Happy Planet Index clustering script

HarvardX Data Science Capstone Own Project

Codrin Kruijne

2019-06-04

```
# PLEASE NOTE: SCRIPT SHOULD RUN ALL NECESSARY DATA, BUT HAS ONLY BEEN TESTED ON WINDOWS
# IF NEEDED, PLEASE DOWNLOAD THE TWO REQUIRED FILES IN a "Data" SUBDIR OF YOUR WORKING DIRECTORY
# HPI DATA https://www.dropbox.com/s/cqtynj47altwo3d/hpi-data-2016.xlsx?dl=0
# SPI DATA https://www.dropbox.com/s/hb2e3h5l69n0vrx/Social%20Progress%20Index%202018-Results.xlsx?dl=0
# YOU CAN UNCOMMENT CODE BELOW TO READ IN THE LOCAL FILES

# Script settings
knitr::opts_chunk$set(
    message = FALSE,
    warning = FALSE,
    cache = TRUE,
    tidy.opts = list(width.cutoff = 100),
    tidy = TRUE
)
script_start <- Sys.time()

# Load required packages
library(tidyverse)
library(RCurl)
library(foreign)
library(gridExtra)
library(readxl)
library(tidyimpute)
library(cluster)
```

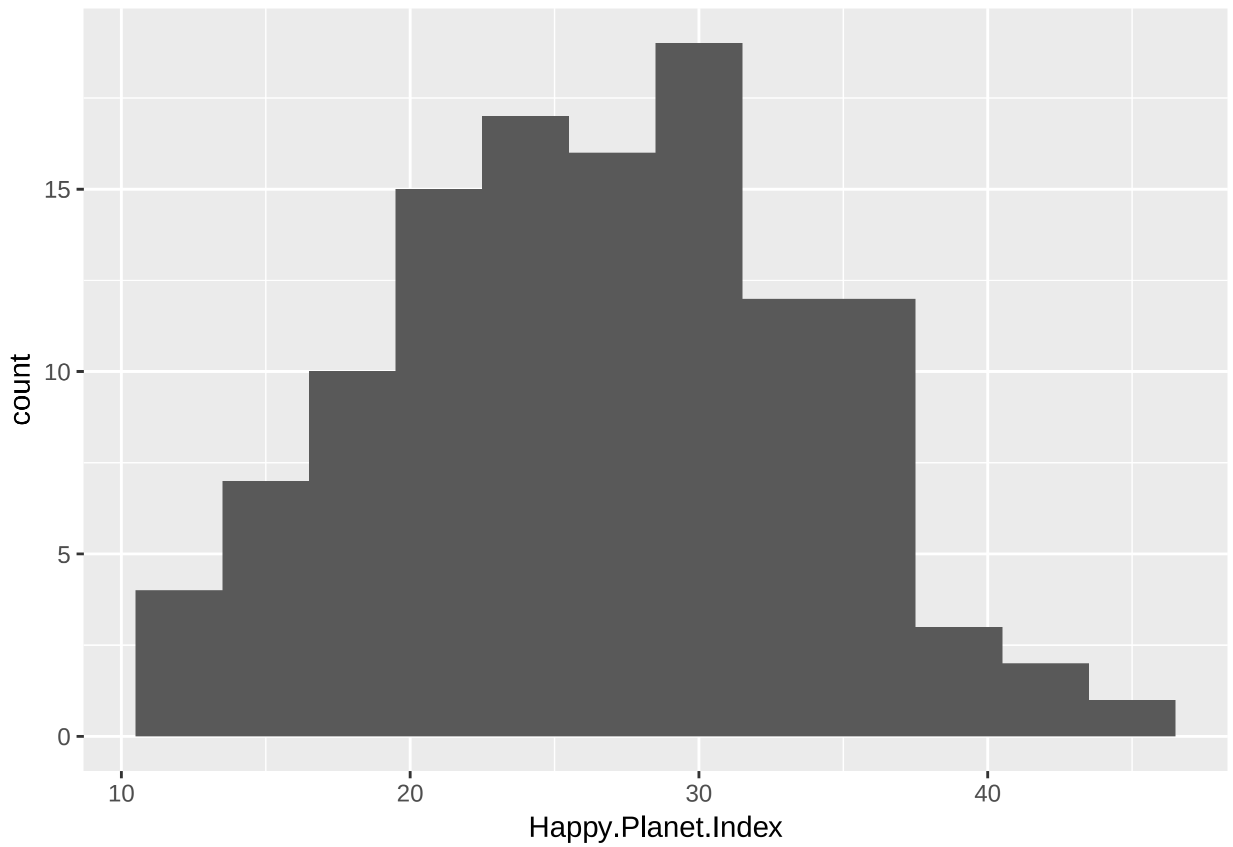
Loading data

```
# Happy Planet Index data from http://happyplanetindex.org/s/hpi-data-2016.xlsx retrieved on 2019-06-01
# Backup on my Dropbox: https://www.dropbox.com/s/cqtynj47altwo3d/hpi-data-2016.xlsx?dl=0

# HPI <- read_excel("Data/hpi-data-2016.xlsx",
#                   sheet = "Complete HPI data",
#                   range = "C6:N146",
#                   trim_ws = TRUE)
#
# write_csv(HPI, "hpi.csv")

url <- "https://raw.githubusercontent.com/codrin-kruijne/HappyPlanetIndex/master/hpi.csv"
hpi_data <- getURL(url)
HPI <- read.csv(textConnection(hpi_data))
```

