Lecture7_note_data_visualization

Shukyee Chan

2023-10-26

1. Data Preparation

```
library(tidyverse)

d_full <- bind_rows(
  read_csv("_DataPublic_/vdem/1789_1827/vdem_1789_1827_external.csv"),
  read_csv("_DataPublic_/vdem/1867_1905/vdem_1867_1905_external.csv"),
  read_csv("_DataPublic_/vdem/1906_1944/vdem_1906_1944_external.csv"),
  read_csv("_DataPublic_/vdem/1945_1983/vdem_1945_1983_external.csv"),
  read_csv("_DataPublic_/vdem/1984_2022/vdem_1984_2022_external.csv")
)
```

Select Indicators of Interest

```
d <- d_full |>
  select(
    country_text_id, year,
    e_regiongeo, e_pelifeex, e_gdppc,
    e_mipopula, e_wb_pop, e_peinfmor,
    e_boix_regime, e_lexical_index, e_p_polity) |>
  rename("region" = "e_regiongeo",
         "life_expectancy" = "e_pelifeex",
         "gdppc" = "e_gdppc",
         "population_ClioInfra" = "e_mipopula",
         "population_WorldBank" = "e_wb_pop",
         "infant_mortality" = "e_peinfmor",
         "democracy_binary" = "e_boix_regime",
         "democracy_lexical" = "e_lexical_index",
         "democracy_polity5" = "e_p_polity") |>
  filter(year >= 1800)

saveRDS(d, "Lecture_07/data/wealth_and_health.rds")
```

```
summary(d)
```

```
##  country_text_id         year          region      life_expectancy
##  Length:23593       Min.   :1800   Min.   : 1.00   Min.   : 1.50
```

```
##  Class :character   1st Qu.:1912   1st Qu.: 5.00   1st Qu.:35.50
##  Mode  :character   Median :1952   Median : 9.00   Median :50.30
##                     Mean   :1944   Mean   : 9.46   Mean   :51.37
##                     3rd Qu.:1989   3rd Qu.:14.00   3rd Qu.:67.10
##                     Max.   :2022   Max.   :19.00   Max.   :85.30
##                                                    NA's   :1232
##      gdppc          population_ClioInfra population_WorldBank infant_mortality
##  Min.   :  0.286   Min.   :     17.9    Min.   :4.170e+04    Min.   :  1.47
##  1st Qu.:  1.599   1st Qu.:   1021.9    1st Qu.:2.348e+06    1st Qu.: 53.70
##  Median :  2.774   Median :   3522.3    Median :7.144e+06    Median :200.00
##  Mean   :  7.194   Mean   :  18688.0    Mean   :3.239e+07    Mean   :217.20
##  3rd Qu.:  7.606   3rd Qu.:   9718.8    3rd Qu.:2.103e+07    3rd Qu.:380.00
##  Max.   :156.628   Max.   :1262645.0    Max.   :1.412e+09    Max.   :756.00
##  NA's   :4571      NA's   :7173         NA's   :13583        NA's   :1232
##  democracy_binary democracy_lexical democracy_polity5
##  Min.   :0.000    Min.   :0.000     Min.   :-88.000
##  1st Qu.:0.000    1st Qu.:0.000     1st Qu.: -7.000
##  Median :0.000    Median :2.000     Median : -3.000
##  Mean   :0.364    Mean   :2.338     Mean   : -3.616
##  3rd Qu.:1.000    3rd Qu.:4.000     3rd Qu.:  7.000
##  Max.   :1.000    Max.   :6.000     Max.   : 10.000
##  NA's   :7623     NA's   :675       NA's   :8195
```

Multiple Population Data Sources

```
# Check years that are available in both datasets
d_pop_overlap <- d |> select(country_text_id, year, starts_with("population_")) |>
  drop_na()
print(d_pop_overlap, n = 3)
```

```
## # A tibble: 5,818 x 4
##   country_text_id  year population_ClioInfra population_WorldBank
##   <chr>           <dbl>                <dbl>                <dbl>
## 1 MEX              1960               38578.             37771861
## 2 MEX              1961               39998.             38966049
## 3 MEX              1962               41418.             40195318
## # ... with 5,815 more rows
```

```
unique(d_pop_overlap$year)
```

```
##  [1] 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974
## [16] 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989
## [31] 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000
```

```
cor(d_pop_overlap$population_ClioInfra, d_pop_overlap$population_WorldBank)
```

```
## [1] 0.9997128
```

Set a Rule to Merge the Two Population Columns

- Different units: Divide `population_WorldBank` by 1000 (so that the unit of population is "in thousands"
- Different coverage but almost perfect correlation
  - For years that only one dataset has coverage, take the value from the dataset that has available data points.
  - For years that both datasets have coverage, take their `mean`.
  - --> taking the `mean` and allow `na.rm = TRUE`.

