

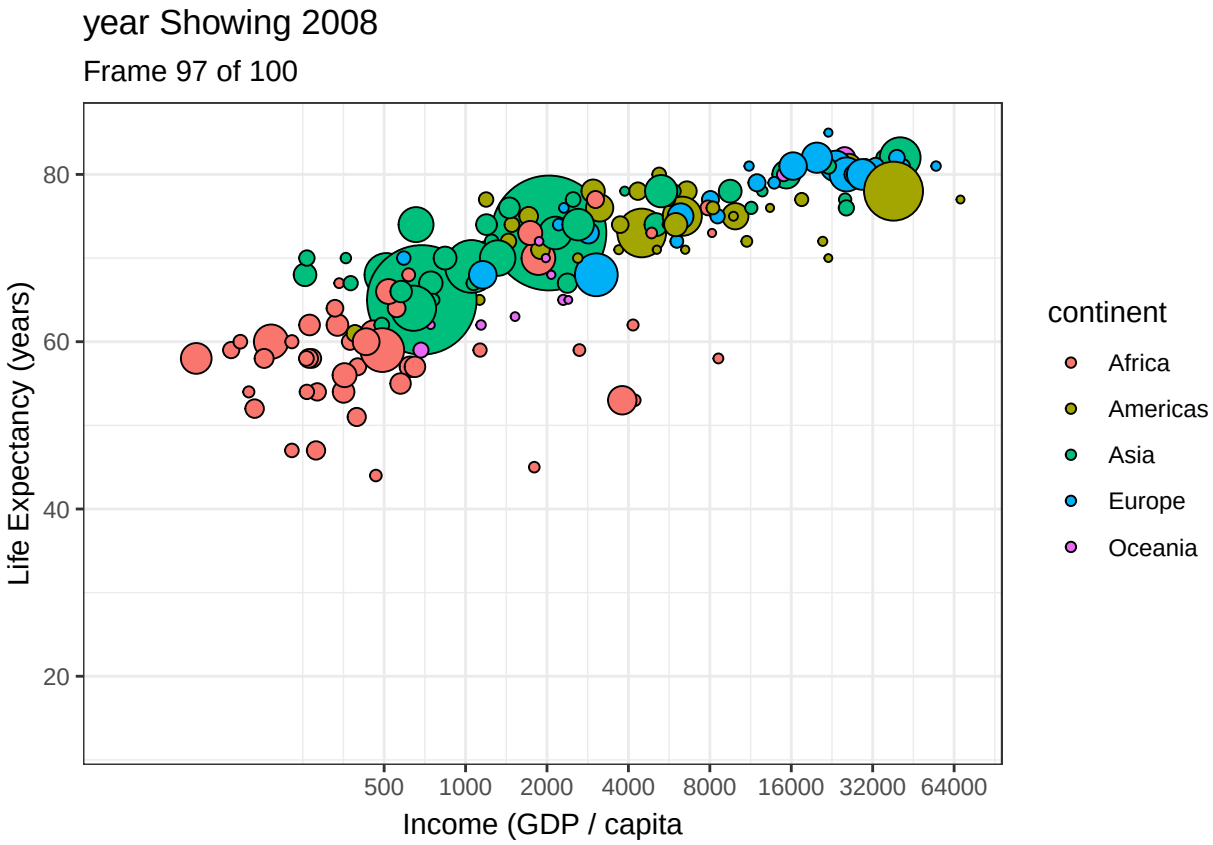

year Showing 2005
Frame 92 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

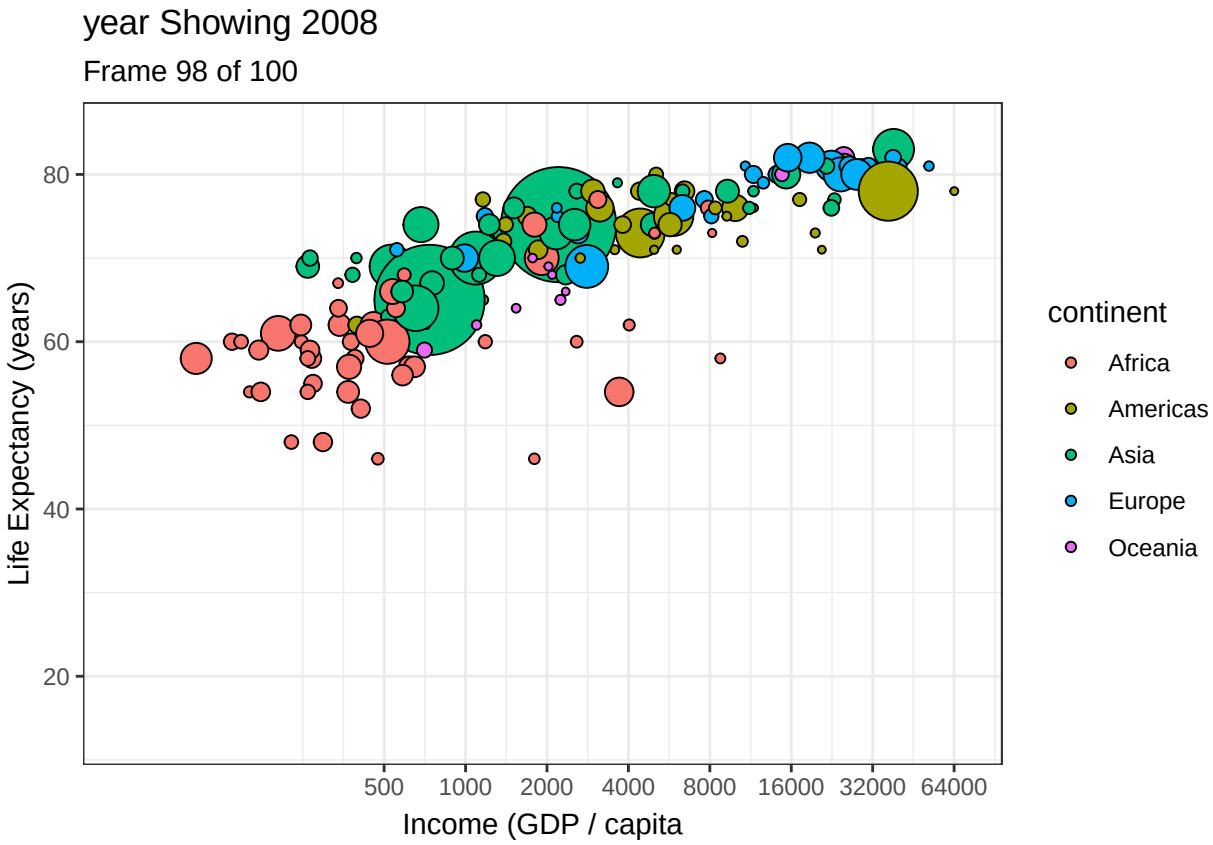

year Showing 2006
Frame 93 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

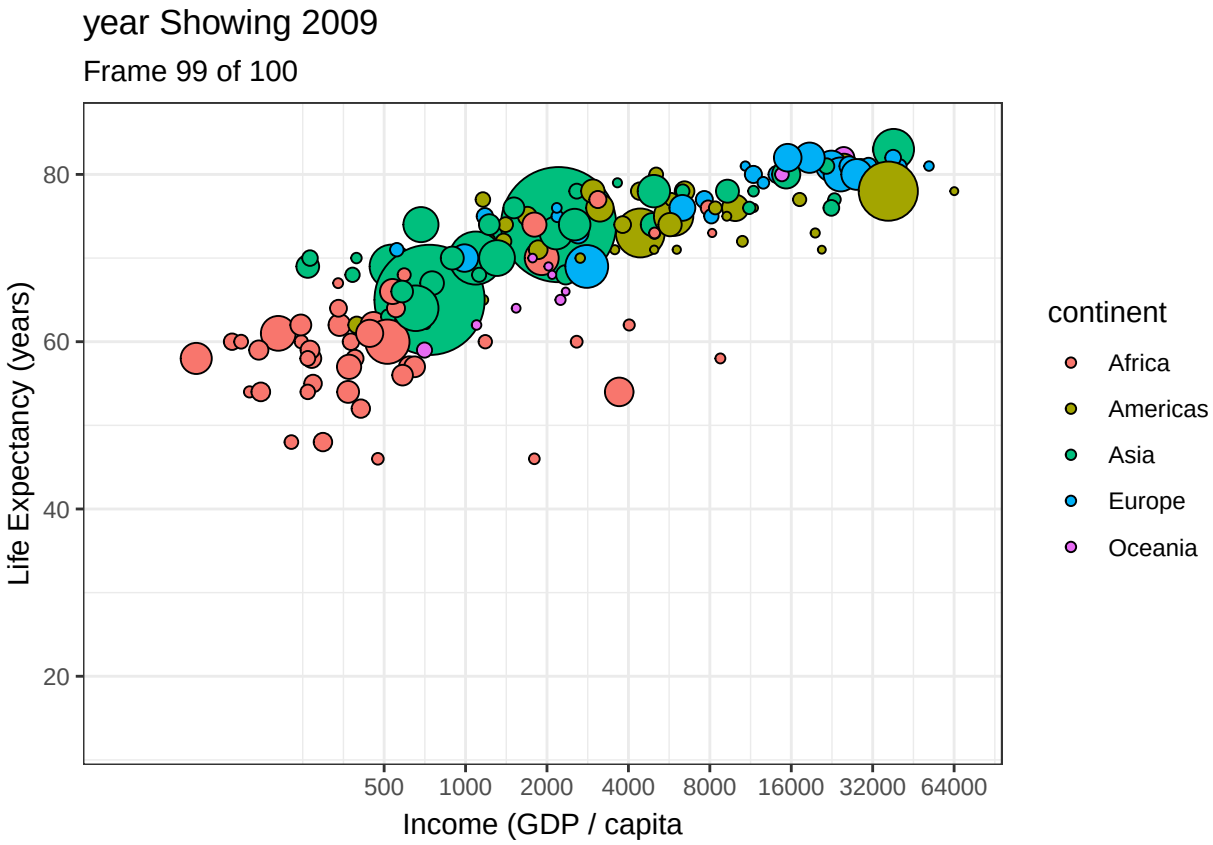

year Showing 2006