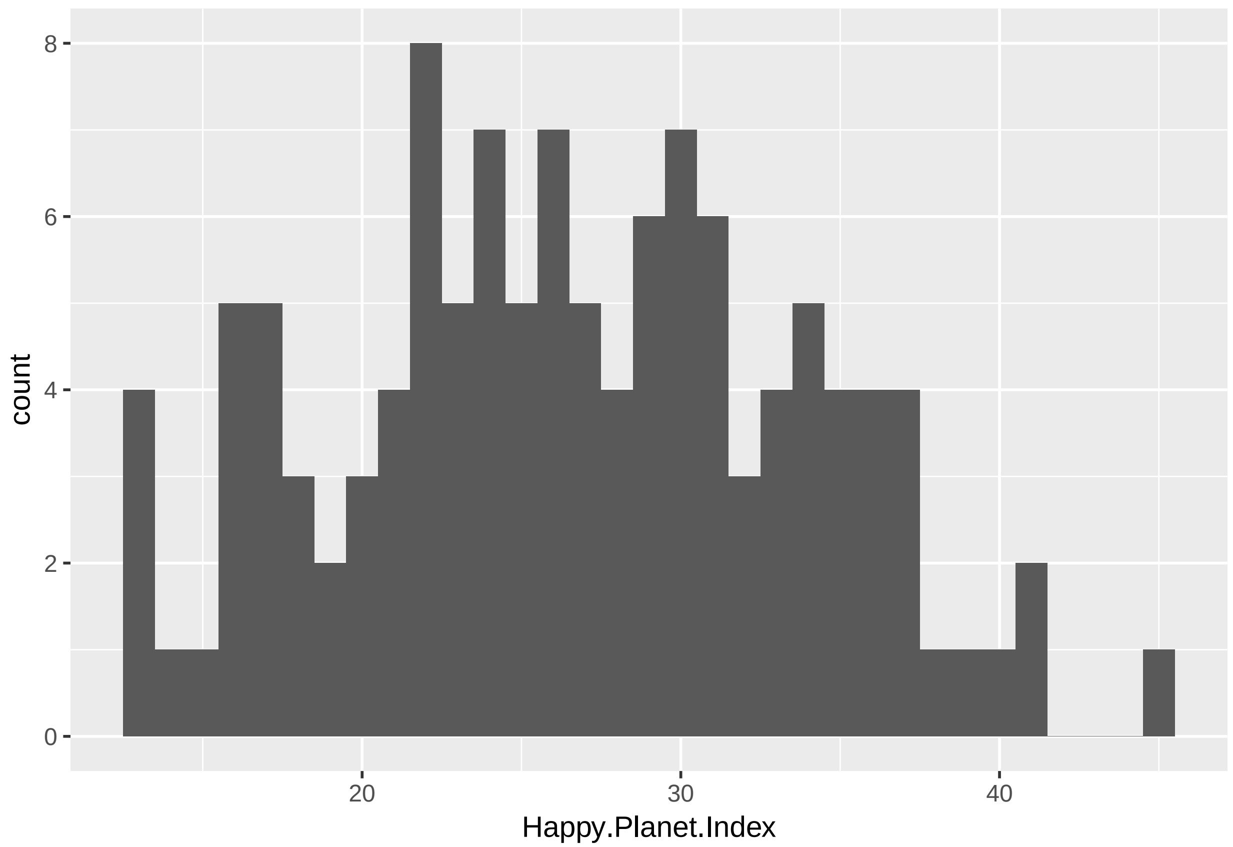
```
HPI[1:2] <- lapply(HPI[1:2], as.factor)
HPI <- HPI %>% mutate(Footprint = 1.73 - `Footprint..gha.capita.`) # Calculate net footprint by subtrac

# Social Progress Initiative data from https://www.socialprogress.org/download retrieved in 2019-06-01
# Backup on my Dropbox: https://www.dropbox.com/s/hb2e3h5l69n0vrx/Social%20Progress%20Index%202018-Resu

# SPI <- read_excel("Data/Social Progress Index 2018-Results.xlsx",
#                   sheet = "2016",
#                   range = "A1:BQ147", # data for countries that are in the index
#                   trim_ws = TRUE)
#
# write_csv(SPI, "spi.csv")

spi_url <- "https://raw.githubusercontent.com/codrin-kruijne/HappyPlanetIndex/master/spi.csv"
spi_data <- getURL(spi_url)
SPI <- read.csv(textConnection(spi_data))

SPI[3:76] <- lapply(SPI[3:69], as.numeric)
SPI[1:2] <- lapply(SPI[1:2], as.factor)
SPI <- SPI %>% select(-Code)
```

```
# Explore differing countries ### FIX DIFFERENT SPELLINGS
diff_countries <- HPI %>% anti_join(SPI, by = "Country") %>% select(Country)

# Join data
raw <- HPI %>% inner_join(SPI, by = "Country") %>%
                mutate(Country = as.factor(Country))

# Impute scaled data
imputed <- raw %>% impute_all(.na = mean, na.rm = TRUE)

# Scale numeric data
scaled <- imputed %>% mutate_if(is.numeric, scale) %>%
                      rename_if(is.numeric, paste, "SCALED")
```

