什么是幸福

——基于因子和聚类的幸福指数分析

致理-数理 1　刘苏青* 2021013371

2023 / 06 / 23

摘　要

本文基于 2015 年到 2022 年的世界幸福指数报告，采用主成分分析、因子分析、回归分析和聚类分析等统计学方法，深入探讨了影响人们幸福程度的主要因素，定量预测了不同因子得分下人们的幸福指数，并定性分析了幸福程度在空间和时间上的分布，有助于帮助人们建立起对于幸福感的再认识。

关键词：幸福指数、数据降维、因子分析、回归分析、聚类分析

*liu-sq21@mails.tsinghua.edu.cn

目录

1 研究背景

2023 年 3 月的第十四届全国人民代表大会第一次会议上曾强调“必须以满足人民日益增长的美好生活需要为出发点和落脚点”。究竟什么更会增加人们的『幸福感』，是亲情、自由、社会地位、金钱，还是其他因素?

针对这一话题，本文基于 2015 年到 2022 年的世界幸福指数报告，并采用多元统计分析课程讲授的方法，提出并尝试回答以下亟待解决的问题，意图帮助大家寻找『幸福感』的落脚点:

- 幸福程度和什么变量有关? 是否存在潜在的影响因子?
- 如何通过其他已知信息来评估一个国家的幸福程度?
- 幸福程度是否会受到空间和时间的影响?

2 探索性数据分析

2.1 数据集描述

本研究所使用的数据来自课程所提供的 World Happiness Report 数据集，这是一份从 2015 年到 2022 年的世界幸福指数报告，收集了 195 个国家的幸福指数和相关指标，包含了这 8 年以来共计 1229 个样本[1]。除去前两列的序号信息，该数据集共有 10 个变量，其中前两个变量分别为国家名称 (Country) 和所属地区 (Region)，第三个变量为幸福指数 (Happiness Score)，第四至九个变量分别为经济情况 (Economy/GDP per Capita)、社会支持 (Family/Social Support)、预期寿命 (Health/Life Expectancy)、自由 (Freedom)、公信力 (Trust/Government Corruption) 和慷慨 (Generosity)，最后一个变量是年份 (Year)。除国家名称、所属地区和年份是离散型变量，其余各变量均为连续型变量。

2.2 数据预处理

对数据集 happiness.csv 中的数据进行初步观察，不难发现该数据集存在以下问题:

- 在 2022 年的数据中，第三至九个变量的数据中的小数点“.”变为了逗号“,”;

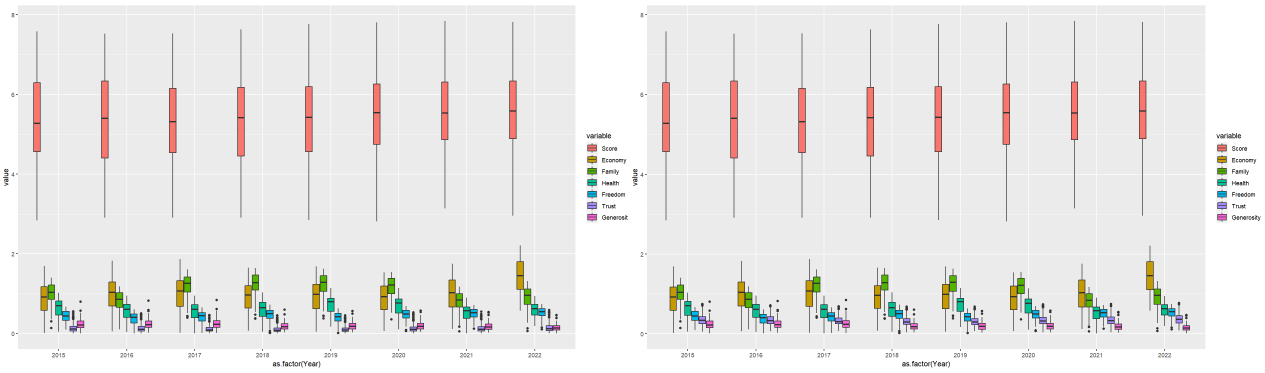
[1]不是每个国家都有这 8 年以来的幸福指数报告，部分国家在某些年份缺失幸福指数报告。

- 在 2022 年的数据中，部分国家名称后面出现星号 “*”；
- 在 2015 年到 2022 年的数据中，不同年份下部分国家名称并未统一，例如香港、台湾、刚果等地；
- 在 2015 年到 2022 年的数据中，部分数据记录为 0，实则可能是缺失值 NA。

因此，我们需要根据以上问题对数据进行预处理 (见代码)。

2.3 单变量描述

首先，我们可以绘制第三至九个变量关于年份的多组箱线图如下，以考察各变量在各年份的尺度信息以及异常值情况：

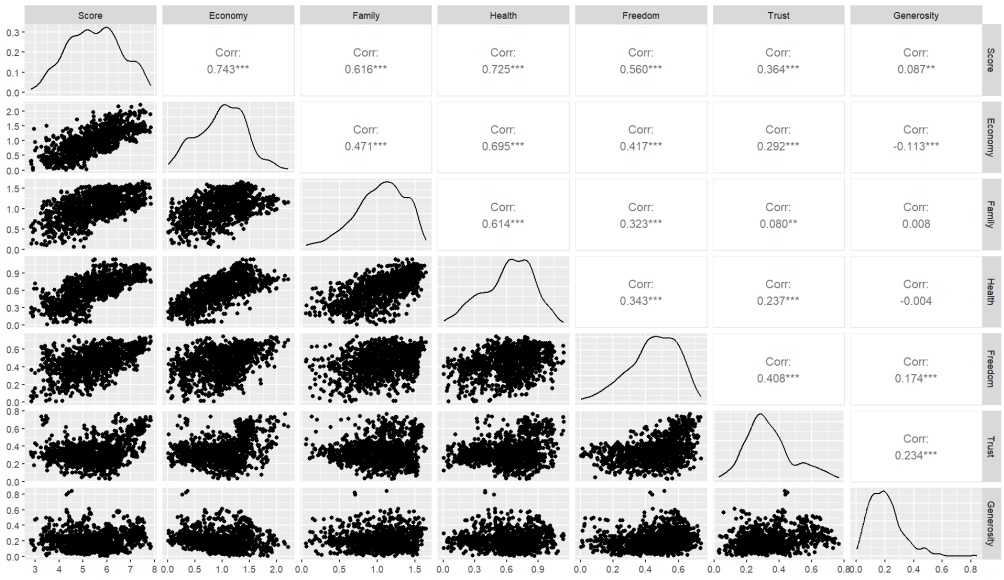


由左图可知，各变量在不同年份的尺度大致相同；除因变量 (幸福指数) 以外，其余各自变量的尺度大致相同；除公信力变量呈明显右偏分布而导致其异常值较多，其余各变量异常值均较少。因此，我们考虑对公信力变量进行平方根变换以降低其偏度，同时尽量不改变其尺度信息，处理后的结果如右图所示。