Frame 94 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

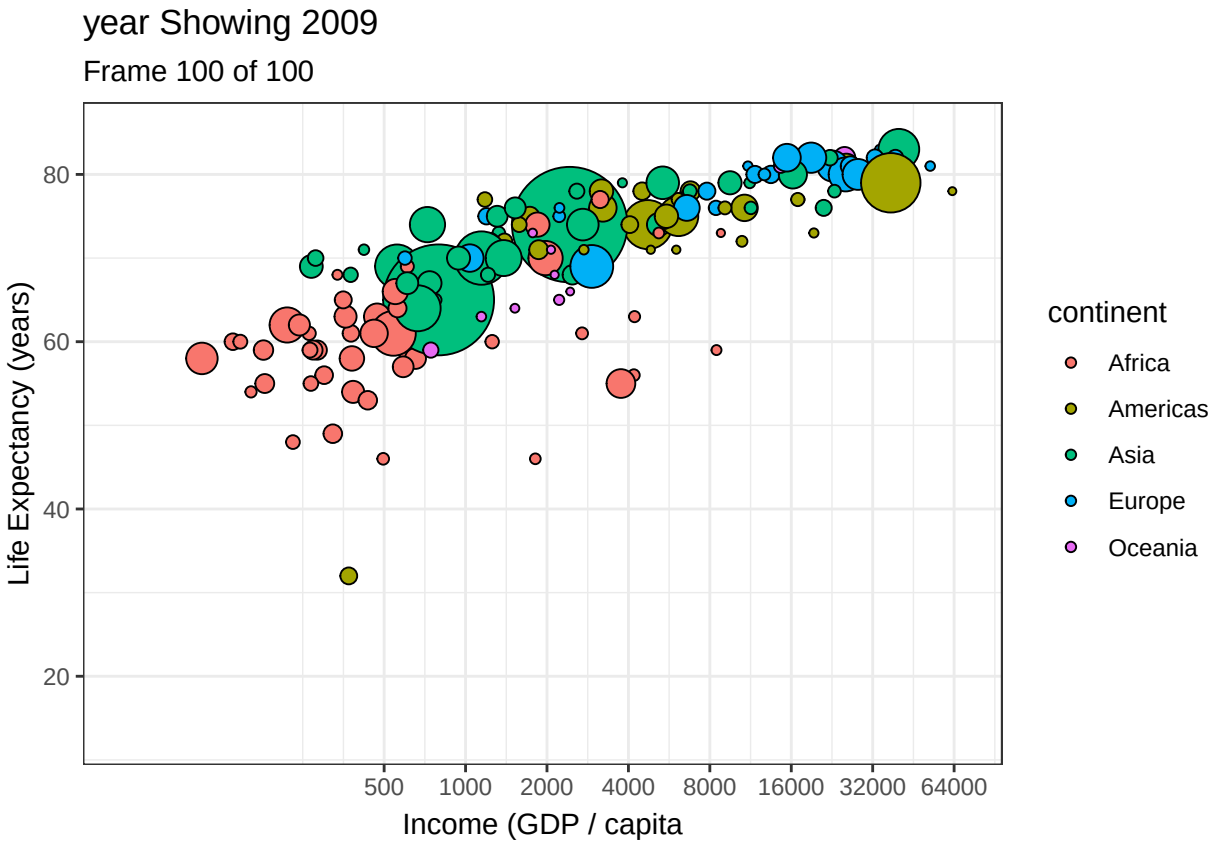

year Showing 2007
Frame 95 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania


year Showing 2007
Frame 96 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania


year Showing 2008
Frame 97 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania