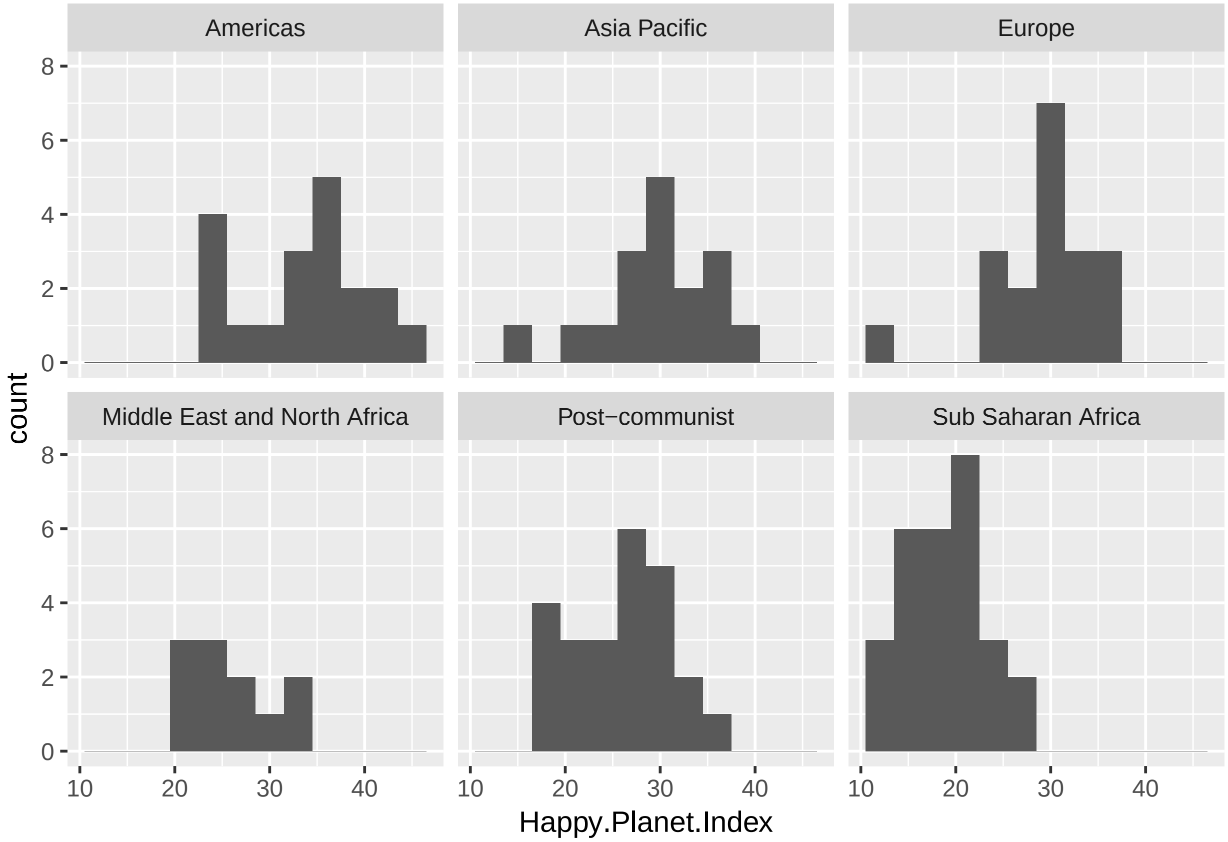
```
# Let's have a quick look at the HPI data
hpi_hist_3 <- ggplot(raw, aes(`Happy.Planet.Index`)) + geom_histogram(binwidth = 3)
saveRDS(hpi_hist_3, "hpi_hist_3.rds")
hpi_hist_1 <- ggplot(raw, aes(`Happy.Planet.Index`)) + geom_histogram(binwidth = 1)
saveRDS(hpi_hist_1, "hpi_hist_1.rds")

hpi_region_hist <- ggplot(raw, aes(`Happy.Planet.Index`)) + facet_wrap(~Region) +
                                         geom_histogram(binwidth = 3)
saveRDS(hpi_region_hist, "hpi_region_hist.rds")

hpi_scatter <- ggplot(raw, aes(x = `Happy.Life.Years`, y = Footprint, size = `Happy.Planet.Index`, colo
  geom_point()
saveRDS(hpi_scatter, "hpi_scatter.rds")

plot(hpi_hist_3)
```