2.4 相关性描述

然后，我们可以绘制第三至九个变量的散点图矩阵如下[2]，以考察各变量之间的相关性：

[2]这里引入各变量在不同年份的尺度相同的假设，方便将各变量在不同年份的数据合并到一起来处理。

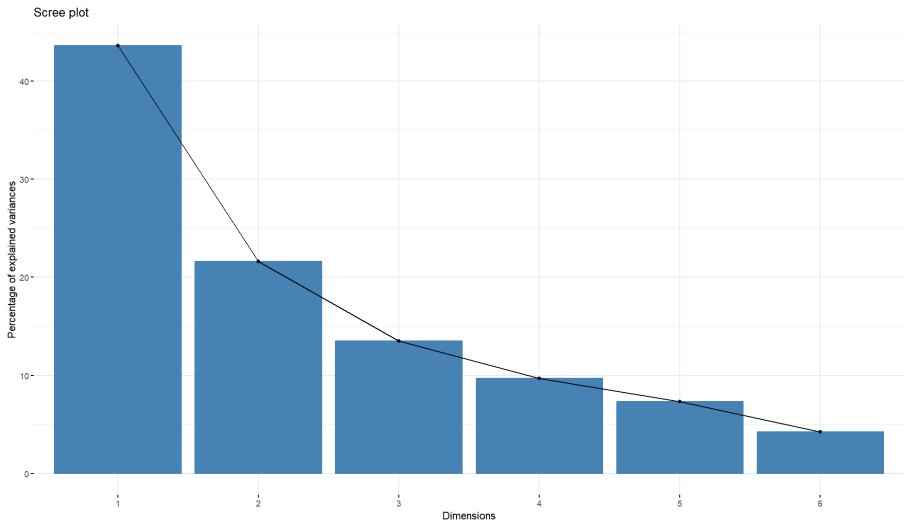


由上图可知，社会支持、预期寿命和自由变量呈左偏分布，而慷慨变量呈右偏分布，因此在后续的分析中，我们应当尽量避免引入正态性假设等条件。此外，因变量与各自变量之间、各自变量之间均存在较强的相关性，因此我们需要考虑对自变量进行数据降维以降低各自变量之间的共线性。

3 数据降维

3.1 主成分分析

在数据降维的过程中，最重要的一步就是确定数据应该降到的维数 n，我们可以采用主成分分析 (PCA) 的方法来绘制崖底碎石图如下[3]：



[3]尽管各自变量的尺度大致相同，我们在这里仍需要进行标准化处理，以消除未知量纲造成的影响。

由上图可知，$n = 3$ 是一个拐点，前三个变量可解释的方差占比较大，约为 78.7%，因此我们可以考虑将六个自变量降成三维。

3.2 因子分析

下面我们采用因子分析 (FA) 的方法对自变量进行降维，同时获得自变量背后潜在的公共因子。根据前一节的结果，我们考虑取因子个数 $n = 3$，分别通过主成分 (PC) 的方法和极大似然估计 (MLE) 的方法来求解因子模型，并采用 varimax 的旋转方法来增强因子的解释性，然后通过 corrplot 包绘制可视化结果如下：

经比较可知，PC 方法的三个因子可解释的方差占比约为 78.7%，MLE 方法的三个因子可解释的方差占比约为 61.6%(见代码)。此外，因子分析所得结果如下表所示，PC 方法获得的公共因子实际意义更明确：

	RC1	RC3	RC2
Economy	0.725	0.415	−0.272
Family	0.879	−0.038	0.121
Health	0.861	0.223	−0.052
Freedom	0.358	0.679	0.141
Trust	0.006	0.896	0.100
Generosity	−0.025	0.169	0.959

由因子载荷可知，RC1 主要影响经济情况、社会支持和预期寿命，RC3 主要影响自由和公信力，RC2 主要影响慷慨。从可解释性的角度来说，RC1 可解释为物质条件因子，RC3 可解释为社会环境因子，RC2 可解释为邻里氛围因子。因此，我们可以认为幸福指数与物质条件、社会环境和邻里氛围有关，且每个样本的因子得分可视为降维后的数据。

4 回归分析

鉴于因变量幸福指数并非离散 (逻辑) 型变量而是连续型变量，此处不适合采用判别分析 (DA) 的方法来进行分类，所以我们可以考虑采用线性回归分析的方法来进行预测。假设幸福指数与因子得分满足线性模型：

$$\text{幸福指数} = \beta_0 + \beta_1 \cdot \text{物质条件} + \beta_2 \cdot \text{社会环境} + \beta_3 \cdot \text{邻里氛围}$$

多重线性回归所得结果如下表所示：

	Dependent variable:
	Score
fa.scoreRC1	0.818***
	(0.017)
fa.scoreRC3	0.466***
	(0.017)
fa.scoreRC2	0.023
	(0.017)
Constant	5.467***
	(0.017)
Observations	1,184
R^2	0.730
Adjusted R^2	0.730
Residual Std. Error	0.573 (df = 1180)
F Statistic	1,065.960*** (df = 3; 1180)
Note:	*p<0.1; **p<0.05; ***p<0.01

由上表可知幸福指数与因子得分满足线性模型：

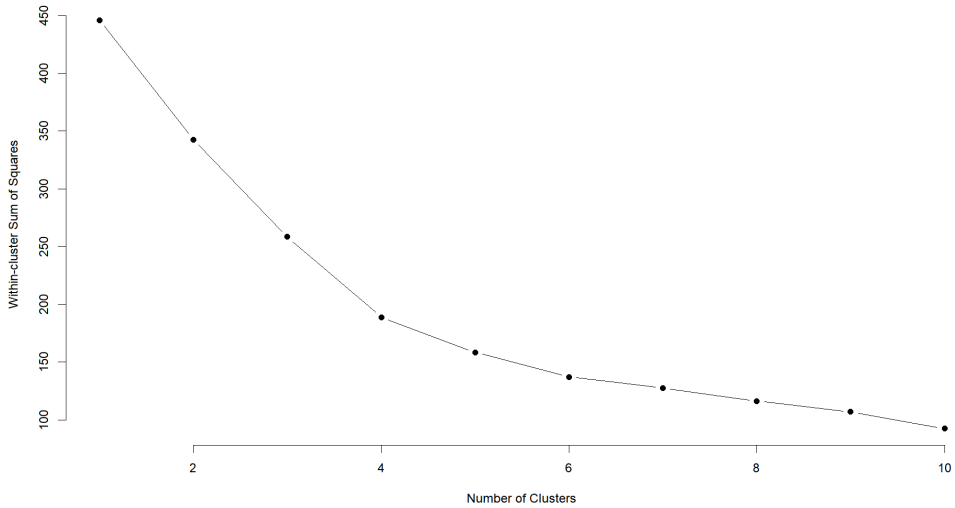
$$\hat{\text{幸福指数}} = 5.4671 + 0.8181 \cdot \text{物质条件} + 0.4662 \cdot \text{社会环境} + 0.0231 \cdot \text{邻里氛围}$$