Merge the Two Population Columns

```
# STEP 1: "Harmonize" the units
d <- d |> mutate(population_WorldBank = population_WorldBank / 1000)

# STEP 2 Method 1: Slower but use only tidyverse functionality
# [Slow! Not recommended!]
d <- d |> rowwise() |>
  mutate(population = mean(c_across(c("population_ClioInfra", "population_WorldBank")),
                           na.rm = TRUE), .after = population_WorldBank) |>
  ungroup()

# STEP 2 Method 2: Faster but use a non-tidyverse function rowMeans()
# and create a temporary vector tmp_population, which I remove after use with rm()
# [Faster !Recommended!]
tmp_population <- d |> select(population_ClioInfra, population_WorldBank) |> rowMeans(na.rm = TRUE)
d <- d |> mutate(population = !!(tmp_population), .after = population_WorldBank)

rm(tmp_population)

# Remove the columns we no longer need
d <- d |> select(-population_ClioInfra, -population_WorldBank)
```

Sanity Check

```
summary(d %>% select(-country_text_id, -year, -region))
```

```
##  life_expectancy     gdppc           population        infant_mortality
##  Min.   : 1.50   Min.   :  0.286   Min.   :     17.9   Min.   :  1.47
##  1st Qu.:35.50   1st Qu.:  1.599   1st Qu.:   1246.3   1st Qu.: 53.70
##  Median :50.30   Median :  2.774   Median :   4234.3   Median :200.00
##  Mean   :51.37   Mean   :  7.194   Mean   :  23083.3   Mean   :217.20
##  3rd Qu.:67.10   3rd Qu.:  7.606   3rd Qu.:  11914.2   3rd Qu.:380.00
##  Max.   :85.30   Max.   :156.628   Max.   :1412360.0   Max.   :756.00
##  NA's   :1232    NA's   :4571      NA's   :2981        NA's   :1232
##  democracy_binary democracy_lexical democracy_polity5
##  Min.   :0.000    Min.   :0.000     Min.   :-88.000
```

```
##  1st Qu.:0.000    1st Qu.:0.000     1st Qu.: -7.000
##  Median :0.000    Median :2.000     Median : -3.000
##  Mean   :0.364    Mean   :2.338     Mean   : -3.616
##  3rd Qu.:1.000    3rd Qu.:4.000     3rd Qu.:  7.000
##  Max.   :1.000    Max.   :6.000     Max.   : 10.000
##  NA's   :7623     NA's   :675       NA's   :8195
```