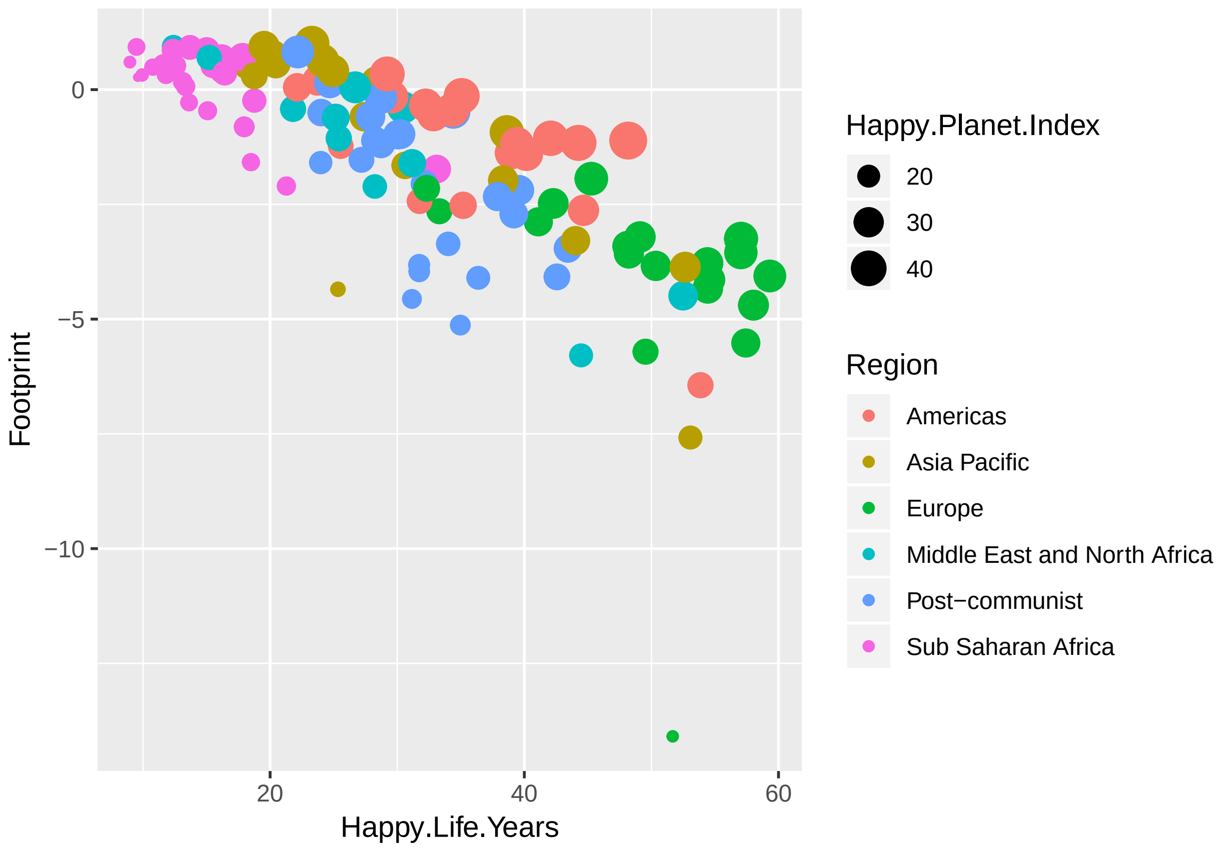
```r
plot(hpi_hist_1)
```



```r
plot(hpi_region_hist)
```