因此，当我们知道某个国家其他各指标的数据时，我们可以先通过因子分析获得其因子得分，然后通过回归分析预测其幸福指数。

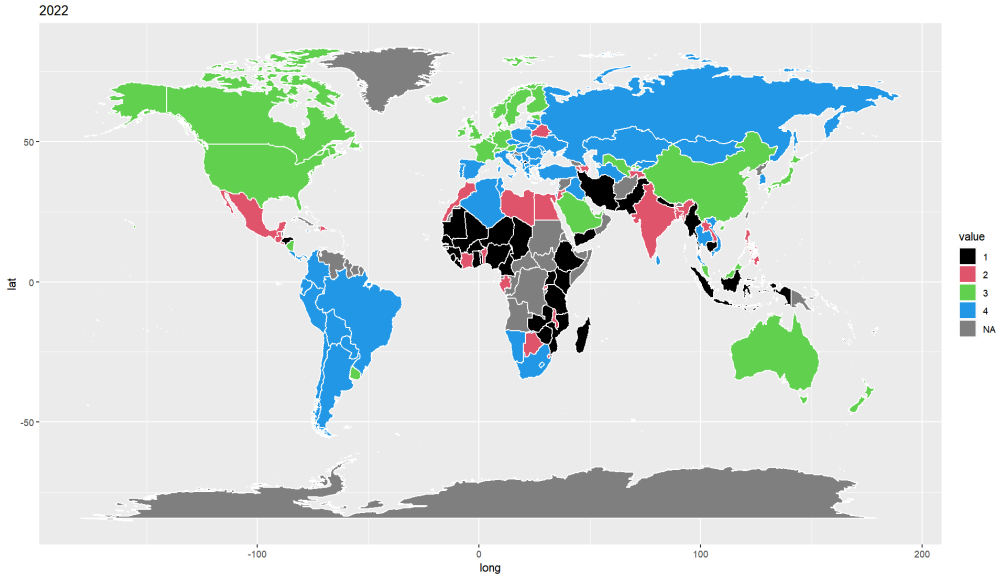
5 聚类分析

5.1 聚类在空间的分布

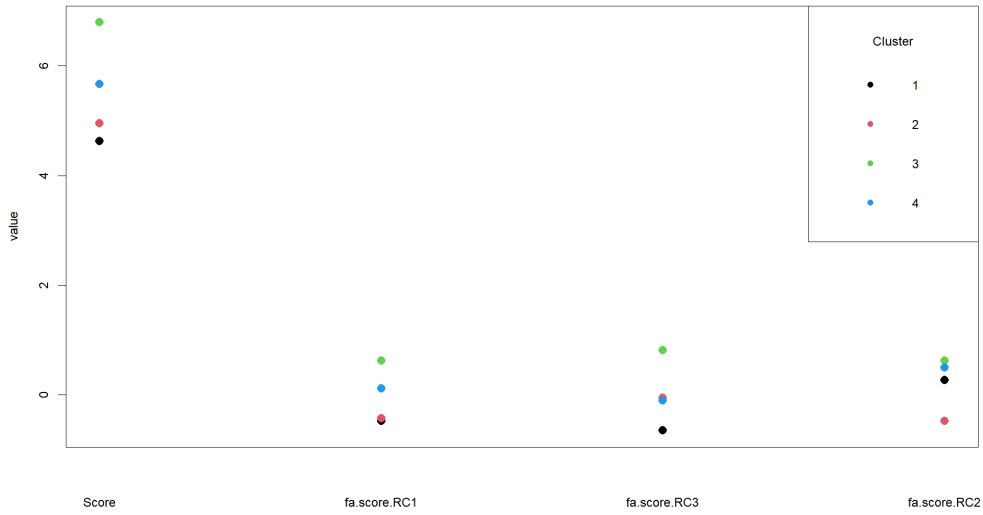
接下来，我们尝试对这些国家进行聚类分析，来观察各聚类下各国家的特征，以及各聚类随空间和时间的变化关系。首先，我们选取前文中获得的样本的因子得分作为降维后的数据，并以 2022 年为例，采用 k-means 的方法来研究聚类在空间的分布。为了确定最佳的聚类个数 n，我们可以绘制崖底碎石图如下：



由上图可知，$n = 4$ 是一个拐点，因此我们可以考虑将样本聚成四类，然后绘制可视化结果如下，其中相同颜色的国家属于同一类：



分别绘制四个聚类的幸福指数和因子得分的平均值如下：

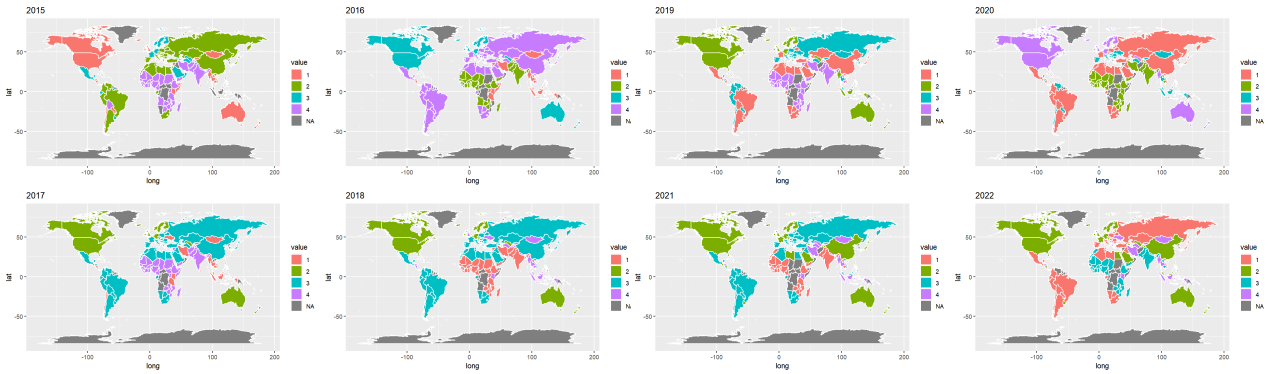


如上图所示，由 k-means 方法所得的聚类结果基本等同于按照幸福程度进行分类的结果。第一个聚类[4]的特点是幸福指数低、物质条件差、社会环境差和邻里氛围适中，代表国家有赤道附近的非洲各国和东南亚各国；第二个聚类的特点是幸福指数低、物质条件差、社会环境适中和邻里氛围差，代表国家有北非和南亚各国；第三个聚类的特点是幸福指数高、物质条件好、社会环境好和邻里氛围好，代表国家有北美、西欧、澳洲和东亚各国；第四个聚类的特点是幸福指数适中、物质条件适中、社会环境适中和邻里氛围好，代表国家有南美、南非和东欧各国。从地理学的角度来看，第一和第二个聚类易受热带气候及自然灾害的影响而经济不稳定，所以幸福指数较低；第三个聚类有雄厚的经济实力、充裕的社会福利和稳定的政治体系，所以幸福指数较高；第四个聚类虽然发展情况尚可，但仍存在一些经济和社会问题亟待解决，所以幸福指数适中。