Check Data Availability

```
check_data_available <- d |>
  mutate(Available = (!is.na(life_expectancy) & !is.na(gdppc) & !is.na(population)))
# Check number of missing values by country-year
table(check_data_available$Available, useNA = "always")
```

```
##
## FALSE  TRUE  <NA>
##  6003 17590     0
```

```
check_data_available |> print(n = 3)
```

```
## # A tibble: 23,593 x 11
##   country_t~1  year region life_~2 gdppc popul~3 infan~4 democ~5 democ~6 democ~7
##   <chr>       <dbl>  <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 MEX          1800     17    26.9  1.35   5100      487      NA       0      NA
## 2 MEX          1801     17    26.9  1.34   5174.     487      NA       0      NA
## 3 MEX          1802     17    26.9  1.32   5249.     487      NA       0      NA
## # ... with 23,590 more rows, 1 more variable: Available <lgl>, and abbreviated
## #   variable names 1: country_text_id, 2: life_expectancy, 3: population,
## #   4: infant_mortality, 5: democracy_binary, 6: democracy_lexical,
## #   7: democracy_polity5
```

```
check_data_available_wide <- check_data_available |>
  select(country_text_id, year, Available) |>
  pivot_wider(names_from = "country_text_id", values_from = "Available",
              names_prefix = "c_") |>
  arrange(year)

check_data_available_wide |> print(n = 3)
```

```
## # A tibble: 184 x 203
##    year c_MEX c_SWE c_CHE c_JPN c_MMR c_RUS c_EGY c_YEM c_COL c_POL c_BRA c_USA
##   <dbl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl>
## 1  1800 TRUE  TRUE  TRUE  TRUE  FALSE FALSE TRUE  FALSE TRUE  NA    TRUE  FALSE
## 2  1801 TRUE  TRUE  TRUE  TRUE  FALSE FALSE TRUE  FALSE TRUE  NA    TRUE  FALSE
## 3  1802 TRUE  TRUE  TRUE  TRUE  FALSE FALSE TRUE  FALSE TRUE  NA    TRUE  FALSE
## # ... with 181 more rows, and 190 more variables: c_PRT <lgl>, c_BOL <lgl>,
## #   c_HTI <lgl>, c_PER <lgl>, c_VDR <lgl>, c_AFG <lgl>, c_ARG <lgl>,
## #   c_ETH <lgl>, c_IND <lgl>, c_KOR <lgl>, c_THA <lgl>, c_VEN <lgl>,
## #   c_IDN <lgl>, c_NPL <lgl>, c_AUS <lgl>, c_CHL <lgl>, c_FRA <lgl>,
## #   c_DEU <lgl>, c_GTM <lgl>, c_IRN <lgl>, c_LBR <lgl>, c_MAR <lgl>,
## #   c_NLD <lgl>, c_ESP <lgl>, c_TUN <lgl>, c_TUR <lgl>, c_GBR <lgl>,
## #   c_URY <lgl>, c_CHN <lgl>, c_DOM <lgl>, c_LBY <lgl>, c_MDG <lgl>, ...
```

```r
# Check, for each year, the availability of each column
check_data_available_by_column <- d |>
  group_by(year) |>
  summarise(
    life_expectancy = sum(is.na(life_expectancy)),
    gdppc = sum(is.na(gdppc)),
    population = sum(is.na(population))
  )
  # summarise_at(vars(life_expectancy, gdppc, population), ~sum(!is.na(.)))
  # above is an alternative way to write the summarise() step

check_data_available_by_column |> print(n = 3)
```

```
## # A tibble: 184 x 4
##    year life_expectancy gdppc population
##   <dbl>           <int> <int>      <int>
## 1  1800              16    31         25
## 2  1801              16    31         25
## 3  1802              17    32         26
## # ... with 181 more rows
```