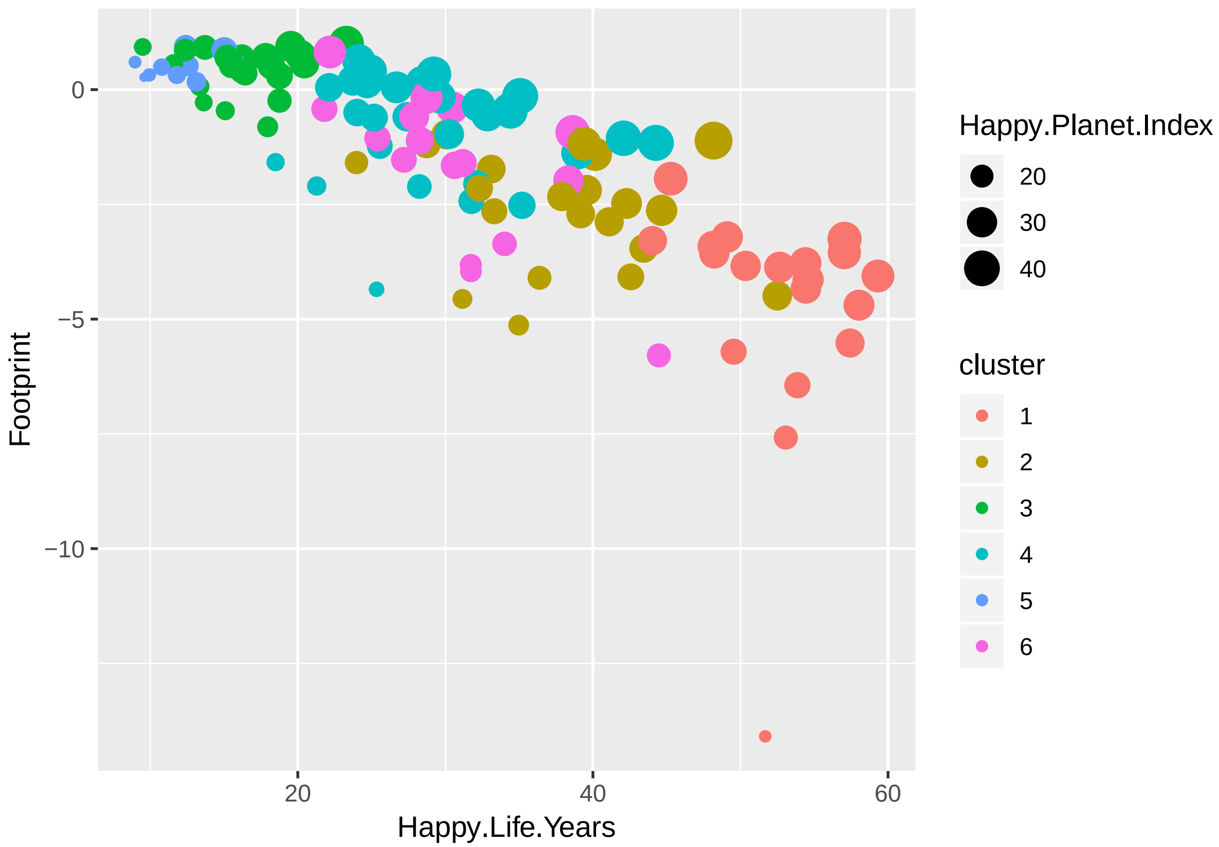


```
plot(hpi_scatter)
```



Cluster modeling

```
data_clusters <- kmeans(scaled[, -c(1, 2)], 6, nstart = 20)

raw_clustered <- mutate(raw, cluster = as.factor(data_clusters$cluster))
scaled_clustered <- mutate(scaled, cluster = as.factor(data_clusters$cluster))

cluster_expl <- ggplot(raw_clustered, aes(x = `Happy.Life.Years`, y = `Footprint`, size = `Happy.Planet
saveRDS(cluster_expl, "cluster_expl.rds")

plot(cluster_expl)
```



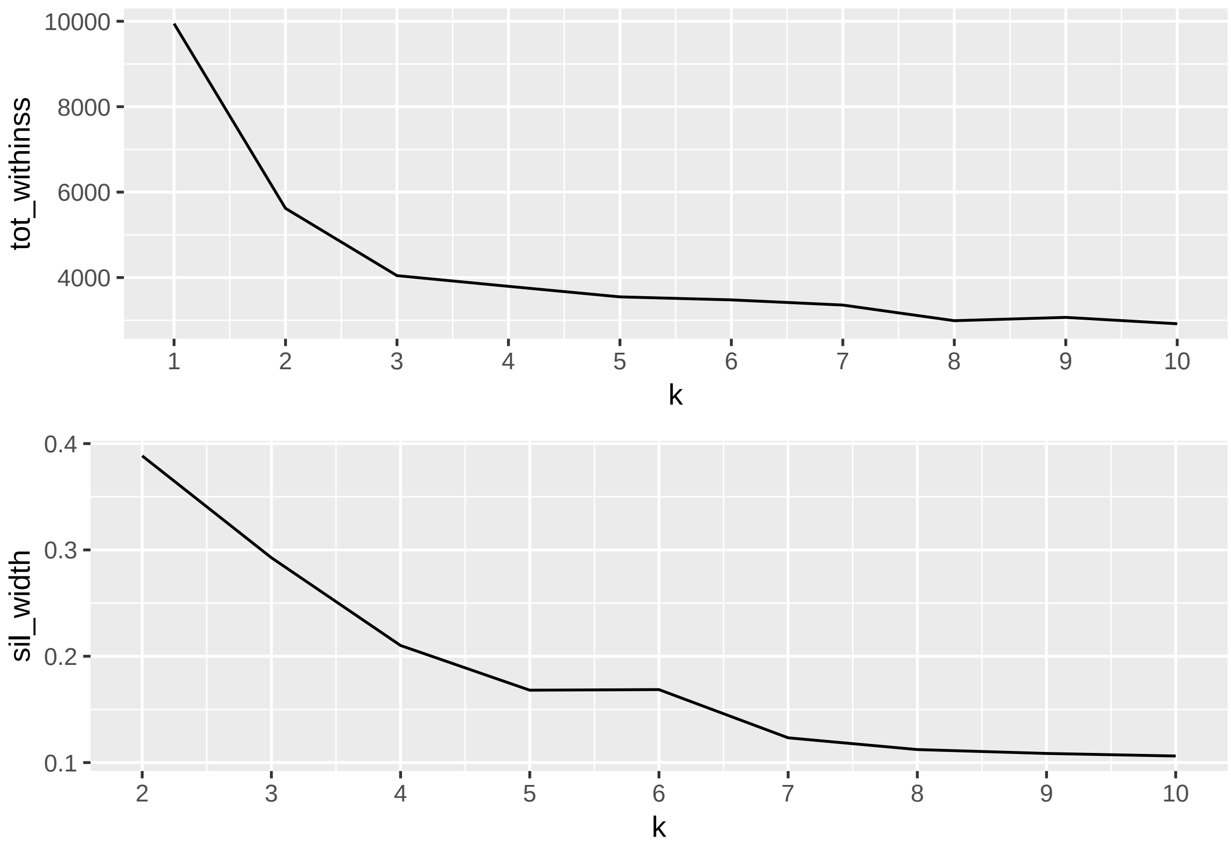
```
# Using the elbow method

tot_withinss <- map_dbl(1:10,  function(k){
  model <- kmeans(x = scaled[, -c(1, 2)], centers = k)
  model$tot.withinss
})

elbow_df <- data.frame(
  k = 1:10,
  tot_withinss = tot_withinss
)

print(elbow_df)
```

```
##     k tot_withinss
## 1   1     9945.000
## 2   2     5619.825
## 3   3     4045.048
## 4   4     3794.686
## 5   5     3547.508
## 6   6     3477.236
## 7   7     3355.714
## 8   8     2989.959
## 9   9     3067.542
## 10 10     2918.336
```

```
elbow_plot <- ggplot(elbow_df, aes(x = k, y = tot_withinss)) +
                geom_line() +
                scale_x_continuous(breaks = 1:10)
saveRDS(elbow_plot, "elbow_plot.rds")

# Silhouette width method

sil_width <- map_dbl(2:10,  function(k){
  model <- pam(x = scaled[, -c(1, 2)], k = k)
  model$silinfo$avg.width
})

sil_df <- data.frame(
  k = 2:10,
  sil_width = sil_width
)

print(sil_df)
```

