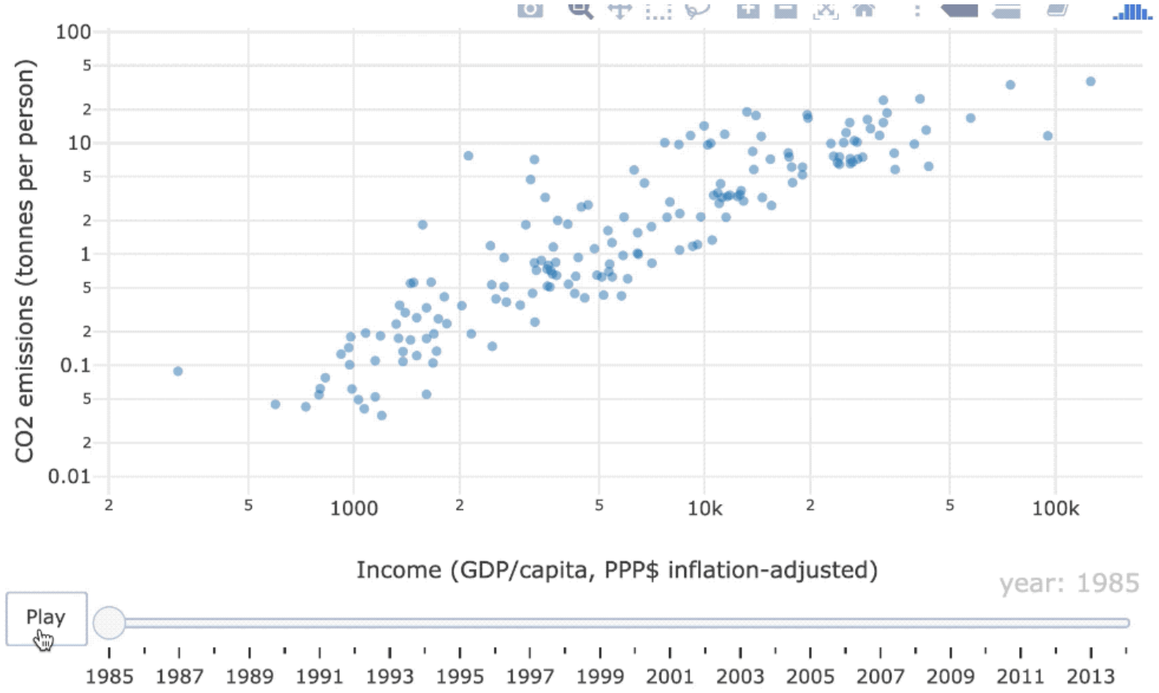
Introduction

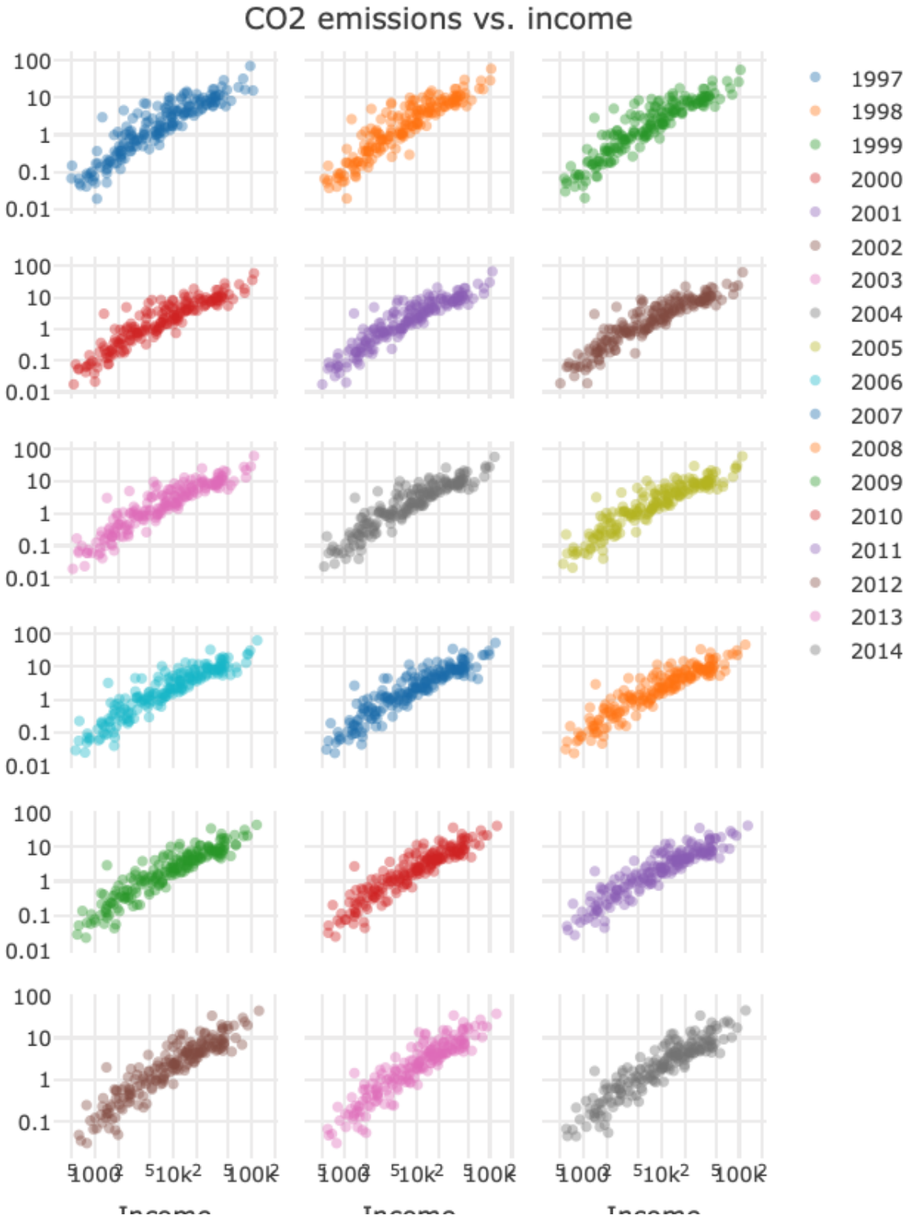
INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Adam Loy
Statistician, Carleton College

datacamp

Motivation





Is it easier to see the changes over time based on the animation? Or the faceted views?