year Showing 2008
Frame 98 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania

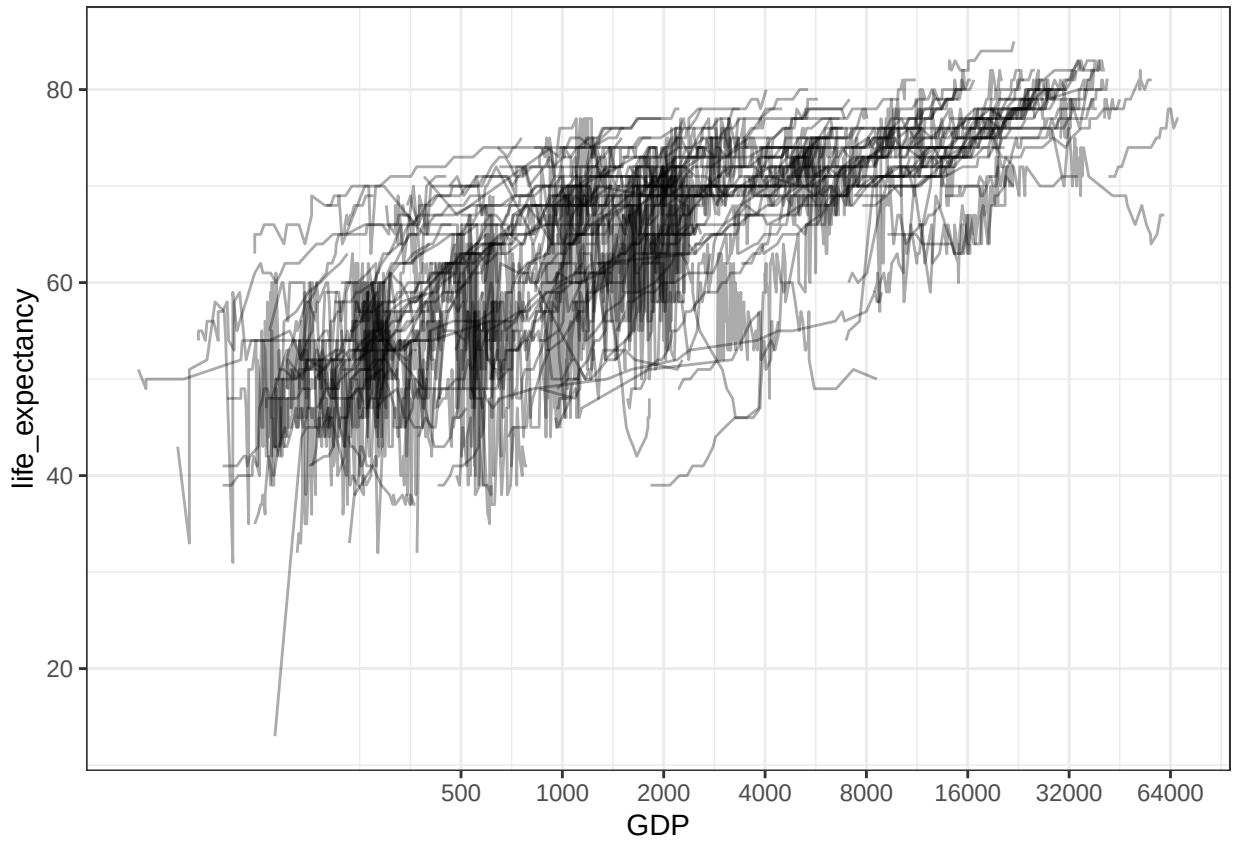

year Showing 2009
Frame 99 of 100
Life Expectancy (years)
80
60
40
20
500
1000
2000
4000
8000
16000
32000
64000
Income (GDP / capita
continent
Africa
Americas
Asia
Europe
Oceania