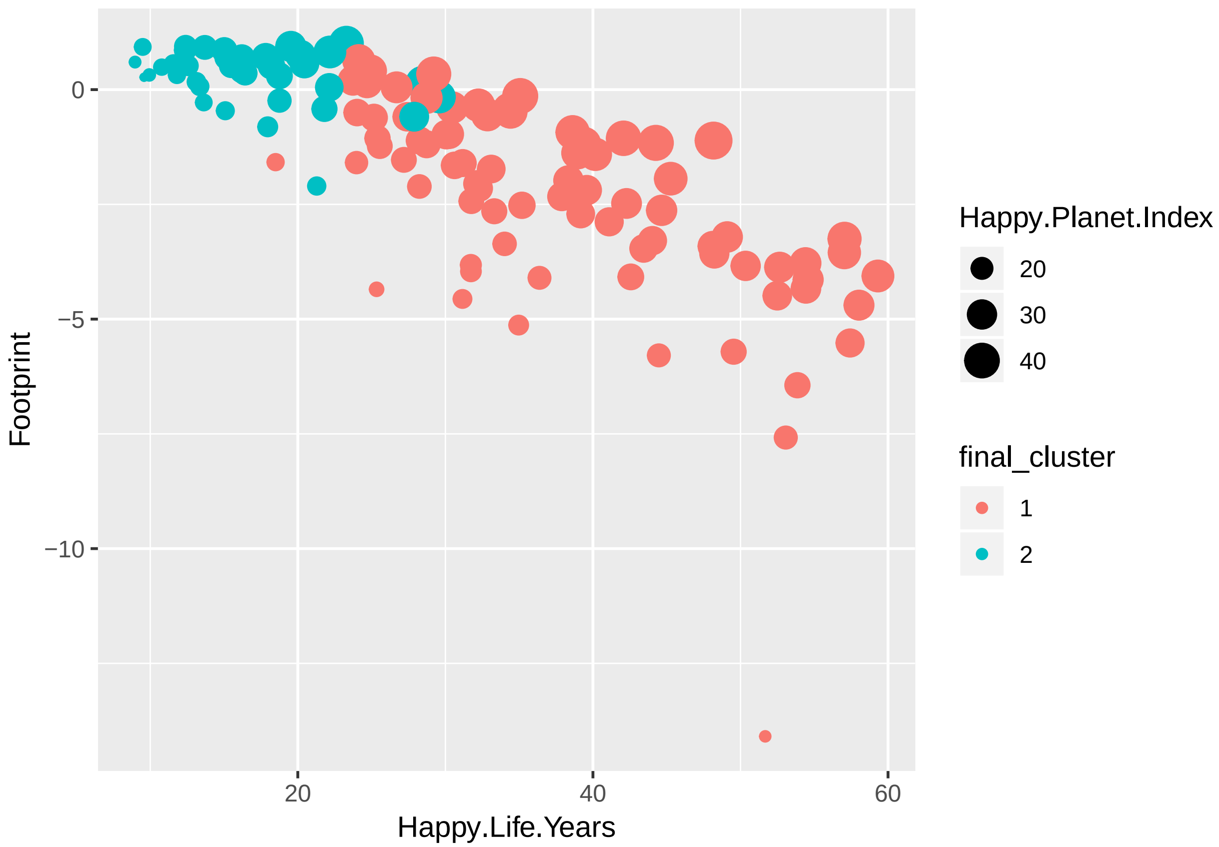
```
##    k sil_width
## 1  2 0.3884523
## 2  3 0.2926770
## 3  4 0.2101102
## 4  5 0.1680787
## 5  6 0.1686458
## 6  7 0.1232974
## 7  8 0.1122524
## 8  9 0.1085817
## 9 10 0.1061960
```

```
sil_width_plot <- ggplot(sil_df, aes(x = k, y = sil_width)) +
                    geom_line() +
                    scale_x_continuous(breaks = 2:10)
saveRDS(sil_width_plot, "sil_width_plot.rds")

# Plot
grid.arrange(elbow_plot, sil_width_plot)
```