plotly

- Visualization library for interactive and dynamic web-based graphics
- Still under active development

Types of graphics

- Static
- Interactive
- Dynamic

Static graphics

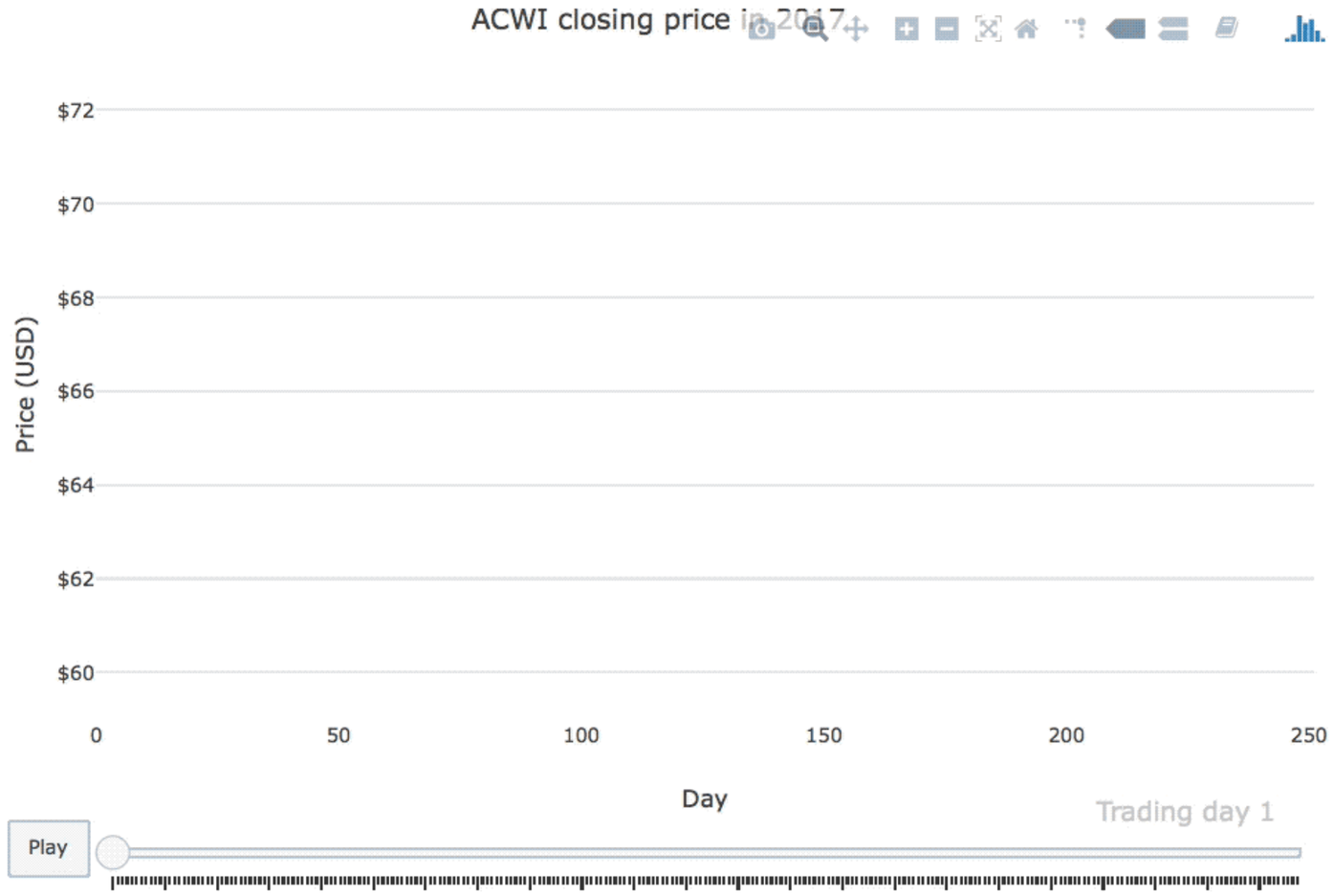
A **static graphic** is permanently fixed after it is created

Interactive graphics

An **interactive graphic** changes based on an action performed by the user

Dynamic graphics

A **dynamic graphic** changes periodically without user input



plotly review

```
msci
```

```
# A tibble: 251 x 7
   Date        Open  High   Low Close Volume Adjusted
   <date>     <dbl> <dbl> <dbl> <dbl>  <int>    <dbl>
 1 2017-01-03  79.8  79.8  78.4  78.7 646000     77.4
 2 2017-01-04  79.1  81.1  79.1  80.7 849200     79.3
 3 2017-01-05  80.4  81.8  80.4  81.6 557500     80.2
 4 2017-01-06  81.8  83.9  81.8  83.4 597800     82.0
 5 2017-01-09  83.1  83.5  82.6  82.7 668100     81.3
 6 2017-01-10  82.3  82.6  81.1  81.5 558900     80.1
 7 2017-01-11  81.2  81.6  80.8  81.5 365500     80.1
# ... with 244 more rows
```

plotly review

```
library(plotly)
msci %>%
  plot_ly(x = ~Date, y = ~Close) %>%
  add_lines()
```


72
70
68
66
64
62
60
Close
Mar 2017
May 2017
Jul 2017
Sep 2017
Nov 2017
Date

Let's practice!