5.2 聚类随时间的演化

然后，我们取 2015 年到 2022 年的全部数据，仍采用 k-means 的方法来研究聚类随时间演化的空间分布，并绘制可视化结果如下；

[4]这里的第一、二、三和四个聚类仅针对上图和下图，对于别的聚类结果，其对应关系可能发生变化。



由上图可知，从 2015 年到 2022 年，聚类在空间的分布大体不随时间而演化。极个别国家，例如中国，在这 8 年期间的经济和社会发展情况相交其他国家提升较快，因而从幸福指数适中的聚类跃升到了幸福指数较高的聚类。这一现象证明非战乱动荡时期，世界各国的经济和社会发展情况基本保持稳定，或者在一定范围内缓慢增长。

6 总结与改进

通过以上的数据分析，我们不难得出以下结论来回答本文开头处抛出的三个问题：

- 幸福程度和经济情况、社会支持、预期寿命、自由、公信力、慷慨六个自变量均强相关。但是由于六个自变量之间存在较强的共线性，我们可以对其进行降维。根据因子分析的方法，我们可以提炼出三个潜在的公共因子：物质条件、社会环境和邻里氛围。

- 如果一个国家的经济情况、社会支持、预期寿命、自由、公信力和慷慨均已知，那么我们可以采用因子分析的方法计算出该国家的因子得分，然后利用 $\widehat{\text{幸福指数}} = 5.4671 + 0.8181 \cdot \text{物质条件} + 0.4662 \cdot \text{社会环境} + 0.0231 \cdot \text{邻里氛围}$ 来预测该国家的幸福程度。

- 幸福程度会受到空间和时间的影响。空间层面上是由于资源分配的空间异质性以及经济和社会发展的区域不平衡而造成幸福程度有所差异。时间层面上是由于经济和社会的发展速率不尽相同而造成幸福程度有所变化。

本文虽然获得了一些浅显的结论，但是整个研究仍然存在诸多有待改进的地方：

- 首先，邓老师提供的数据来源为 kaggle 上的 World Happiness Report 数据集，这份数据集与 World Happiness Report 的官方网站所提供的数据集相比略有出入。

针对这一问题，我认为在研究过程中，采用一手数据进行建模和分析的效果应该要优于采用二手数据，因此我们可以考虑改用官方数据重新分析，

- 其次，在采用因子分析的方法对数据进行降维的过程中，我引入了各年份数据尺度大体相同 (独立同分布) 的假设，因而可以合并成一个整体来进行因子分析降维。这种做法虽然提升了样本量，有利于我们在回归分析中获得更精准的结果，且便于我们将所有样本降维到同一个“因子空间”来进行比较。但是这种做法预先抹除了时间对数据的影响，可能导致在后续探究“聚类随时间的演化”时结果不显著甚至无意义。针对这一问题，我认为我们可以考虑引入时间序列的相关知识来获取更好的解决方案。

- 最后，由于本数据集的因变量幸福指数是一个连续型变量，我仅对幸福指数进行了回归预测，而没有对幸福指数进行判别分析。针对这一问题，我认为我们可以采用聚类所得结果作为分类信息，对数据进行有监督学习。不过由于聚类个数 $n = 4$ 属于较为复杂的多分类问题，我们也可以重新将幸福指数高和幸福指数适中的国家分为一类，将幸福指数低的国家分为另一类，然后通过线性判别分析 (LDA)[5]或二次判别分析 (QDA) 等方法来解决分类问题。

7 附录

7.1 R 代码

```r
## packages
library(reshape2)
library(ggplot2)
library(GGally)
library(factoextra)
library(psych)
library(corrplot)
library(gridExtra)
library(cluster)
library(stargazer)


```

[5]线性判别分析需要引入同方差假设，在这种情况下不一定适用

```r
## EDA
rawdata <- read.csv('C:/Users/liusu/Desktop/happiness.csv')
data <- rawdata
colnames(data) <- c('X', 'Rank', 'Country', 'Region', 'Score', 'Economy',
    'Family', 'Health', 'Freedom', 'Trust', 'Generosity', 'Year')
data[, 5:11] <- apply(data[, 5:11], 2, function(x) gsub(',', '.', x))
data$Country <- gsub('\\*', '', data$Country)
data[, 5:12] <- apply(data[, 5:12], 2, as.numeric)
data[, 5:12][data[, 5:12] == 0] <- NA
data <- data[complete.cases(data), ]

##-- box plot
data_long <- melt(as.data.frame(data[, 5:12]), id.vars = 'Year')
boxplot <- ggplot(data_long, aes(x = as.factor(Year), y = value, fill =
    variable)) + geom_boxplot()
boxplot
data$Trust <- sqrt(data$Trust)
data_long <- melt(as.data.frame(data[, 5:12]), id.vars = 'Year')
boxplot2 <- ggplot(data_long, aes(x = as.factor(Year), y = value, fill =
    variable)) + geom_boxplot()
boxplot2

##-- scatter plot
scatterplot <- ggpairs(data[, 5:11])
scatterplot


## PCA
pca <- prcomp(data[, 6:11], scale. = TRUE)
print(pca$rotation[, 1:3])
pcaplot <- fviz_eig(pca)
pcaplot


## FA
##-- pc
pc <- principal(data[, 6:11], nfactors = 3, rotate = 'varimax')
pc
```

```r
stargazer(as.data.frame.matrix(pc$loadings), summary = FALSE)

##-- mle
mle <- factanal(data[, 6:11], factors = 3, rotation = 'varimax')
mle

par(mfrow = c(1, 2))
corrplot(pc$loadings)
title('PC', line = 3)
corrplot(mle$loadings)
title('MLE', line = 3)
par(mfrow = c(1, 1))


## Regression
fa.score <- principal(data[, 6:11], nfactors = 3, rotate = 'varimax', n.
    obs = dim(data)[1], scores = T, method = 'Bartlett')
data$fa.score <- fa.score$scores
reg <- lm(data$Score ~ data$fa.score)
summary(reg)
stargazer(reg)


## CA
world <- map_data('world')
data$Country[!data$Country %in% unique(world$region)]

##-- data correction
data$Country <- gsub('United States', 'USA', data$Country)
data$Country <- gsub('United Kingdom', 'UK', data$Country)
data$Country <- gsub('North Cyprus', 'Cyprus', data$Country)
data$Country <- gsub('Northern Cyprus', 'Cyprus', data$Country)
data$Country <- gsub('Somaliland region', 'Somalia', data$Country)
data$Country <- gsub('North Macedonia', 'Macedonia', data$Country)
data$Country <- gsub('Macedonia', 'North Macedonia', data$Country)
data$Country <- gsub('Palestinian Territories', 'Palestine', data$Country)
data$Country <- gsub('Congo \\(Brazzaville\\)', 'Republic of Congo', data$
    Country)
```