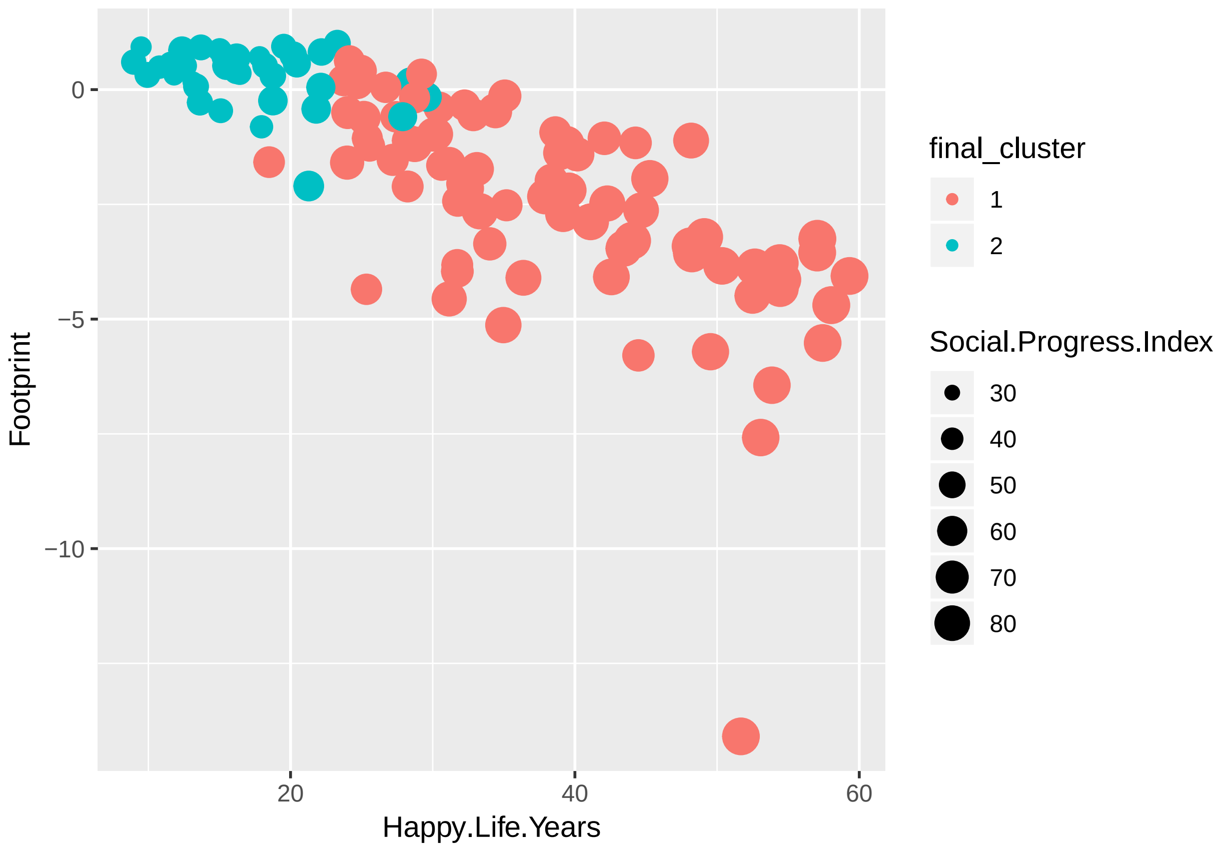


Visualising clustered data

```
# It seems 2 or 3 clusters is optimal, let's have a look

bi_clusters <- kmeans(scaled[, -c(1, 2)], centers = 2, nstart = 20)

raw_clustered <- raw_clustered %>% mutate(final_cluster = as.factor(bi_clusters$cluster))

bi_plot_hpi <- ggplot(raw_clustered, aes(x = `Happy.Life.Years`, y = `Footprint`, size = `Happy.Planet.I
saveRDS(bi_plot_hpi, "bi_plot_hpi.rds")

bi_plot_spi <- ggplot(raw_clustered, aes(x = `Happy.Life.Years`, y = `Footprint`, size = `Social.Progres
saveRDS(bi_plot_spi, "bi_plot_spi.rds")

# Arrange plots
plot(bi_plot_hpi)
```