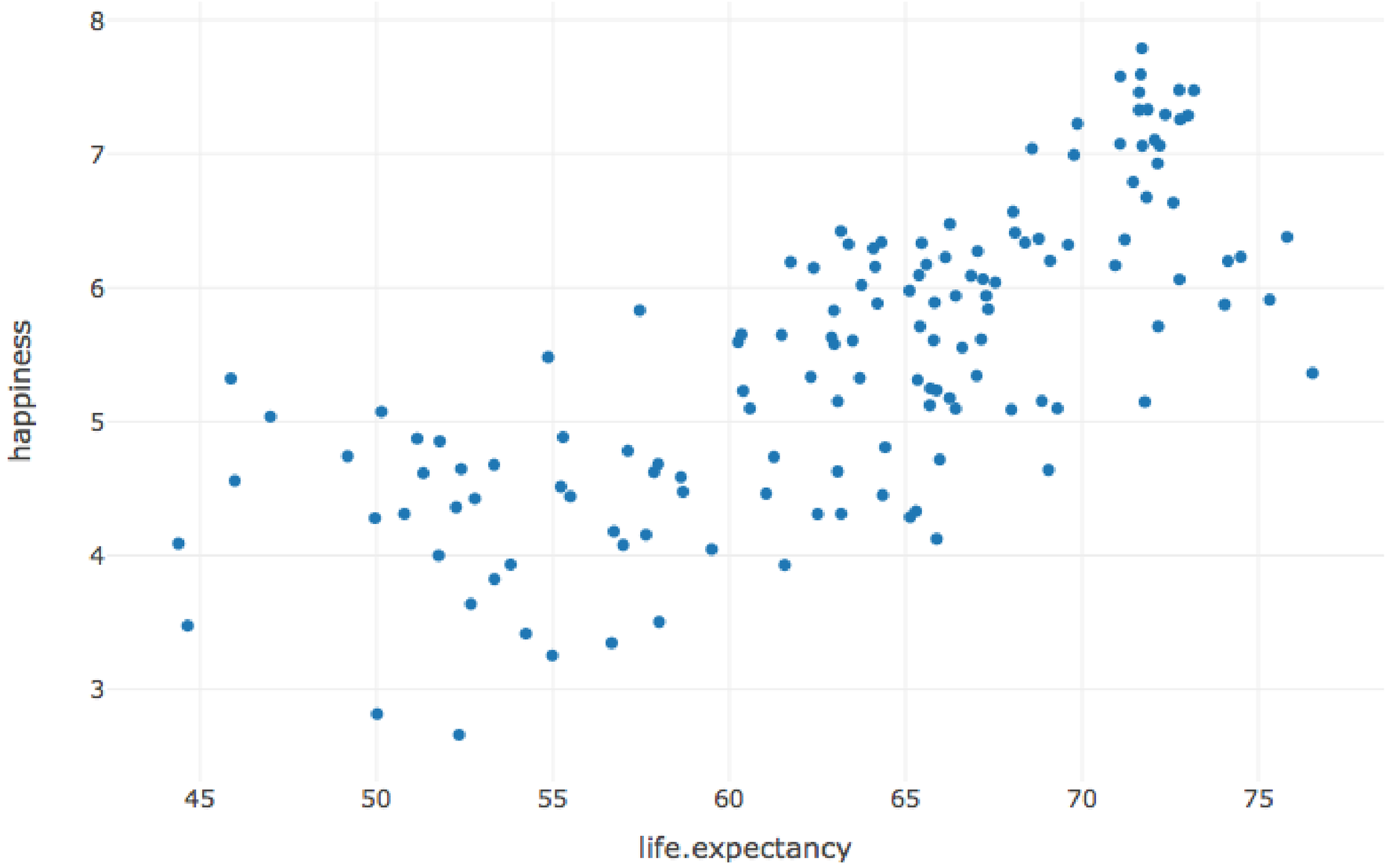
INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Utilizing color, symbol and size

INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Adam Loy
Statistician, Carleton College

datacamp


8
7
6
5
4
3
happiness
45
50
55
60
65
70
75
life.expectancy

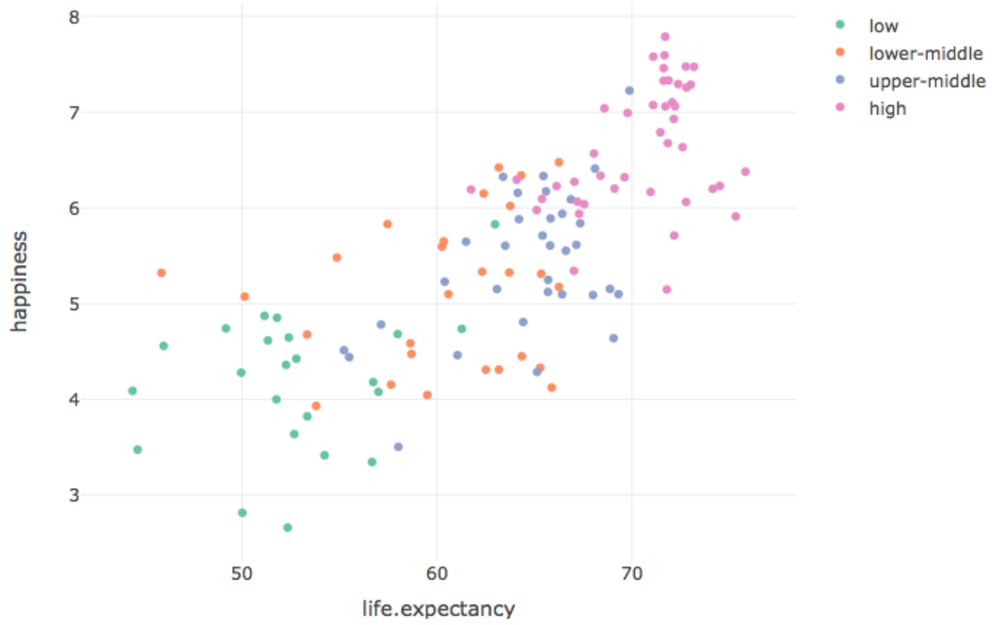
World happiness data

```
dplyr::glimpse(happy)
```

```
Observations: 141
Variables: 11
$ country         <chr> "Afghanistan", "Albania", "Algeria", ...
$ happiness       <dbl> 2.661718, 4.639548, 5.248912, 6.039330, ...
$ region          <chr> "South Asia", "Central and Eastern Europe", ...
$ population      <dbl> 35530081, 2873457, 41318142, 44271041, ...
$ log.gdp         <dbl> 7.460144, 9.373718, 9.540244, 9.843519, ...
$ income          <fct> low, upper-middle, upper-middle, high, ...
$ life.expectancy <dbl> 52.33953, 69.05166, 65.69919, 67.53870, ...
$ social.support  <dbl> 0.4908801, 0.6376983, 0.8067539, 0.9066991, ...
$ freedom         <dbl> 0.4270109, 0.7496110, 0.4366705, 0.8319662, ...
$ generosity      <dbl> -0.106340349, -0.035140377, -0.194670126, -0.18629...
$ corruption      <dbl> 0.9543926, 0.8761346, 0.6997742, 0.8410525, ...
```