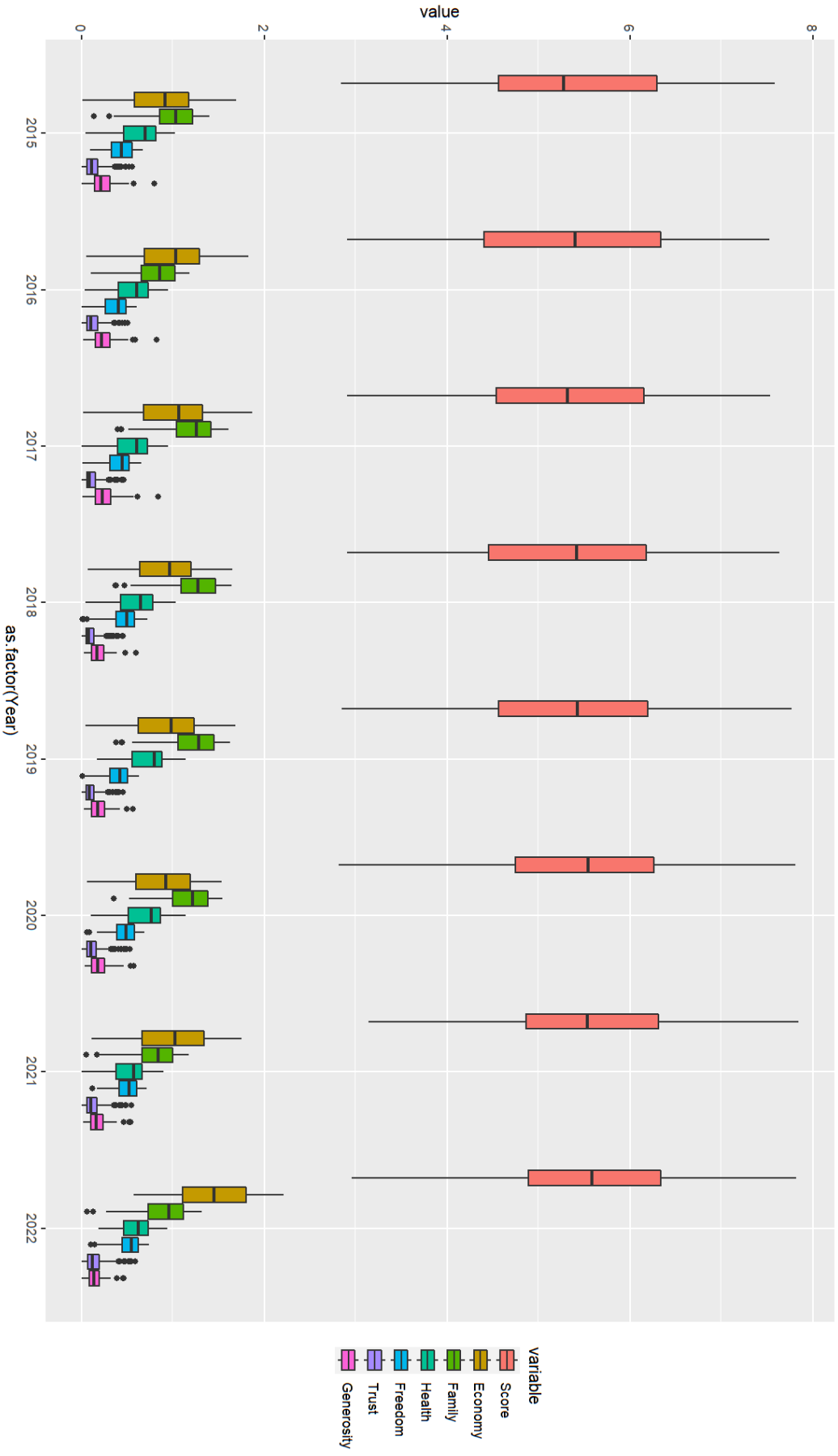
```r
data$Country <- gsub('Congo \\(Kinshasa\\)', 'Democratic Republic of the Congo', data$Country)
data$Country <- gsub('Eswatini, Kingdom of', 'Swaziland', data$Country)
data$Country <- gsub('Somaliland Region', 'Somalia', data$Country)
data$Country <- gsub('Czechia', 'Czech Republic', data$Country)

row_to_duplicate <- data[which(data$Country %in% c('Trinidad and Tobago', 'Trinidad & Tobago')), ]
data$Country <- gsub('Trinidad and Tobago|Trinidad & Tobago', 'Trinidad', data$Country)
data <- rbind(data, row_to_duplicate)
data$Country <- gsub('Trinidad and Tobago|Trinidad & Tobago', 'Tobago', data$Country)

data <- data[!grepl('Hong Kong', data$Country) & !grepl('Taiwan', data$Country) & !grepl('Congo', data$Country), ]

##-- data score (which i didn't use)
##---- (find the best n)
wss <- vector()
max_clusters <- 10
for (k in 1:max_clusters) {
  kmeans_result <- kmeans(data[which(data$Year == 2015), 6:11], centers = k)
  wss[k] <- kmeans_result$tot.withinss
}
plot(1:max_clusters, wss, type = "b", pch = 19, frame = FALSE, xlab = "Number of Clusters", ylab = "Within-cluster Sum of Squares")

plot_list <- list()
for (i in 2015:2018) {
  result <- cbind(data[which(data$Year == i), 'Country'], kmeans(data[which(data$Year == i), 6:11], centers = 3)$cluster)
  colnames(result) <- c('region', 'value')
  map_data <- left_join(world, as.data.frame(result), by = 'region', relationship = 'many-to-many')

  plot <- ggplot(map_data, aes(long, lat, group = group)) +
```