Part 2: Analyze life expectancy as function of GDP

1. Now, model life expectancy for each country as a function of GDP? (hint – use plot_data from your plot above)

```
plot_data %>%
  ggplot(aes(GDP, life_expectancy, group = country)) +
  geom_line(alpha = 1/3)+
  scale_x_log10(breaks = 2^(-1:7)*1000)
```




```
by_country <- plot_data %>%
  group_by(country, continent) %>%
  nest()

by_country
```

```
## # A tibble: 191 x 3
## # Groups:   country, continent [191]
##    country             continent data
##    <chr>               <chr>     <list>
##  1 Albania             Europe    <tibble [32 x 4]>
##  2 Algeria             Africa    <tibble [52 x 4]>
##  3 Andorra             Europe    <tibble [19 x 4]>
##  4 Angola              Africa    <tibble [27 x 4]>
##  5 Antigua and Barbuda Americas  <tibble [35 x 4]>
##  6 Argentina           Americas  <tibble [52 x 4]>
##  7 Armenia             Asia      <tibble [22 x 4]>
##  8 Aruba               Americas  <tibble [17 x 4]>
##  9 Australia           Oceania   <tibble [52 x 4]>
## 10 Austria             Europe    <tibble [52 x 4]>
## # ... with 181 more rows
```

```
country_mod <- function(df){
  lm(life_expectancy ~ GDP, data = df)
}
```

```
by_country <- by_country %>%
  mutate(model = map(data, country_mod))
by_country
```

```
## # A tibble: 191 x 4
## # Groups:   country, continent [191]
##    country             continent data              model
##    <chr>               <chr>     <list>            <list>
##  1 Albania             Europe    <tibble [32 x 4]> <lm>
##  2 Algeria             Africa    <tibble [52 x 4]> <lm>
##  3 Andorra             Europe    <tibble [19 x 4]> <lm>
##  4 Angola              Africa    <tibble [27 x 4]> <lm>
##  5 Antigua and Barbuda Americas  <tibble [35 x 4]> <lm>
##  6 Argentina           Americas  <tibble [52 x 4]> <lm>
##  7 Armenia             Asia      <tibble [22 x 4]> <lm>
##  8 Aruba               Americas  <tibble [17 x 4]> <lm>
##  9 Australia           Oceania   <tibble [52 x 4]> <lm>
## 10 Austria             Europe    <tibble [52 x 4]> <lm>
## # ... with 181 more rows
```

2. Graph your residuals by country and facet by continent.

```
by_country <- by_country %>%
  mutate(resids = map2(data, model, add_residuals))
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
by_country
```

```
## # A tibble: 191 x 5
## # Groups:   country, continent [191]
##    country             continent data              model  resids
##    <chr>               <chr>     <list>            <list> <list>
##  1 Albania             Europe    <tibble [32 x 4]> <lm>   <tibble [32 x 5]>
##  2 Algeria             Africa    <tibble [52 x 4]> <lm>   <tibble [52 x 5]>
##  3 Andorra             Europe    <tibble [19 x 4]> <lm>   <tibble [19 x 5]>
##  4 Angola              Africa    <tibble [27 x 4]> <lm>   <tibble [27 x 5]>
##  5 Antigua and Barbuda Americas  <tibble [35 x 4]> <lm>   <tibble [35 x 5]>
##  6 Argentina           Americas  <tibble [52 x 4]> <lm>   <tibble [52 x 5]>
##  7 Armenia             Asia      <tibble [22 x 4]> <lm>   <tibble [22 x 5]>
##  8 Aruba               Americas  <tibble [17 x 4]> <lm>   <tibble [17 x 5]>
##  9 Australia           Oceania   <tibble [52 x 4]> <lm>   <tibble [52 x 5]>
## 10 Austria             Europe    <tibble [52 x 4]> <lm>   <tibble [52 x 5]>
## # ... with 181 more rows
```