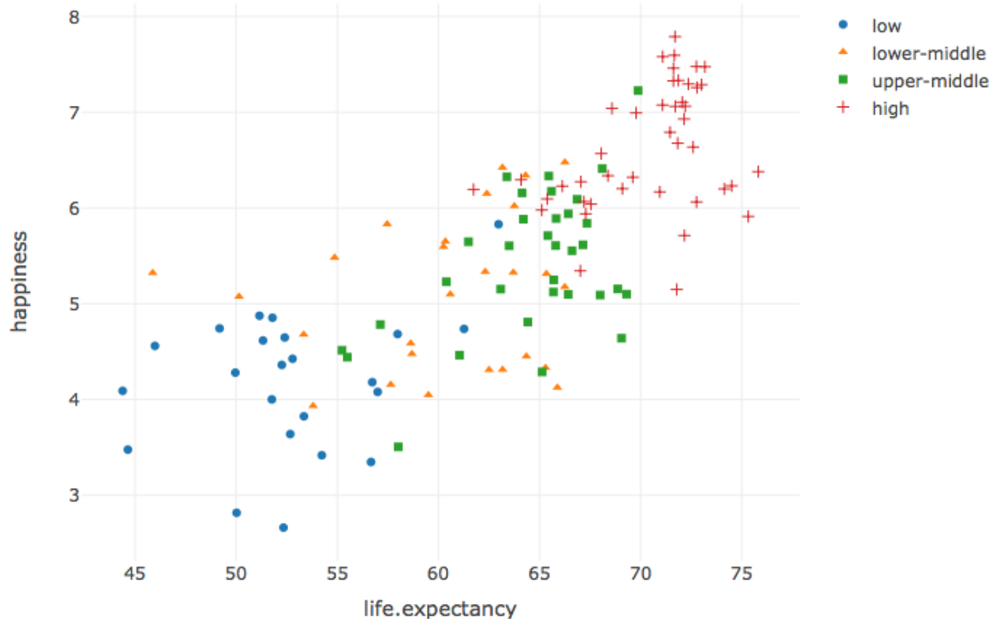
Glyph color

```
happy %>%
  plot_ly(x = ~life.expectancy, y = ~happiness) %>%
  add_markers(color = ~income)
```



Glyph symbol

```
happy %>%
  plot_ly(x = ~life.expectancy, y = ~happiness) %>%
  add_markers(symbol = ~income)
```

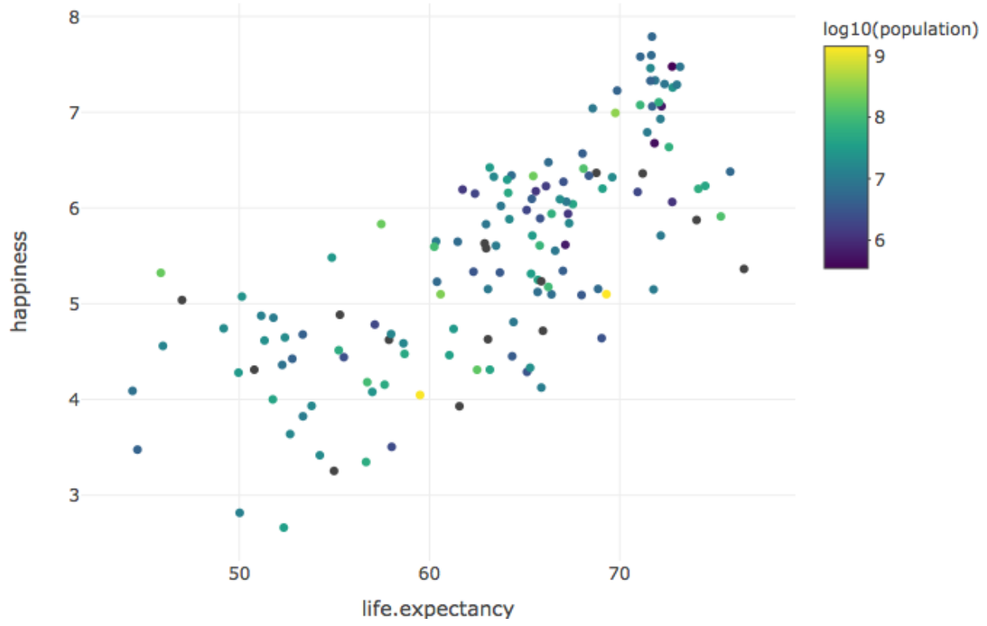


Color based on a quantitative variable

```
happy %>%
  plot_ly(x = ~life.expectancy, y = ~happiness) %>%
  add_markers(color = ~population)
```