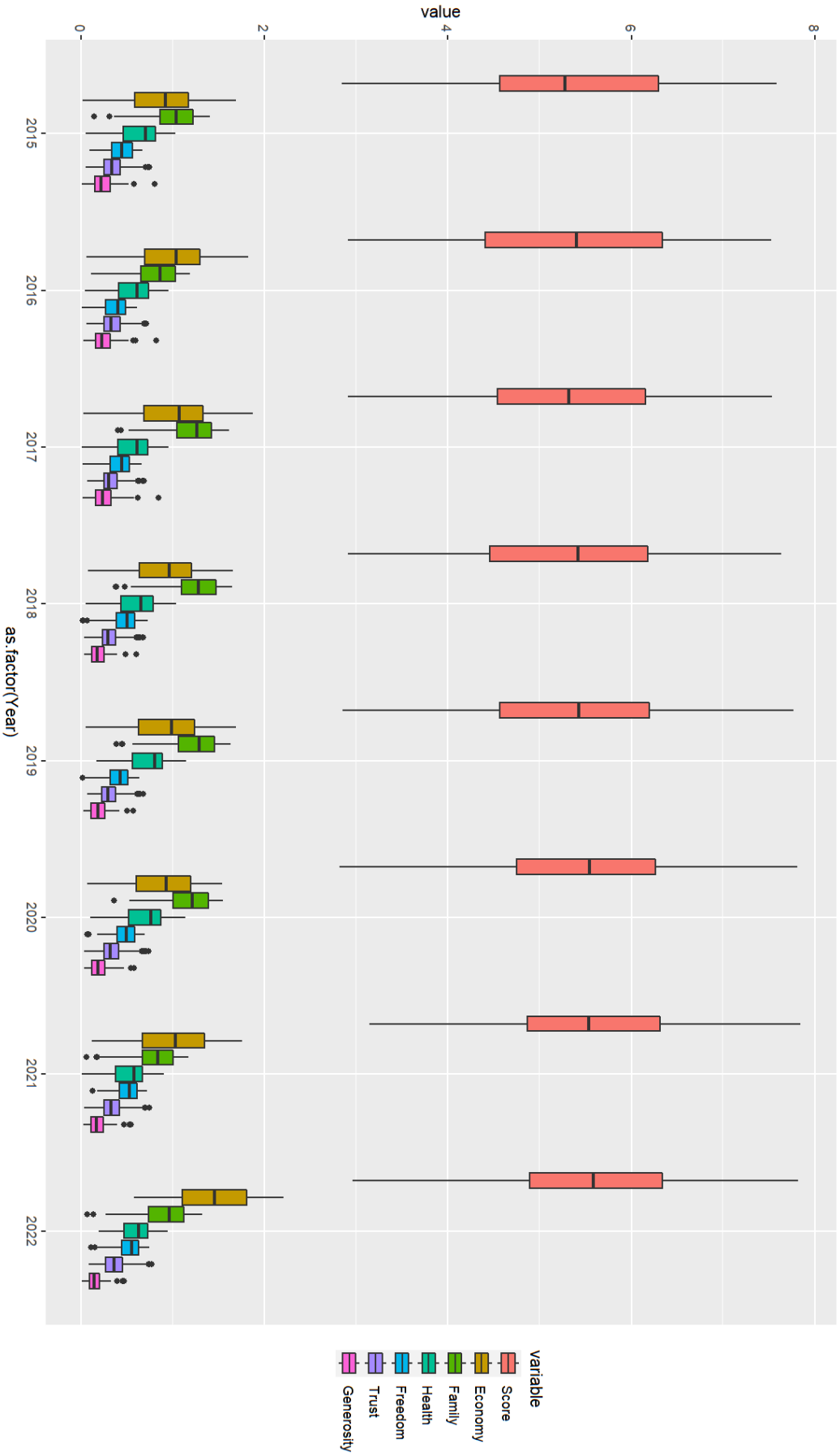
```
    geom_polygon(aes(fill = value), color = 'white') +
    labs(title = as.character(i))
  plot_list[[i - 2014]] <- plot
}
grid.arrange(grobs = plot_list, nrow = 2, ncol = 2)

plot_list <- list()
for (i in 2019:2022) {
  result <- cbind(data[which(data$Year == i), 'Country'], kmeans(data[
      which(data$Year == i), 6:11], centers = 3)$cluster)
  colnames(result) <- c('region', 'value')
  map_data <- left_join(world, as.data.frame(result), by = 'region',
      relationship = 'many-to-many')

  plot <- ggplot(map_data, aes(long, lat, group = group)) +
    geom_polygon(aes(fill = value), color = 'white') +
    labs(title = as.character(i))
  plot_list[[i - 2018]] <- plot
}
grid.arrange(grobs = plot_list, nrow = 2, ncol = 2)

##-- factor score (which i did use)
##---- (find the best n)
wss <- vector()
max_clusters <- 10
for (k in 1:max_clusters) {
  kmeans_result <- kmeans(data[which(data$Year == 2015), 13], centers = k)
  wss[k] <- kmeans_result$tot.withinss
}
plot(1:max_clusters, wss, type = "b", pch = 19, frame = FALSE, xlab = "
    Number of Clusters", ylab = "Within-cluster Sum of Squares")

##---- (2015~2018)
plot_list <- list()
for (i in 2015:2018) {
  result <- cbind(data[which(data$Year == i), 'Country'], kmeans(data[
      which(data$Year == i), 13], centers = 4)$cluster)
  colnames(result) <- c('region', 'value')
```