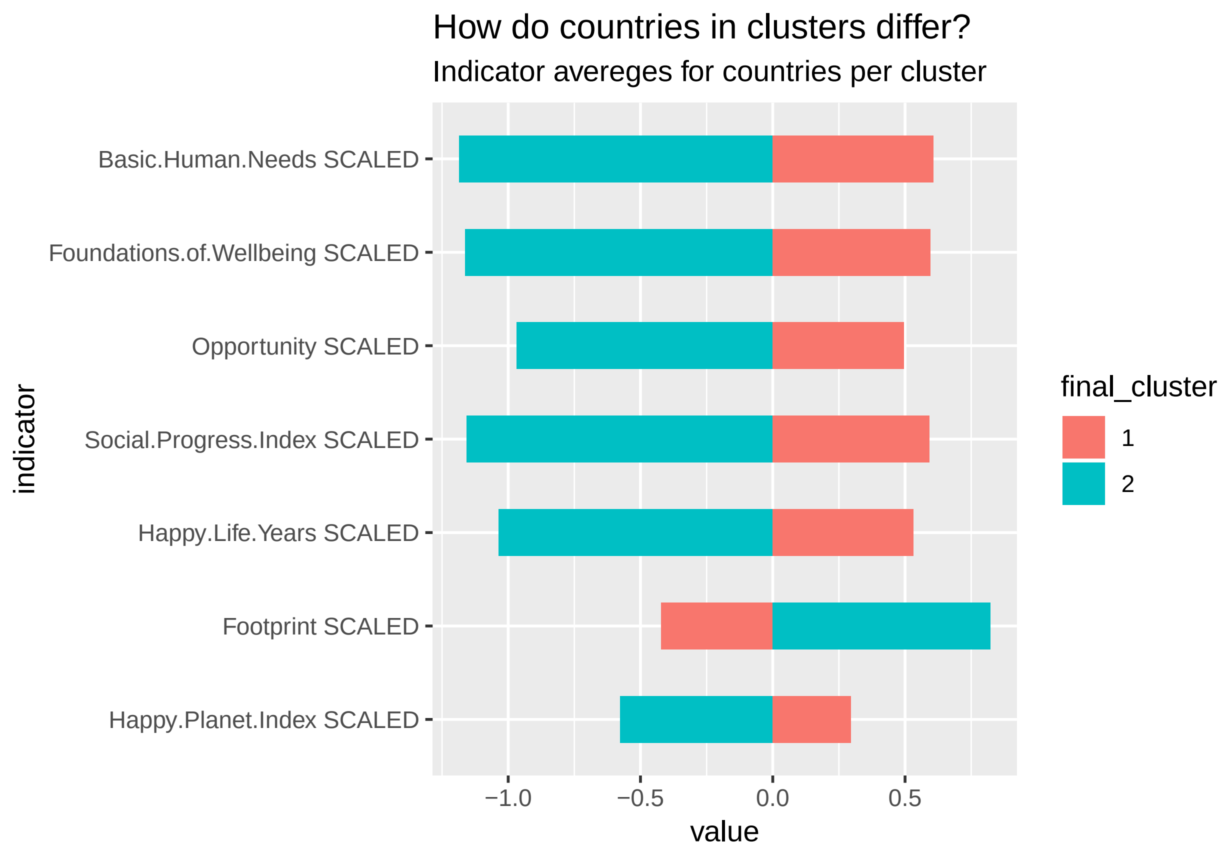


```
plot(bi_plot_spi)
```



Cluster analysis

```
# Lets combina all data for analysis and visualisation
data <- raw_clustered %>% inner_join(scaled_clustered, by = c("Country", "Region"))

# Statistics per bi_cluster to compare
fct_order <- c("Basic.Human.Needs SCALED",
               "Foundations.of.Wellbeing SCALED",
               "Opportunity SCALED",
               "Social.Progress.Index SCALED",
               "Happy.Life.Years SCALED",
               "Footprint SCALED",
               "Happy.Planet.Index SCALED")

bi_averages <- data %>% select(`Happy.Life.Years SCALED`,
                               `Footprint SCALED`,
                               `Happy.Planet.Index SCALED`,
                               `Basic.Human.Needs SCALED`,
                               `Foundations.of.Wellbeing SCALED`,
                               `Opportunity SCALED`,
                               `Social.Progress.Index SCALED`,
                                final_cluster) %>%
                        group_by(final_cluster) %>%
                        summarise_all(mean) %>%
                        gather(-final_cluster, key = "indicator", value = "value") %>%
                        mutate(indicator = as.factor(indicator))

bi_averages$indicator <- fct_rev(fct_relevel(bi_averages$indicator, fct_order))

# Visualising bi averages scaled
diff_bar <- ggplot(bi_averages, aes(x= indicator, y = value, label = value)) +
              geom_bar(stat = "identity", aes(fill = final_cluster), width = 0.5)  +
              labs(title = "How do countries in clusters differ?",
                   subtitle = "Indicator avereges for countries per cluster") +
              coord_flip()
saveRDS(diff_bar, "diff_bar.rds")
plot(diff_bar)
```



# Appendix

```
# Which countries are in these cluster?
cluster_1 <- data %>% filter(final_cluster == 1) %>% select(Country)
cluster_2 <- data %>% filter(final_cluster == 2) %>% select(Country)

k1 = data.frame("Cluster 1" = cluster_1[1:(nrow(cluster_1)/2),])
k2 = data.frame("Cluster 1" = cluster_1[(nrow(cluster_1)/2+1):nrow(cluster_1),])
k3 = data.frame("Cluster 2" = cluster_2[1:(nrow(cluster_2)/2),])
k4 = data.frame("Cluster 2" = cluster_2[(nrow(cluster_2)/2+1):nrow(cluster_2),])
country_list <- list(k1, k2, k3, k4)
```

```r
saveRDS(country_list, "country_list.rds")
knitr::kable(country_list)
```

```r
# How long did the whole script take?
script_end <- Sys.time()

print(paste("Total script running time: ", round(difftime(script_end, script_start, units = "mins"), 1)
```

```
## [1] "Total script running time:  0.1  minutes"
```

Cluster.1
Albania
Algeria
Argentina
Armenia
Australia
Austria
Belarus
Belgium
Bhutan
Bolivia
Brazil
Bulgaria
Canada
Chile
China
Colombia
Costa Rica
Croatia
Cyprus
Czech Republic
Denmark
Dominican Republic
Ecuador
El Salvador
Estonia
Finland
France
Georgia
Germany
Greece
Hungary
Iceland
Iran
Ireland
Israel
Italy
Japan
Kazakhstan
Kyrgyzstan

Cluster.1
Latvia
Lebanon
Lithuania
Luxembourg
Macedonia
Malaysia
Mauritius
Mexico
Mongolia
Montenegro
Morocco
Netherlands
New Zealand
Nicaragua
Norway
Oman
Panama
Paraguay
Peru
Philippines
Poland
Portugal
Romania
Russia
Serbia
Slovakia
Slovenia
South Africa
Spain
Sri Lanka
Suriname
Sweden
Switzerland
Thailand
Tunisia
Turkey
Ukraine
United Kingdom
Uruguay

Cluster.2
Afghanistan
Bangladesh
Benin
Botswana
Burkina Faso
Burundi
Cambodia
Cameroon
Chad
Comoros
Djibouti
Egypt
Ethiopia
Ghana
Guatemala
Guinea
Honduras
India
Indonesia
Kenya

Cluster.2
Lesotho
Liberia
Malawi
Mauritania
Mozambique
Myanmar
Nepal
Niger
Nigeria
Pakistan
Rwanda
Senegal
Sierra Leone
Swaziland
Tajikistan
Tanzania
Togo
Uzbekistan
Yemen
Zimbabwe