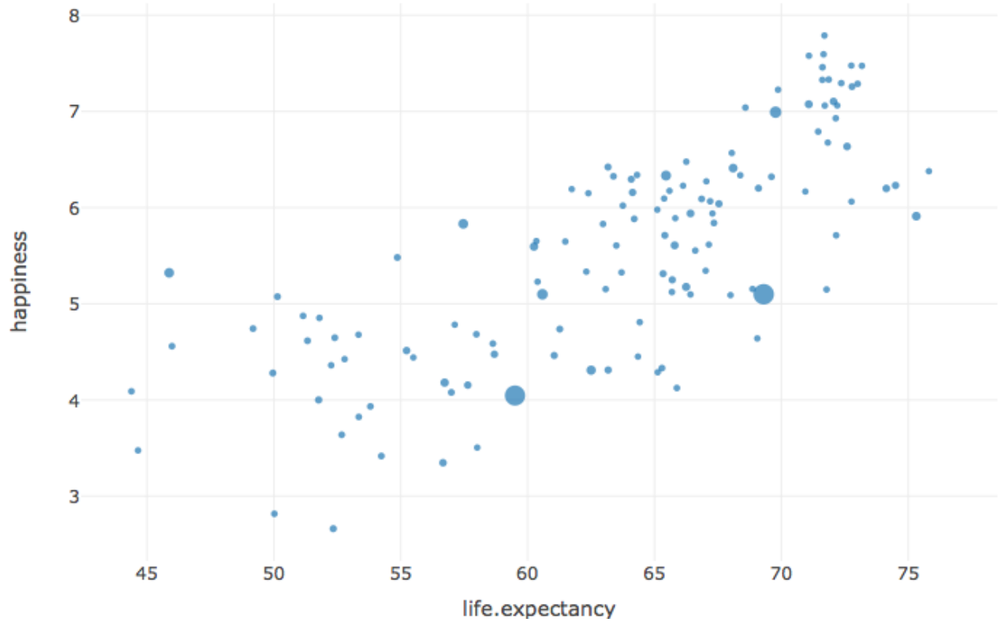
Transformations

```
happy %>%
  plot_ly(x = ~life.expectancy, y = ~happiness) %>%
  add_markers(color = ~log10(population))
```



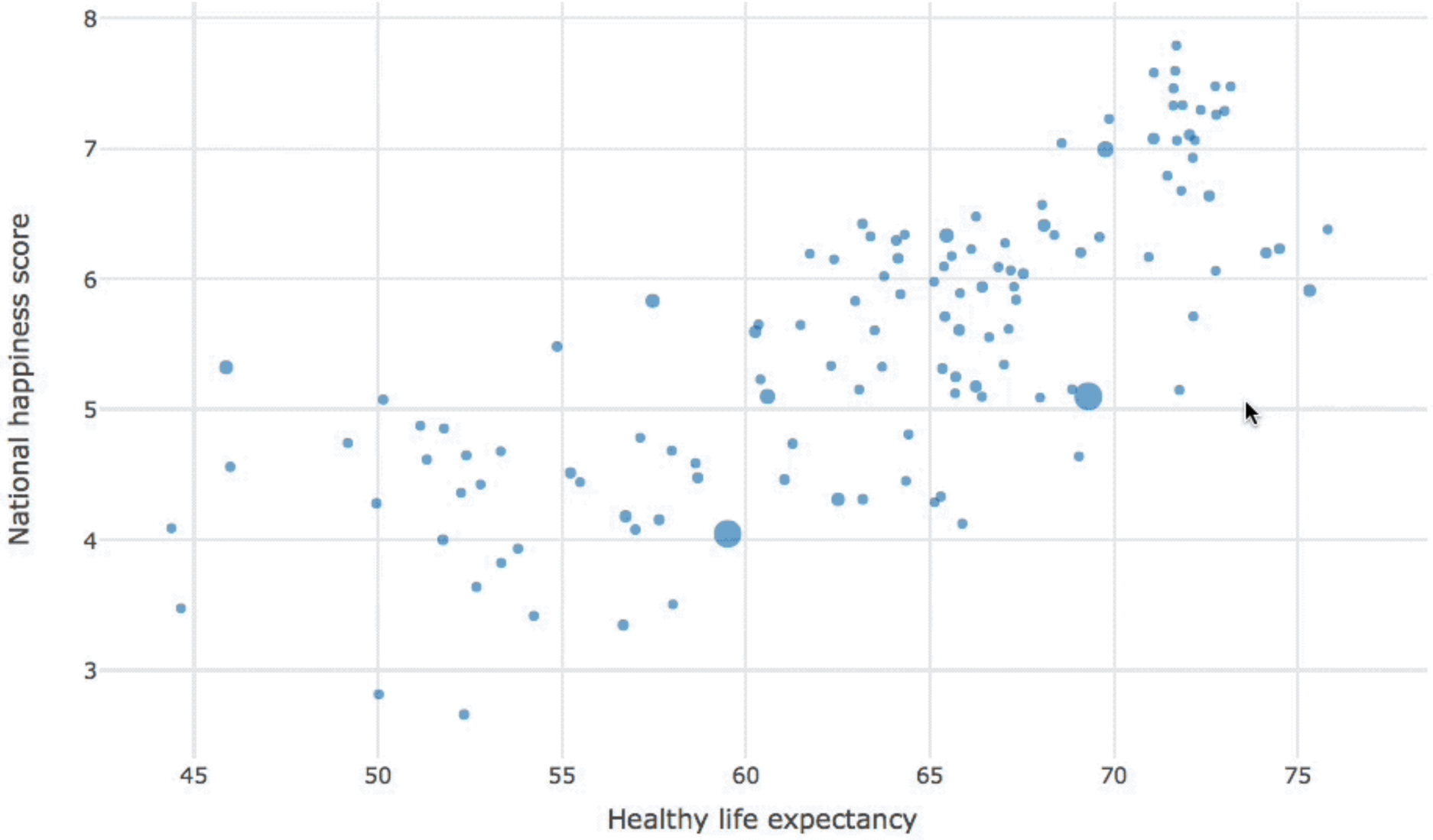
Glyph size

```
happy %>%
  plot_ly(x = ~life.expectancy, y = ~happiness) %>%
  add_markers(size = ~population)
```



Polishing labels

```
happy %>%
  plot_ly(
    x = ~life.expectancy, y = ~happiness,
    hoverinfo = "text",
    text = ~paste("Country: ", country,
                  "</br> Population: ", population)
  ) %>%
  add_markers(size = ~population) %>%
  layout(
    xaxis = list(title = "Healthy life expectancy"),
    yaxis = list(title = "National happiness score")
  )
```


National happiness score
8
7
6
5
4
3
45
50
55
60
65
70
75
Healthy life expectancy

Let's practice!

INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Moving Beyond Simple Interactivity

INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Adam Loy
Statistician, Carleton College

datacamp

Country-level economic indicators

Source: gapminder.org

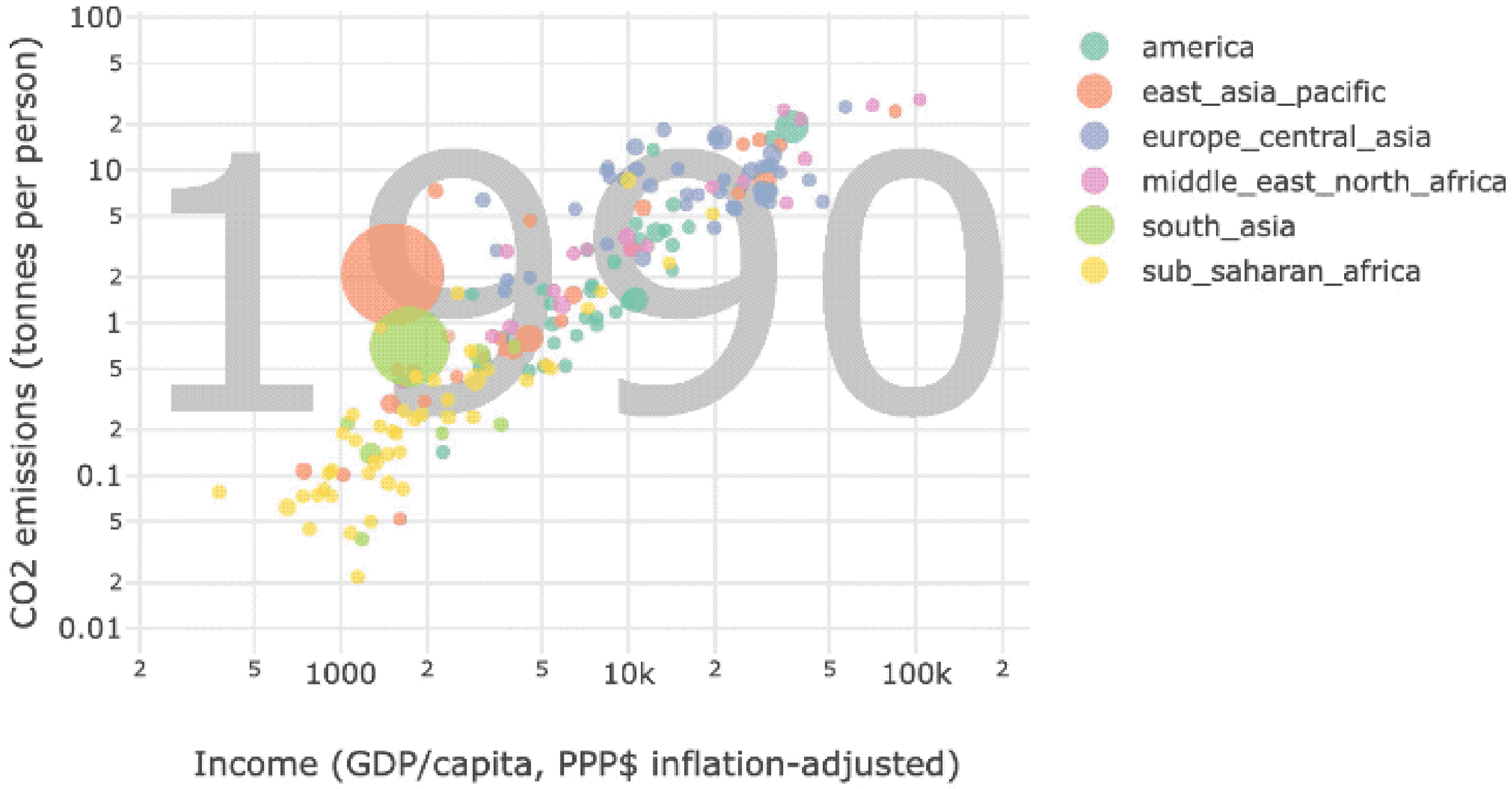
```
world_indicators
```

```
# A tibble: 11,387 x 11
  country  year income     co2 military population  urban life_expectancy four_regions
  <chr>   <dbl>  <dbl>   <dbl>    <dbl>      <dbl>  <dbl>           <dbl> <chr>
1 Afghan…  1960   1210  0.0461       NA    9000000 7.56e5            38.6 asia
2 Albania  1960   2790  1.24         NA    1640000 4.94e5            62.7 europe
3 Algeria  1960   6520  0.554        NA   11100000 3.39e6            52   africa
4 Andorra  1960  15200 NA            NA      13400 7.84e3            NA   europe
5 Angola   1960   3860  0.0975       NA    5640000 5.89e5            42.4 africa
# … with 1.138e%2004 more rows, and 2 more variables: eight_regions <chr>, six_regions <chr>
```