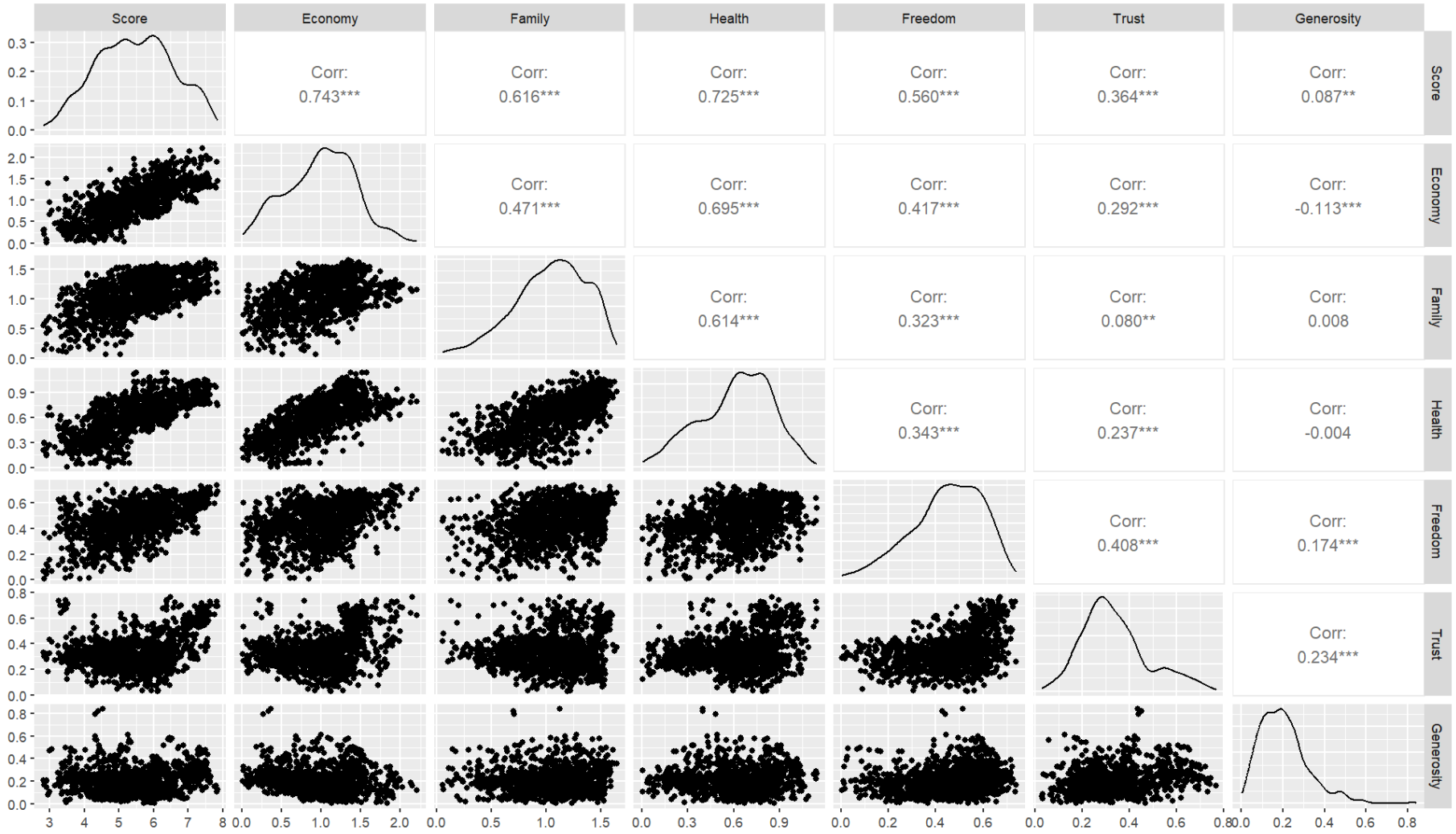
```r
  map_data <- left_join(world, as.data.frame(result), by = 'region',
      relationship = 'many-to-many')

  plot <- ggplot(map_data, aes(long, lat, group = group)) +
    geom_polygon(aes(fill = value), color = 'white') +
    labs(title = as.character(i))
  plot_list[[i - 2014]] <- plot
}
grid.arrange(grobs = plot_list, nrow = 2, ncol = 2)

##---- (2019~2022)
plot_list <- list()
for (i in 2019:2022) {
  result <- cbind(data[which(data$Year == i), 'Country'], kmeans(data[
      which(data$Year == i), 13], centers = 4)$cluster)
  colnames(result) <- c('region', 'value')
  map_data <- left_join(world, as.data.frame(result), by = 'region',
      relationship = 'many-to-many')

  plot <- ggplot(map_data, aes(long, lat, group = group)) +
    geom_polygon(aes(fill = value), color = 'white') +
    labs(title = as.character(i))
  plot_list[[i - 2018]] <- plot
}
grid.arrange(grobs = plot_list, nrow = 2, ncol = 2)

##---- (2022)
plot_list <- list()
result <- cbind(data[which(data$Year == 2022), 'Country'], kmeans(data[
    which(data$Year == 2022), 13], centers = 4)$cluster)
colnames(result) <- c('region', 'value')
map_data <- left_join(world, as.data.frame(result), by = 'region',
    relationship = 'many-to-many')
plot <- ggplot(map_data, aes(long, lat, group = group)) +
  geom_polygon(aes(fill = value), color = 'white') +
  scale_fill_manual(values = 1:4) +
  labs(title = 2022)
plot
```