Save Cleaned Data

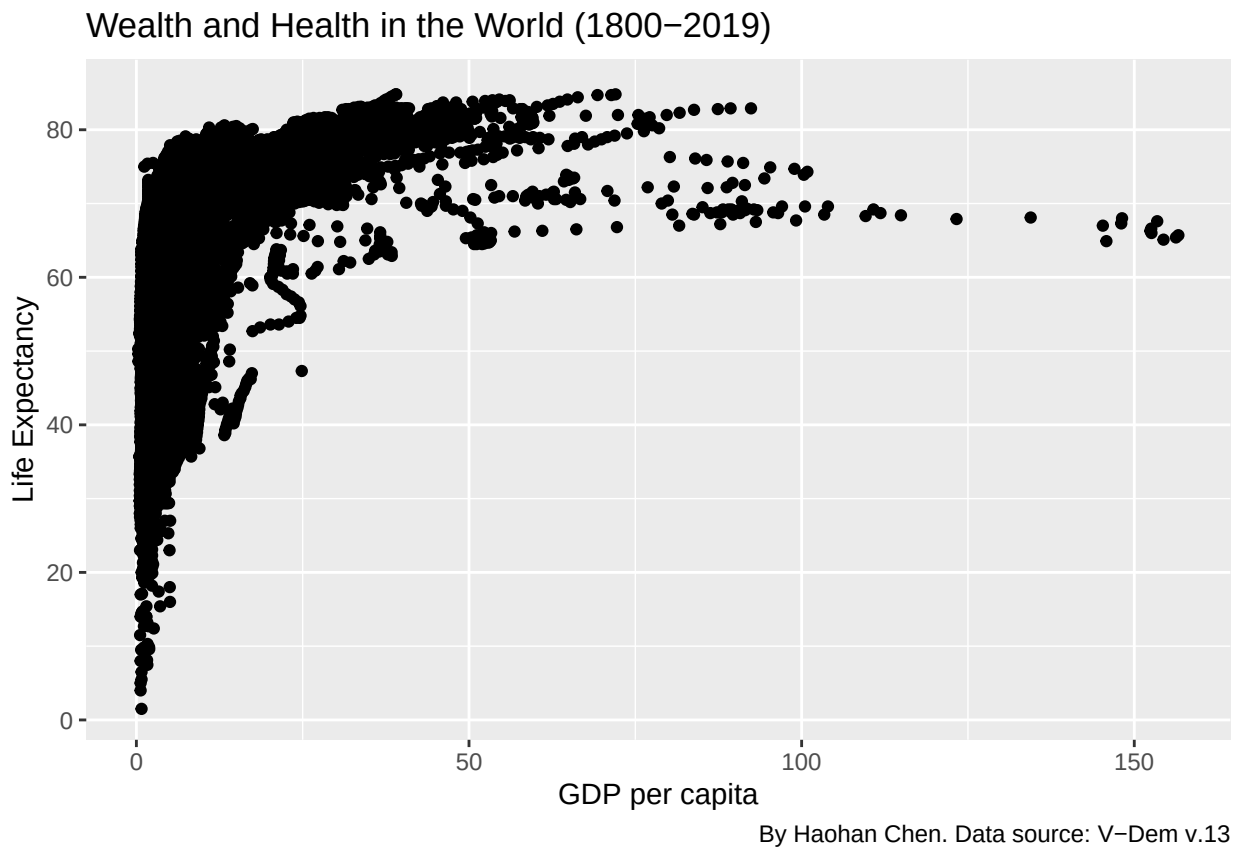
```r
dir.create("Lecture_07/data")
```

```r
saveRDS(d, "Lecture_07/data/wealth_and_health.rds")
```