State-level economic data

```
us_economy
```

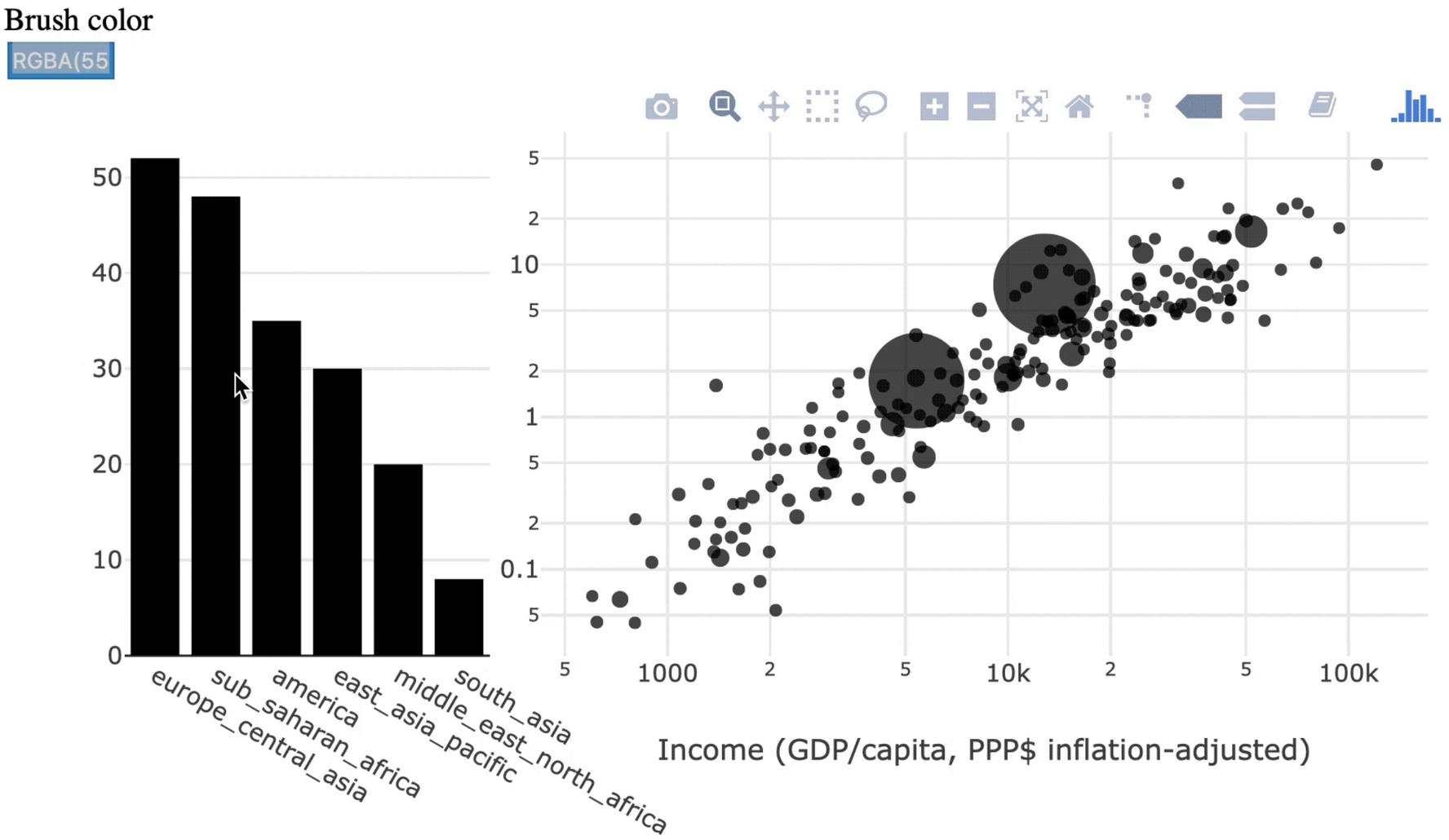
```
# A tibble: 1,071 x 9
  state  year    gdp employment home_owners house_price population region division
  <chr> <dbl>  <dbl>      <dbl>       <dbl>       <dbl>      <dbl> <chr>  <chr>
1 AK     1997 42262.         NA        67.2        159.       609. West   Pacific
2 AK     1998 41157.         NA        66.3        164.       615. West   Pacific
3 AK     1999 40722.         NA        66.4        169.       620. West   Pacific
4 AK     2000 39517.         NA        66.4        172.       628. West   Pacific
5 AK     2001 40974.         NA        65.3        181.       634. West   Pacific
# … with 1,066 more rows
```


1990
CO2 emissions (tonnes per person)
100
5
2
10
5
2
1
5
2
0.1
5
2
0.01
2
5
1000
2
5
10k
2
5
100k
2
Income (GDP/capita, PPP$ inflation-adjusted)
america
east_asia_pacific
europe_central_asia
middle_east_north_africa
south_asia
sub_saharan_africa

Static bubble charts

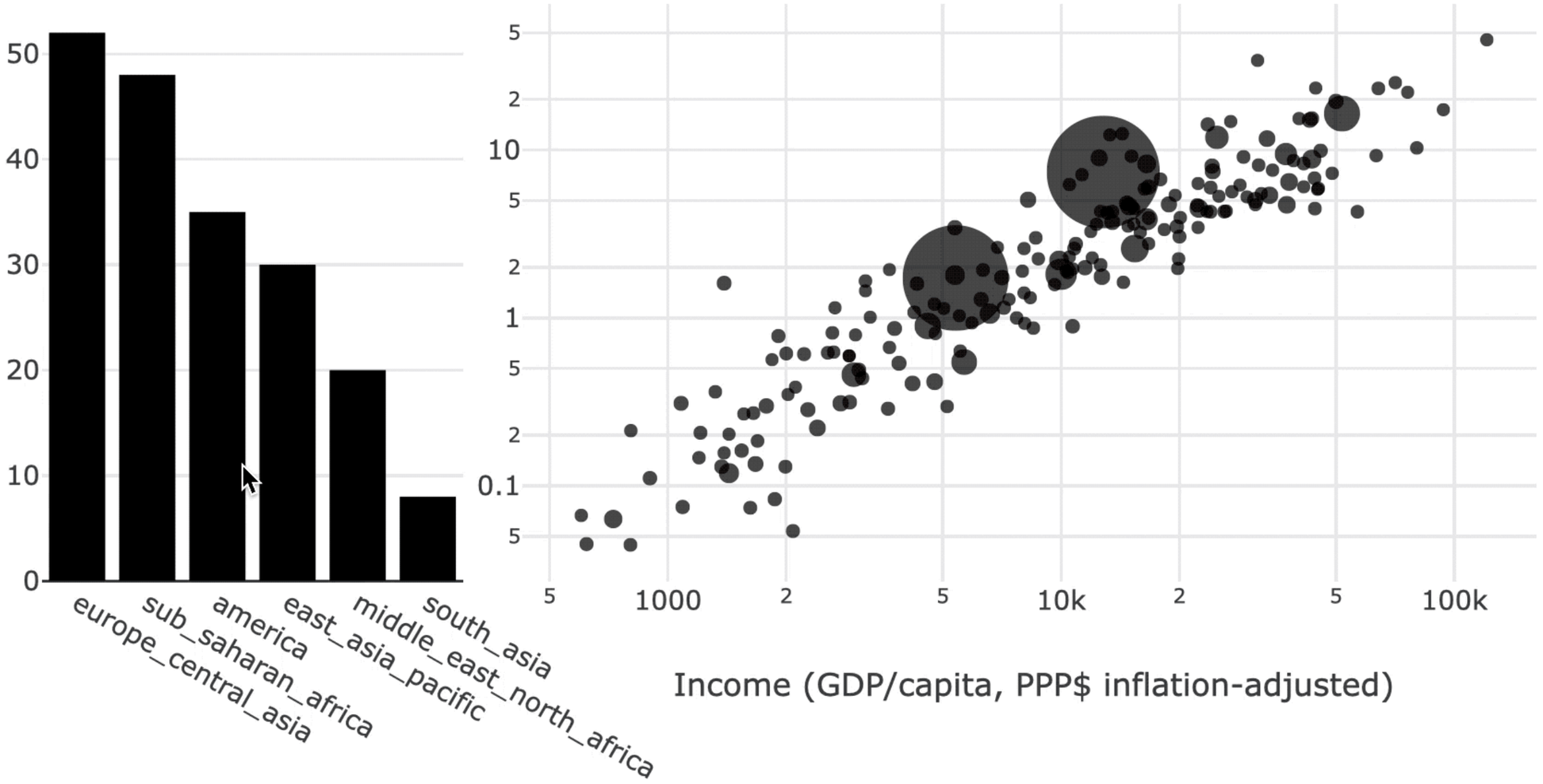
```
world_indicators %>%
   filter(year == 2014) %>%
   plot_ly(
     x = ~income, y = ~co2, hoverinfo = "text",
     text = ~country
   ) %>%
   add_markers(
      size = ~population, color = ~six_regions,
      marker = list(opacity = 0.5,
                    sizemode = "diameter",
                    sizeref = 2)
   )
```