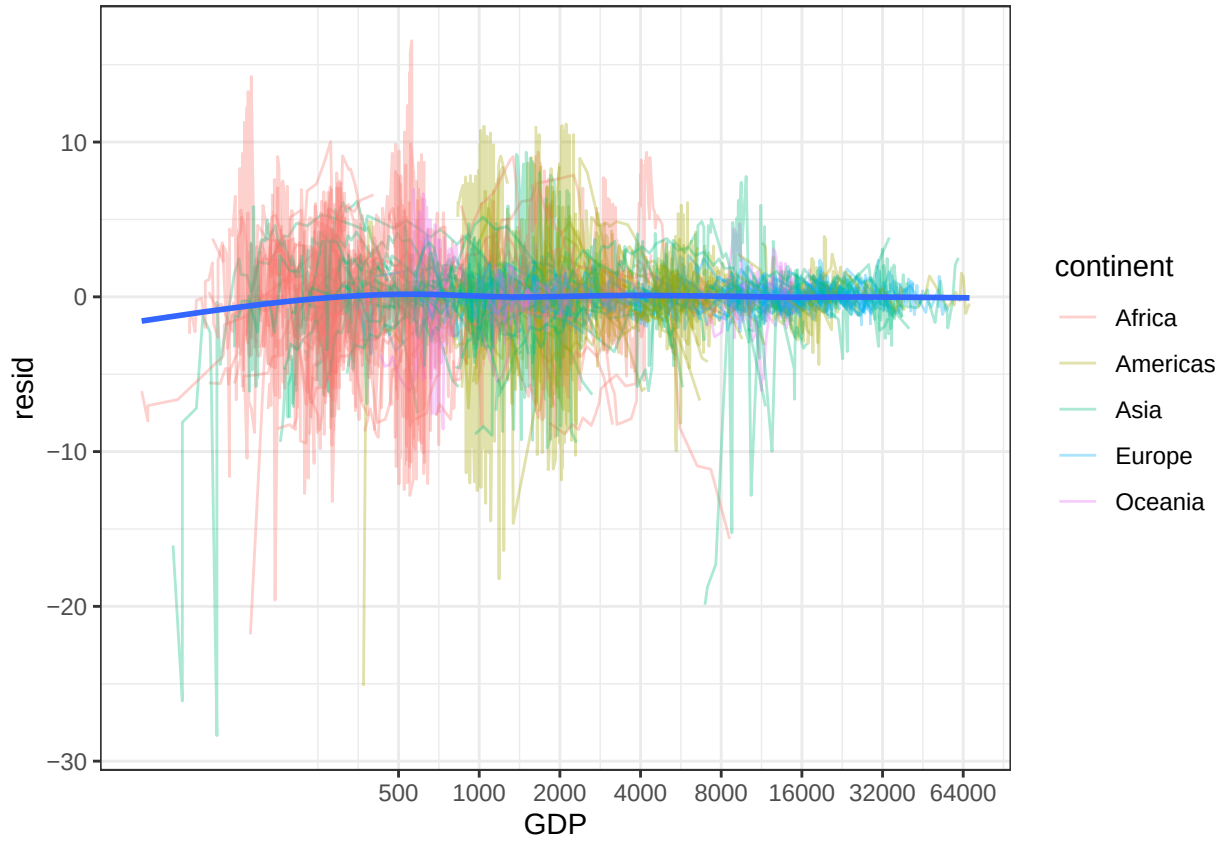
```
resids <- unnest(by_country, resids)
resids
```

```
## # A tibble: 7,682 x 9
```

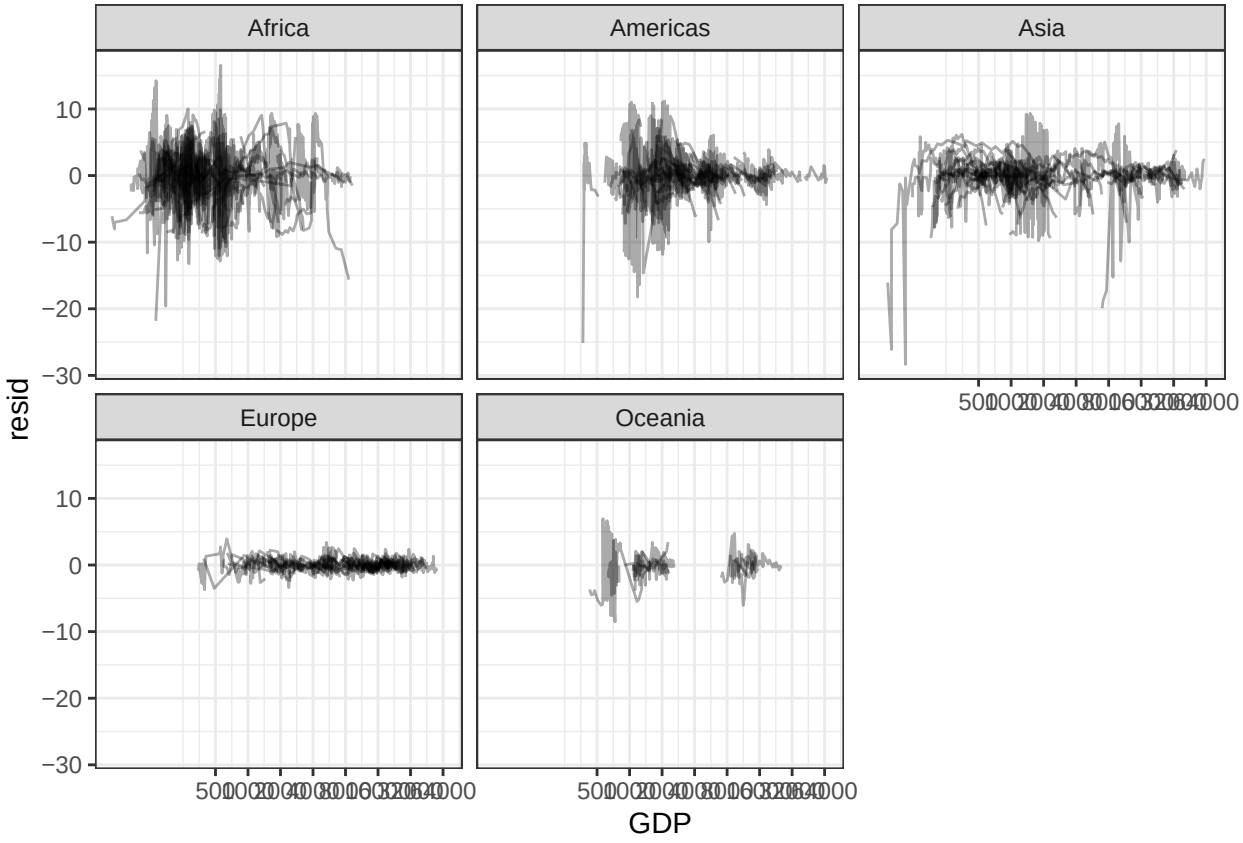
```
## # Groups:   country, continent [191]
##    country continent data    model  year   GDP life_expectancy population  resid
##    <chr>   <chr>     <list>  <lis> <dbl> <dbl>           <dbl>      <dbl>  <dbl>
##  1 Albania Europe    <tibbl~ <lm>   1980  1061              71    2681245 -2.75
##  2 Albania Europe    <tibbl~ <lm>   1981  1100              72    2735329 -1.90
##  3 Albania Europe    <tibbl~ <lm>   1982  1111              72    2788315 -1.94
##  4 Albania Europe    <tibbl~ <lm>   1983  1101              72    2842620 -1.90
##  5 Albania Europe    <tibbl~ <lm>   1984  1065              72    2901590 -1.77
##  6 Albania Europe    <tibbl~ <lm>   1985  1060              73    2966799 -0.747
##  7 Albania Europe    <tibbl~ <lm>   1986  1092              73    3041003 -0.868
##  8 Albania Europe    <tibbl~ <lm>   1987  1054              73    3121336 -0.724
##  9 Albania Europe    <tibbl~ <lm>   1988  1014              73    3197064 -0.574
## 10 Albania Europe    <tibbl~ <lm>   1989  1092              73    3253659 -0.868
## # ... with 7,672 more rows
```

```
resids %>%
  ggplot(aes(GDP,resid)) +
  geom_line(aes(group = country, color = continent), alpha = 1/3) +
  scale_x_log10(breaks = 2^(-1:7)*1000)+
  geom_smooth(se = FALSE)
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