2. Data Viz Basics

Simplest Possible Visualization

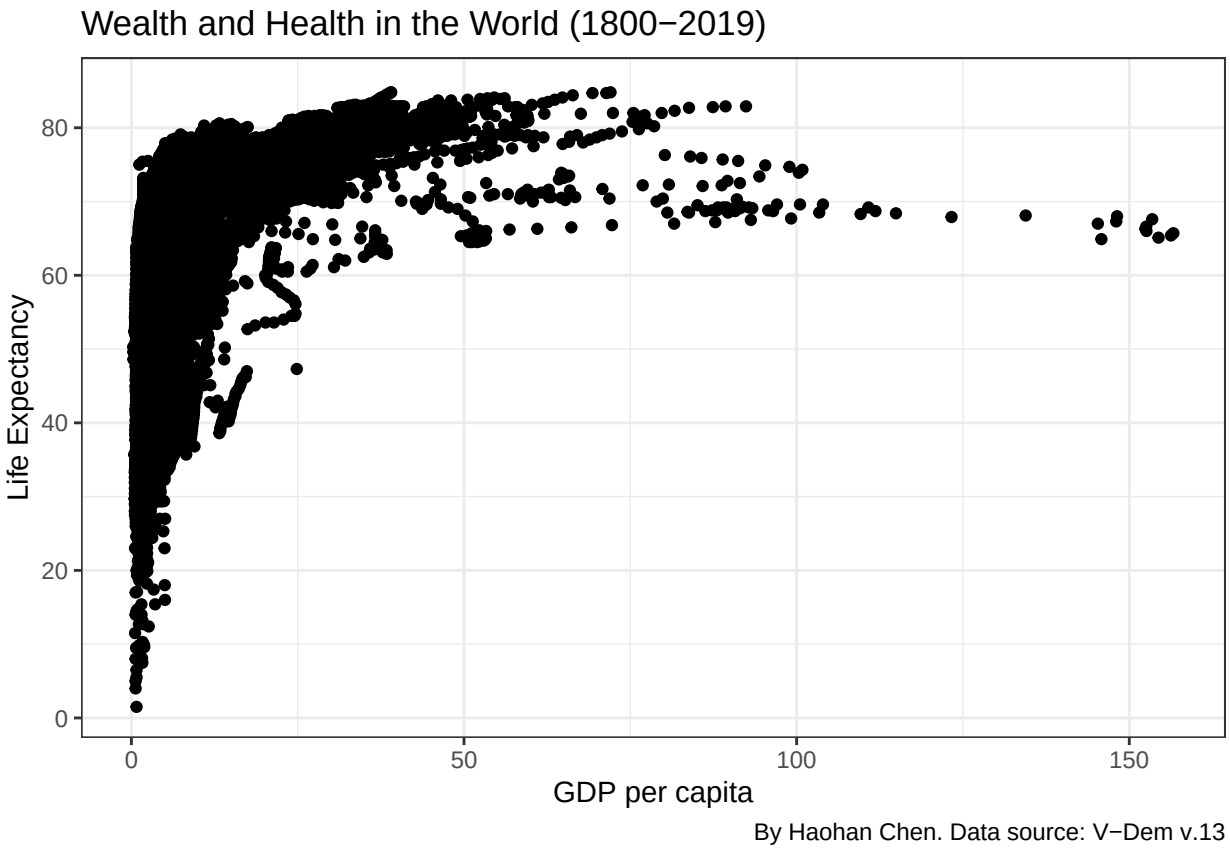
```r
d |>
  ggplot(aes(x = gdppc, y = life_expectancy)) +
  geom_point() +
  labs(x = "GDP per capita", y = "Life Expectancy",
       title = "Wealth and Health in the World (1800-2019)",
       caption = "By Haohan Chen. Data source: V-Dem v.13")
```




```
# Store in R environment (temporary)
p_all <- d |>
  ggplot(aes(x = gdppc, y = life_expectancy)) +
  geom_point() +
  labs(x = "GDP per capita", y = "Life Expectancy",
       title = "Wealth and Health in the World (1800-2019)",
       caption = "By Haohan Chen. Data source: V-Dem v.13")
# Save plot as a .rds file in your folder
dir.create("Lecture_07/1_data_visualization_1/figures")
saveRDS(p_all, "Lecture_07/1_data_visualization_1/figures/welath_and_health_all.rds")
# Save plot as a PDF file in your folder
ggsave(filename = "Lecture_07/1_data_visualization_1/figures/welath_and_health_all.pdf",
       plot = p_all, width = 9, height = 4)
```

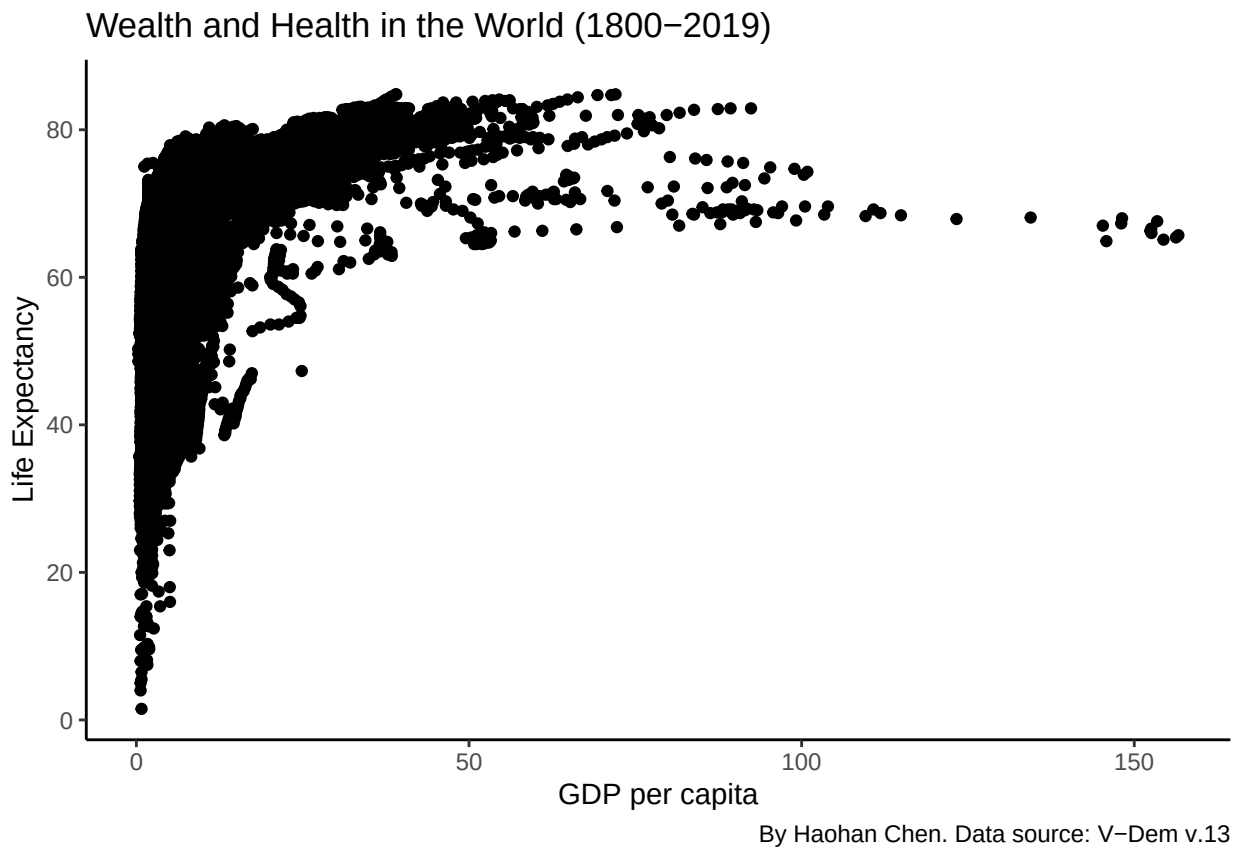
Set Themes: `theme_bw`

```
p_all + theme_bw()
```