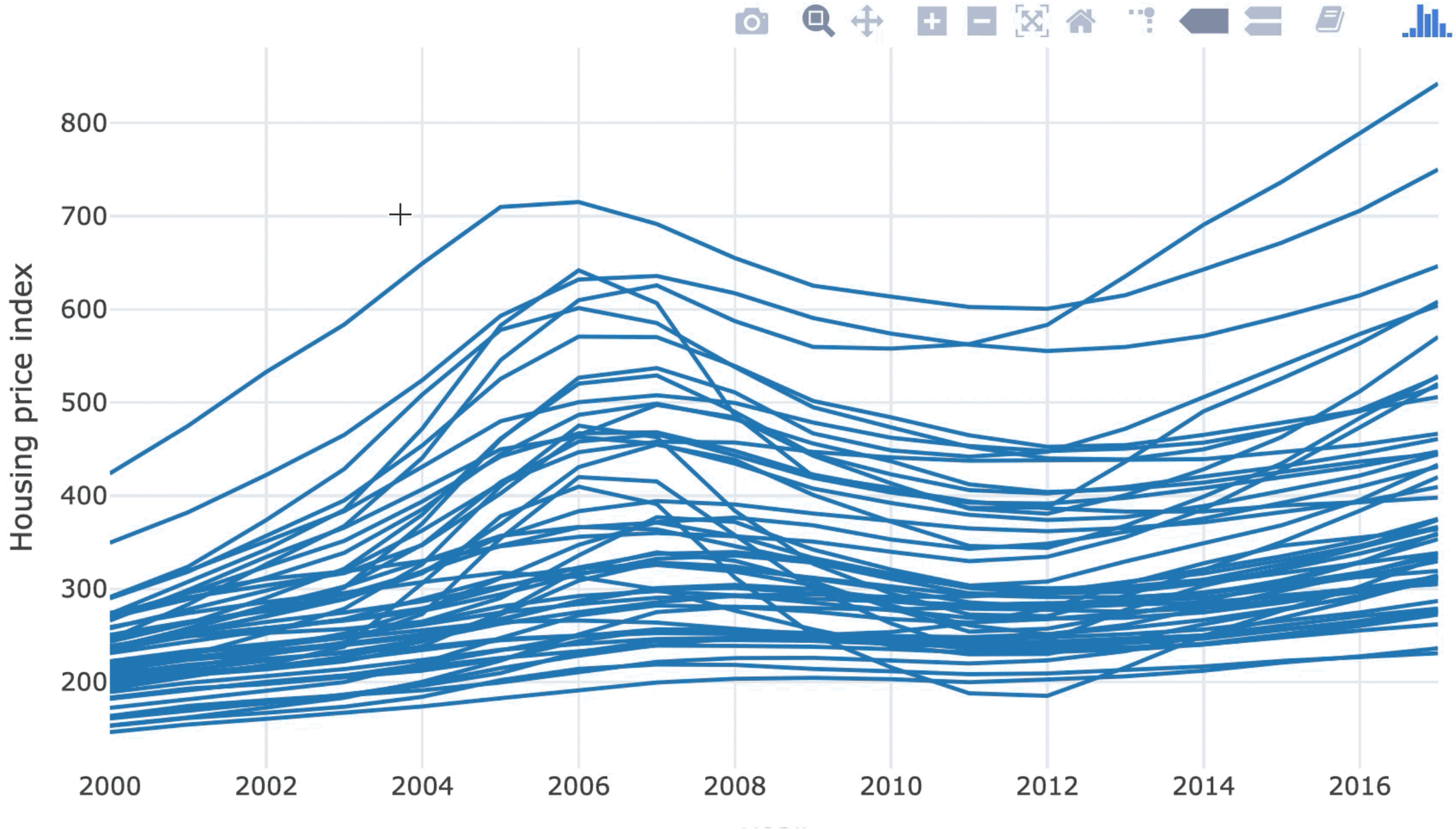
Linked brushing



Brush color

rgba(228




Housing price index
800
700
600
500
400
300
200
2000
2002
2004
2006
2008
2010
2012
2014
2016

Let's explore!

INTERMEDIATE INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R