```
resids %>%
  ggplot(aes(GDP, resid, group = country)) +
  geom_line(alpha = 1/3) +
  scale_x_log10(breaks = 2^(-1:7)*1000)+
  facet_wrap(~continent)
```



3. Are there countries and continents for which this is a particularly bad model? Use broom::glance to make this determination and provide an explanation for your conclusions.

```
glance <- by_country %>%
  mutate(glance = map(model, glance)) %>%
  unnest(glance, .drop = TRUE)
```

```
## Warning: The `.drop` argument of `unnest()` is deprecated as of tidyr 1.0.0.
## All list-columns are now preserved.
## This warning is displayed once per session.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
glance
```

```
## # A tibble: 191 x 16
## # Groups:   country, continent [191]
##    country continent data  model resids r.squared adj.r.squared sigma statistic
```

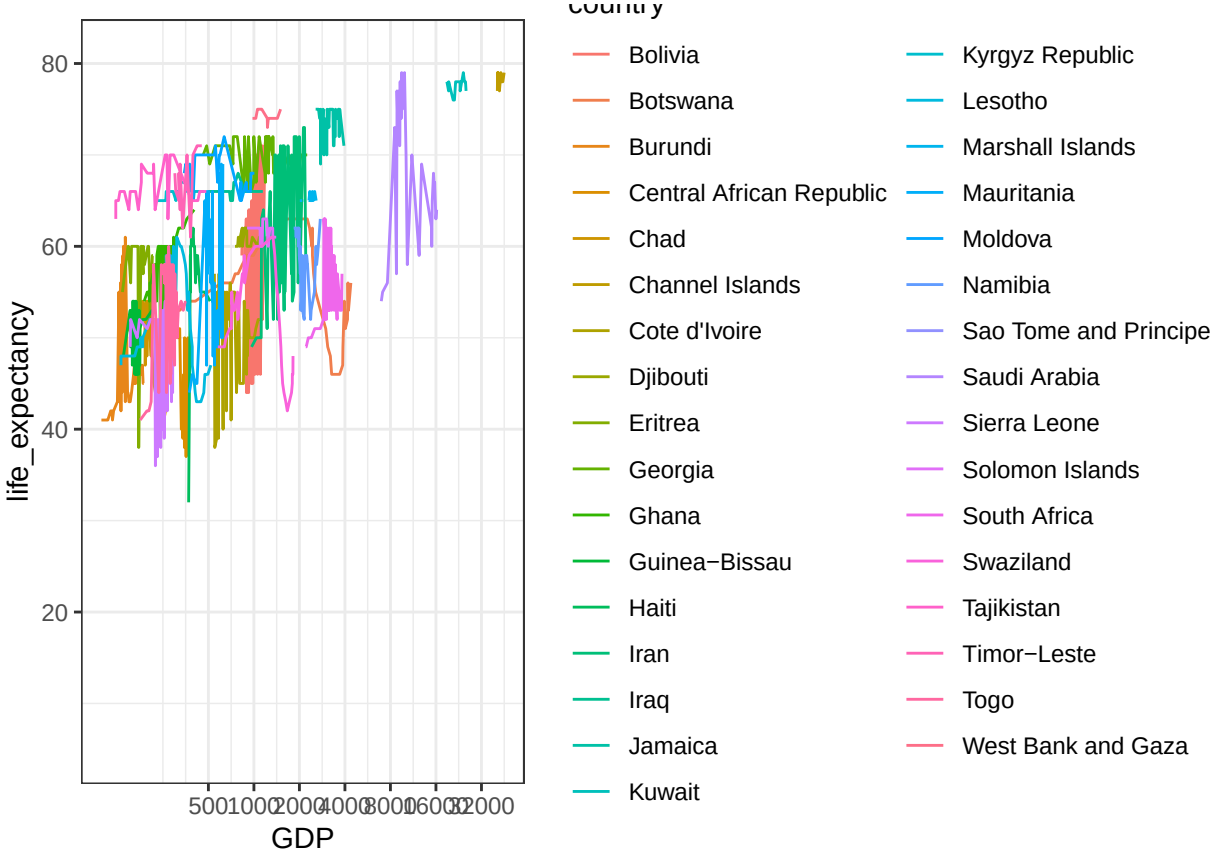
```
##    <chr>   <chr>     <lis> <lis> <list>     <dbl>         <dbl> <dbl>     <dbl>
##  1 Albania Europe    <tib~ <lm>  <tibb~     0.585         0.571 1.16       42.3
##  2 Algeria Africa    <tib~ <lm>  <tibb~     0.697         0.691 5.47      115.
##  3 Andorra Europe    <tib~ <lm>  <tibb~     0.720         0.703 0.515      43.7
##  4 Angola  Africa    <tib~ <lm>  <tibb~     0.507         0.488 2.41       25.8
##  5 Antigu~ Americas  <tib~ <lm>  <tibb~     0.864         0.860 0.650     209.
##  6 Argent~ Americas  <tib~ <lm>  <tibb~     0.523         0.513 2.36       54.8
##  7 Armenia Asia      <tib~ <lm>  <tibb~     0.747         0.734 0.819      59.0
##  8 Aruba   Americas  <tib~ <lm>  <tibb~     0.533         0.502 0.308      17.1
##  9 Austra~ Oceania   <tib~ <lm>  <tibb~     0.956         0.955 0.819    1088.
## 10 Austria Europe    <tib~ <lm>  <tibb~     0.977         0.976 0.576    2092.
## # ... with 181 more rows, and 7 more variables: p.value <dbl>, df <int>,
## #   logLik <dbl>, AIC <dbl>, BIC <dbl>, deviance <dbl>, df.residual <int>
```