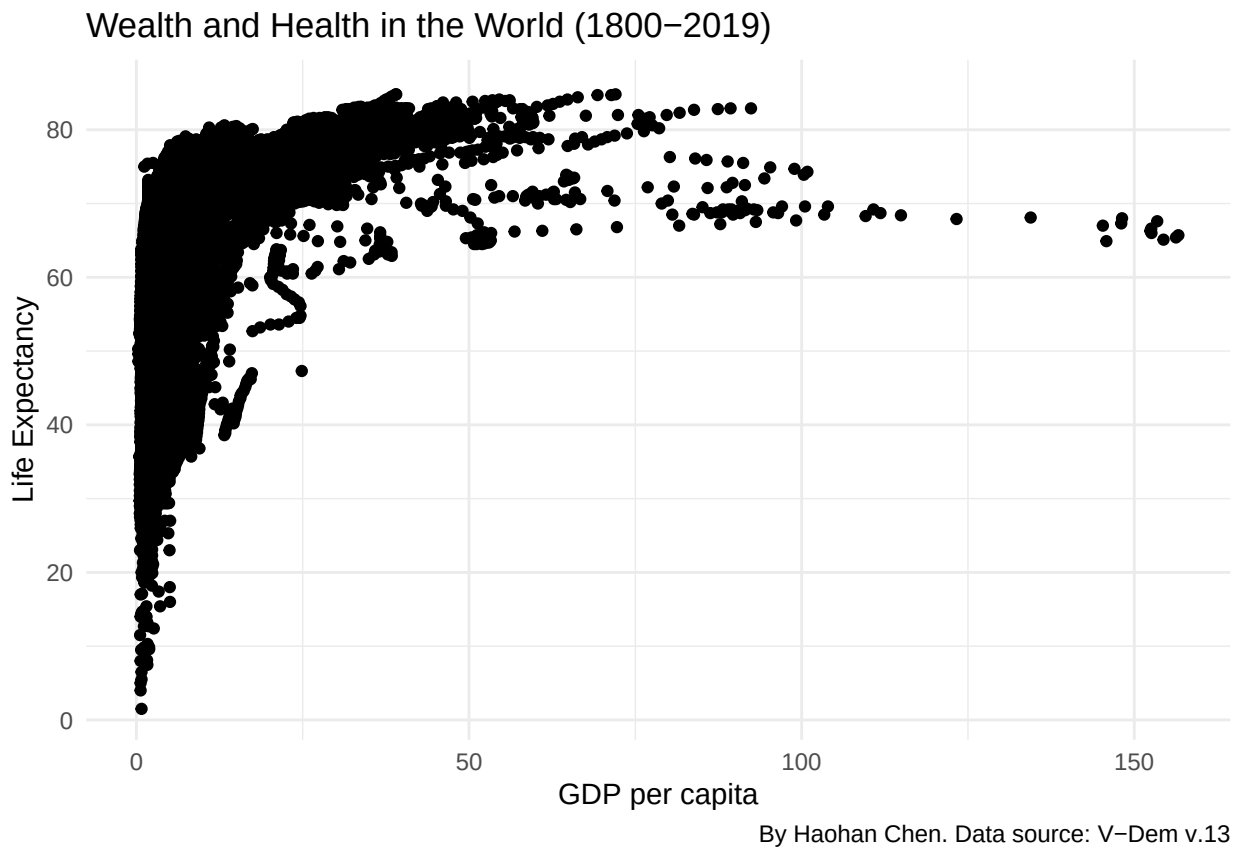
Set Themes: `theme_classic`

```
p_all + theme_classic()
```