```
glance %>%
  arrange(r.squared)
```

```
## # A tibble: 191 x 16
## # Groups:   country, continent [191]
##    country continent data  model resids r.squared adj.r.squared   sigma
##    <chr>   <chr>     <lis> <lis> <list>     <dbl>         <dbl>   <dbl>
##  1 Sao To~ Africa    <tib~ <lm>  <tibb~   0.             0      NaN
##  2 Haiti   Americas  <tib~ <lm>  <tibb~   4.49e-7       -0.0526   6.40
##  3 Eritrea Africa    <tib~ <lm>  <tibb~   8.44e-5       -0.0555   5.23
##  4 Burundi Africa    <tib~ <lm>  <tibb~   4.27e-4       -0.0196   5.34
##  5 Sierra~ Africa    <tib~ <lm>  <tibb~   6.30e-4       -0.0194   5.65
##  6 West B~ Asia      <tib~ <lm>  <tibb~   1.07e-3       -0.0988   0.605
##  7 Iraq    Asia      <tib~ <lm>  <tibb~   1.13e-3       -0.0757   0.865
##  8 Jamaica Americas  <tib~ <lm>  <tibb~   1.32e-3       -0.0280   1.85
##  9 Cote d~ Africa    <tib~ <lm>  <tibb~   2.45e-3       -0.0175   5.31
## 10 Tajiki~ Asia      <tib~ <lm>  <tibb~   4.11e-3       -0.0357   2.19
## # ... with 181 more rows, and 8 more variables: statistic <dbl>, p.value <dbl>,
## #   df <int>, logLik <dbl>, AIC <dbl>, BIC <dbl>, deviance <dbl>,
## #   df.residual <int>
```

```
bad_fit <- filter(glance, r.squared < 0.2)
new_gapminder %>%
  semi_join(bad_fit, by = "country") %>%
  ggplot(aes(GDP, life_expectancy, colour = country)) +
  scale_x_log10(breaks = 2^(-1:7)*1000)+
  geom_line()
```

```
## Warning: Removed 5717 rows containing missing values (geom_path).
```



these are bad fits as there rsquare value is less than 0.2 which means there is less corelation between the life Expectency and GDP.

Part 3:

a. Transform year so that it has a mean of 0

```
transform_data<-plot_data

transform_data$scale_year<-scale(transform_data$year)
mean(transform_data$scale_year)
```

```
## [1] 4.998686e-15
```

##close to zero

b. Model with a quadratic polynomial. How can you interpret the coefficients of the quadratic for Belgium?

```
country_mod1 <- function(df){
  lm(life_expectancy ~ year+scale_year+population, data = df)
}

by_country <- transform_data %>%
  group_by(country, continent) %>%
```

```
  nest()

by_country <- by_country %>%
  mutate(model = map(data, country_mod1))

Belgium<-by_country %>%
  filter(country=="Belgium")

Belgium$model
```

```
## [[1]]
##
## Call:
## lm(formula = life_expectancy ~ year + scale_year + population,
##     data = df)
##
## Coefficients:
## (Intercept)         year   scale_year   population
##  -3.911e+02    2.387e-01           NA   -7.954e-07
```

the coeffxicients here tells yes about how each variable effect the life_expectancy and from what we can see here, scale_year does not effect the life_expectancy and other have less relation as well.

c. Use glance() to identify all countries that do not fit the model well and plot the residuals for the countries that do not fit the model well.

```
by_country <- by_country %>%
  mutate(resids = map2(data, model, add_residuals))
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading

## Warning in predict.lm(model, data): prediction from a rank-deficient fit may be
## misleading
```

```
by_country
```

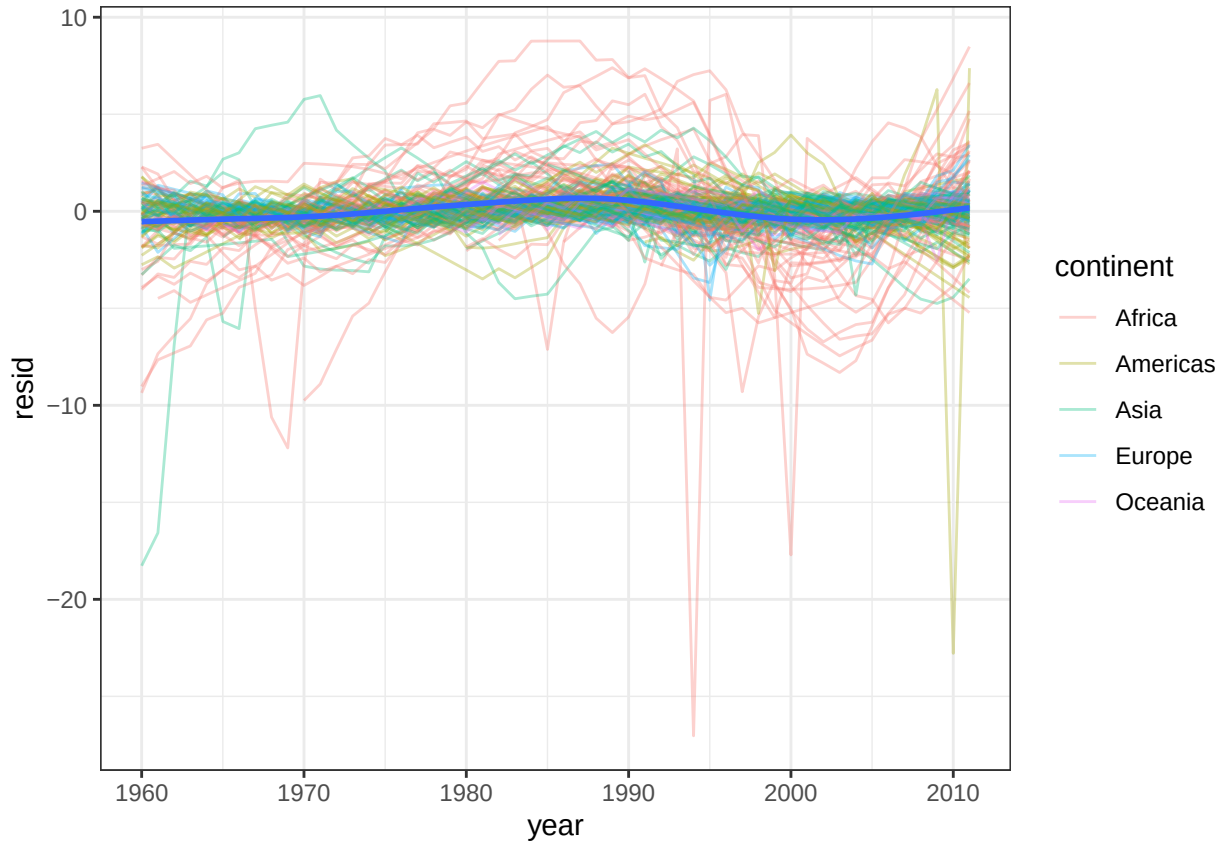
```
## # A tibble: 191 x 5
## # Groups:   country, continent [191]
##    country             continent data              model  resids
##    <chr>               <chr>     <list>            <list> <list>
##  1 Albania             Europe    <tibble [32 x 5]> <lm>   <tibble [32 x 6]>
##  2 Algeria             Africa    <tibble [52 x 5]> <lm>   <tibble [52 x 6]>
##  3 Andorra             Europe    <tibble [19 x 5]> <lm>   <tibble [19 x 6]>
##  4 Angola              Africa    <tibble [27 x 5]> <lm>   <tibble [27 x 6]>
##  5 Antigua and Barbuda Americas  <tibble [35 x 5]> <lm>   <tibble [35 x 6]>
##  6 Argentina           Americas  <tibble [52 x 5]> <lm>   <tibble [52 x 6]>
##  7 Armenia             Asia      <tibble [22 x 5]> <lm>   <tibble [22 x 6]>
##  8 Aruba               Americas  <tibble [17 x 5]> <lm>   <tibble [17 x 6]>
##  9 Australia           Oceania   <tibble [52 x 5]> <lm>   <tibble [52 x 6]>
## 10 Austria             Europe    <tibble [52 x 5]> <lm>   <tibble [52 x 6]>
## # ... with 181 more rows
```

```
resids <- unnest(by_country, resids)
resids
```

```
## # A tibble: 7,682 x 10
## # Groups:   country, continent [191]
##    country continent data  model  year   GDP life_expectancy population
##    <chr>   <chr>     <lis> <lis> <dbl> <dbl>           <dbl>      <dbl>
##  1 Albania Europe    <tib~ <lm>   1980  1061              71    2681245
##  2 Albania Europe    <tib~ <lm>   1981  1100              72    2735329
##  3 Albania Europe    <tib~ <lm>   1982  1111              72    2788315
##  4 Albania Europe    <tib~ <lm>   1983  1101              72    2842620
##  5 Albania Europe    <tib~ <lm>   1984  1065              72    2901590
##  6 Albania Europe    <tib~ <lm>   1985  1060              73    2966799
##  7 Albania Europe    <tib~ <lm>   1986  1092              73    3041003
##  8 Albania Europe    <tib~ <lm>   1987  1054              73    3121336
##  9 Albania Europe    <tib~ <lm>   1988  1014              73    3197064
## 10 Albania Europe    <tib~ <lm>   1989  1092              73    3253659
## # ... with 7,672 more rows, and 2 more variables: scale_year[,1] <dbl>,
## #   resid <dbl>
```

```
resids %>%
  ggplot(aes(year,resid)) +
  geom_line(aes(group = country, color = continent), alpha = 1/3) +
  geom_smooth(se = FALSE)
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
#glance1 <- by_country %>%
#  mutate(glance = map(model, glance)) %>%
#  unnest(glance, .drop = TRUE)
```