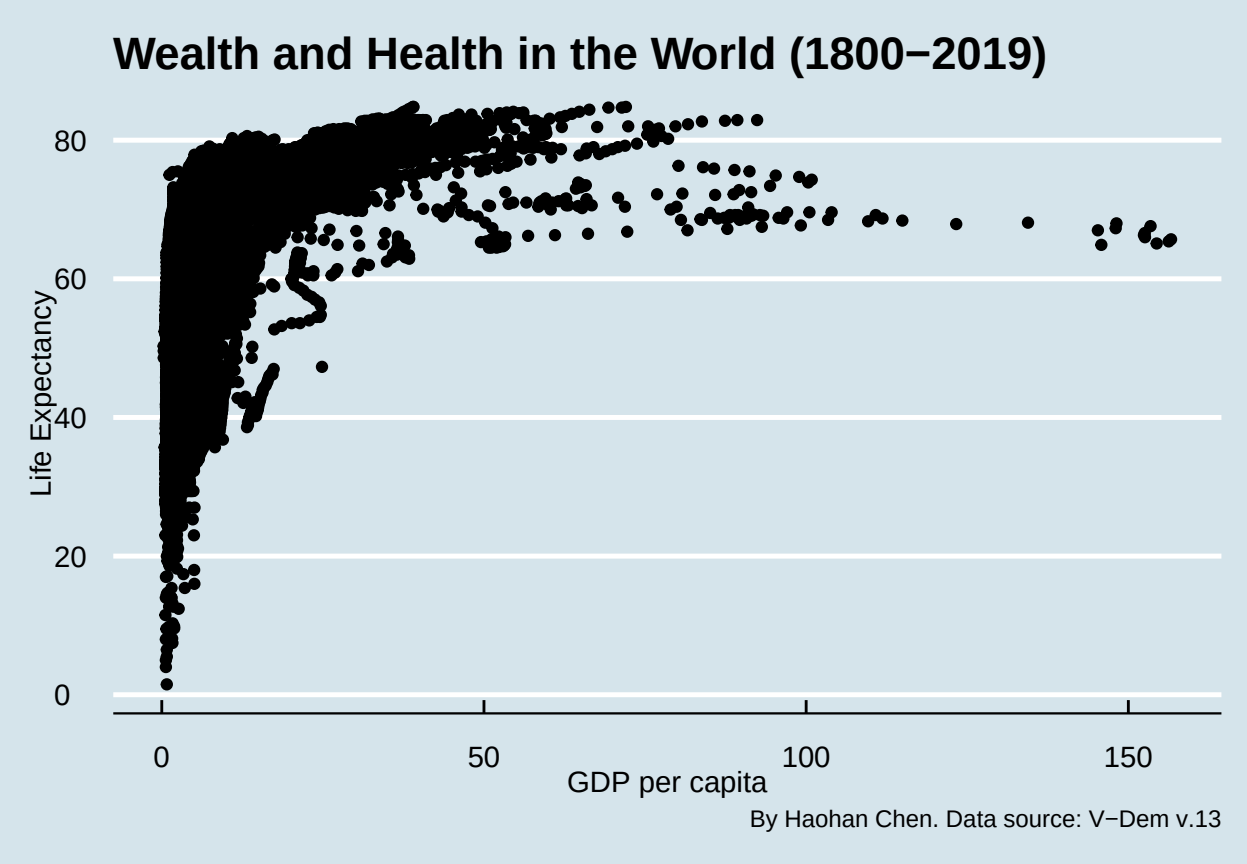
Set Themes: `theme_minimal`

```
p_all + theme_minimal()
```



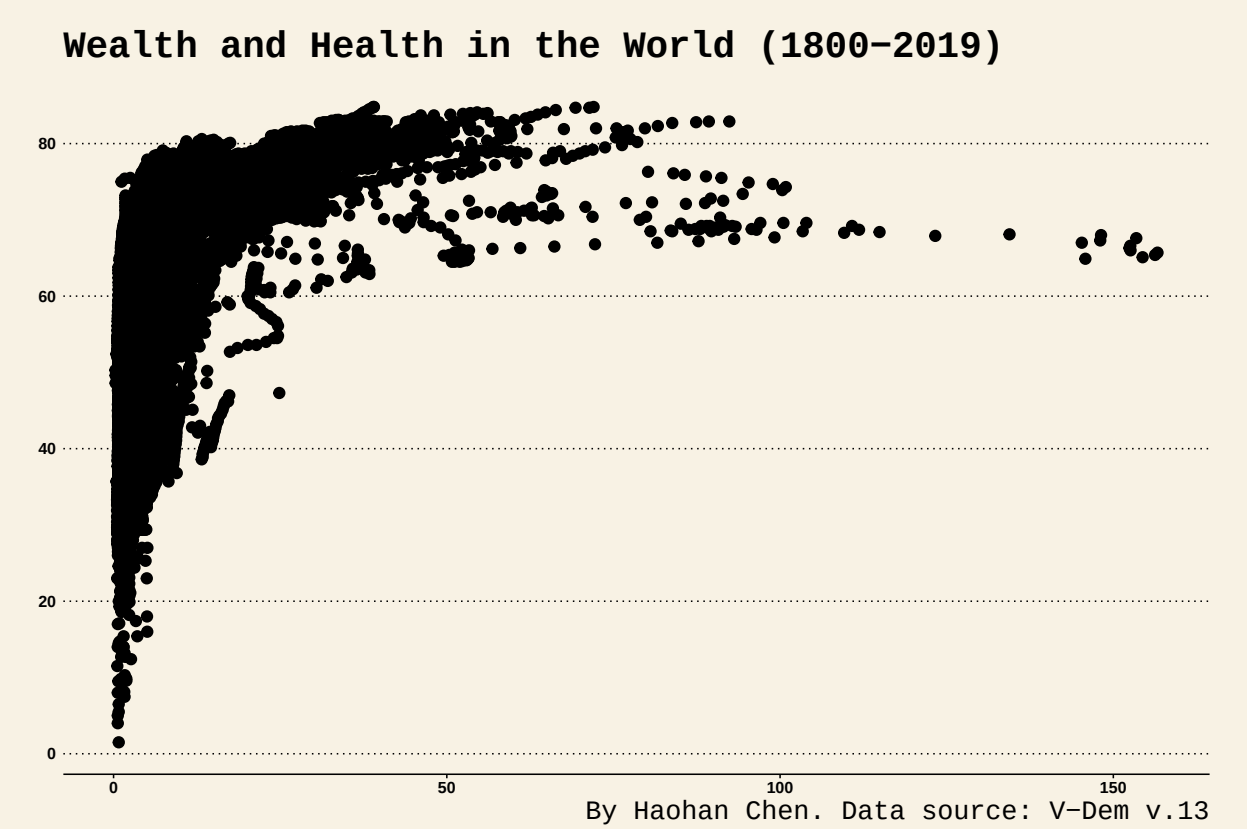
Other Fancy Themes: The Economist

```
# install.packages("ggthemes") # install the package upon your first use.
# Take a look at the package's website: https://yutannihilation.github.io/allYourFigureAreBelongToUs/gg
library(ggthemes)
p_all + theme_economist()
```



Other Fancy Themes: The WSJ

```
p_all + theme_wsj(base_size = 6)
```


Wealth and Health in the World (1800–2019)
80
60
40
20
0
0
50
100
150
By Haohan Chen. Data source: V–Dem v.13