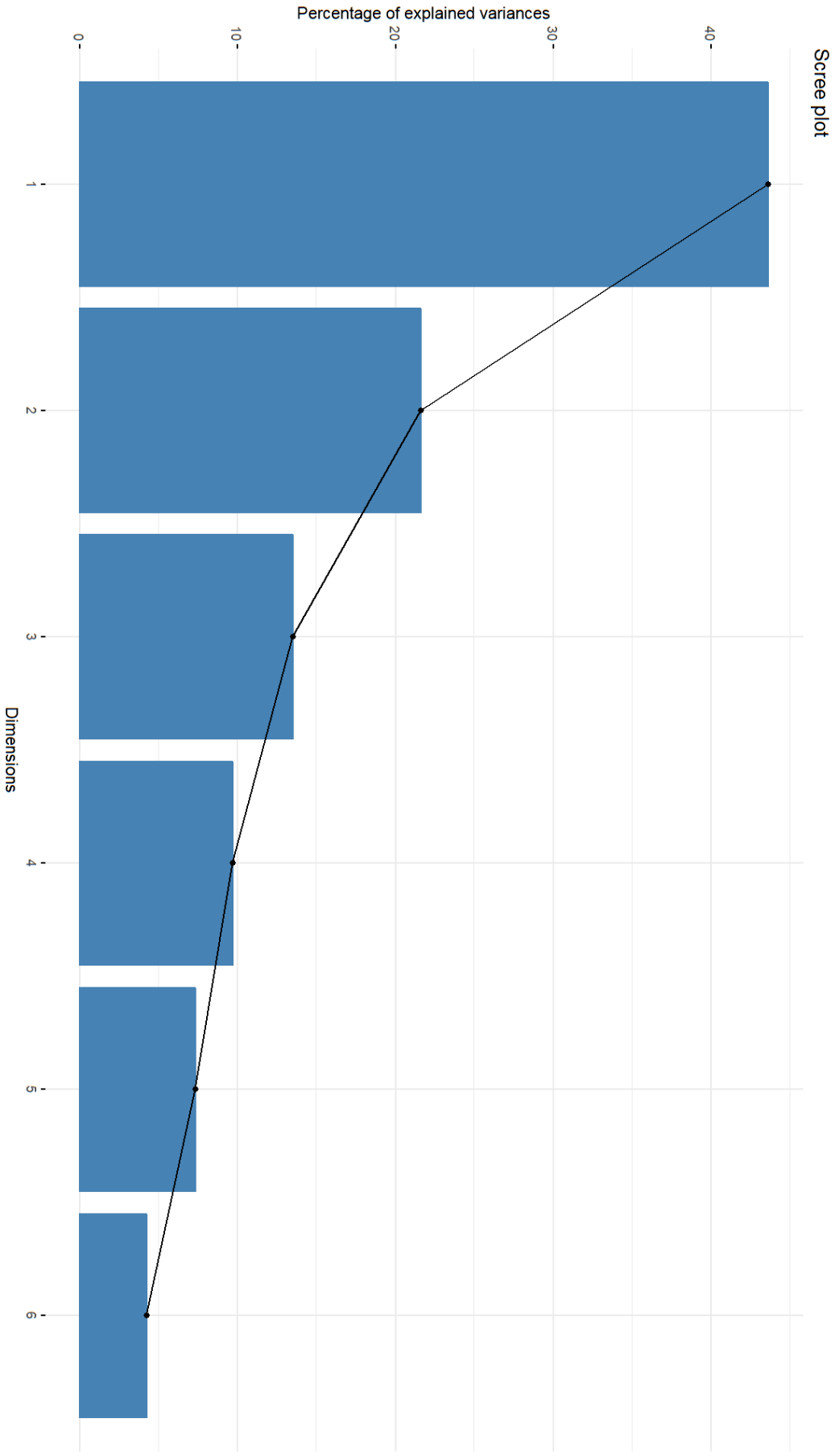
```
180
181 cluster.avg <- data.frame()
182 for (j in 1:4) {
183   colMeans(data[which(result[, 'value'] == 2), c(5, 13)])
184   cluster.avg <- rbind(cluster.avg, colMeans(data[which(result[, 'value']
        == j), c(5, 13)]))
185 }
186 colnames(cluster.avg) <- c('Score', 'fa.score.RC1', 'fa.score.RC3', 'fa.
      score.RC2')
187
188 plot(cluster.avg[, 1], type = "n", ylim = range(cluster.avg), xlab = '',
      ylab = 'value', xaxt = 'n')
189 for (i in 1:4) {
190   points(1:4, cluster.avg[i, ], pch = 16, col = i, cex = 1.5)
191 }
192 text(1:4, par("usr")[3] - 1, colnames(cluster.avg), xpd = TRUE)
193 legend("topright", legend = 1:4, col = 1:4, pch = 16, title = 'Cluster')
```

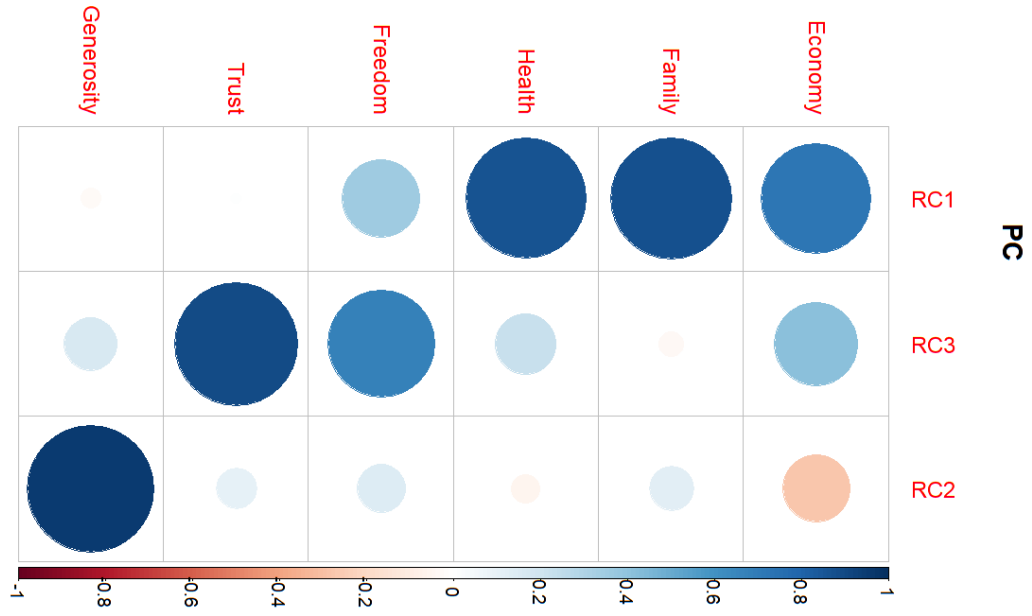
7.2 高清附图

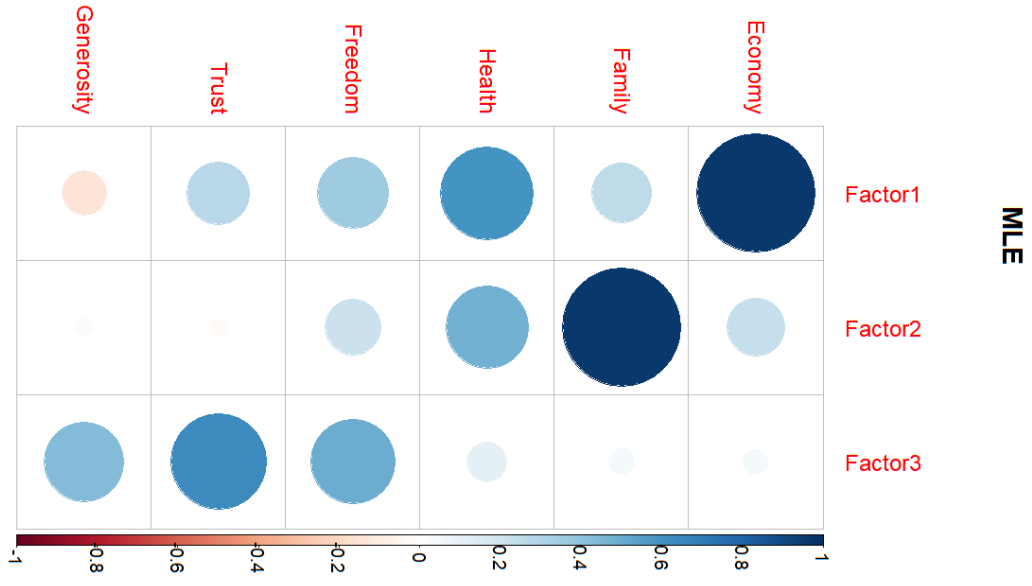

value
0
2
4
6
8
as.factor(Year)
2015
2016
2017
2018
2019
2020
2021
2022
variable
Score
Economy
Family
Health
Freedom
Trust
Generosity

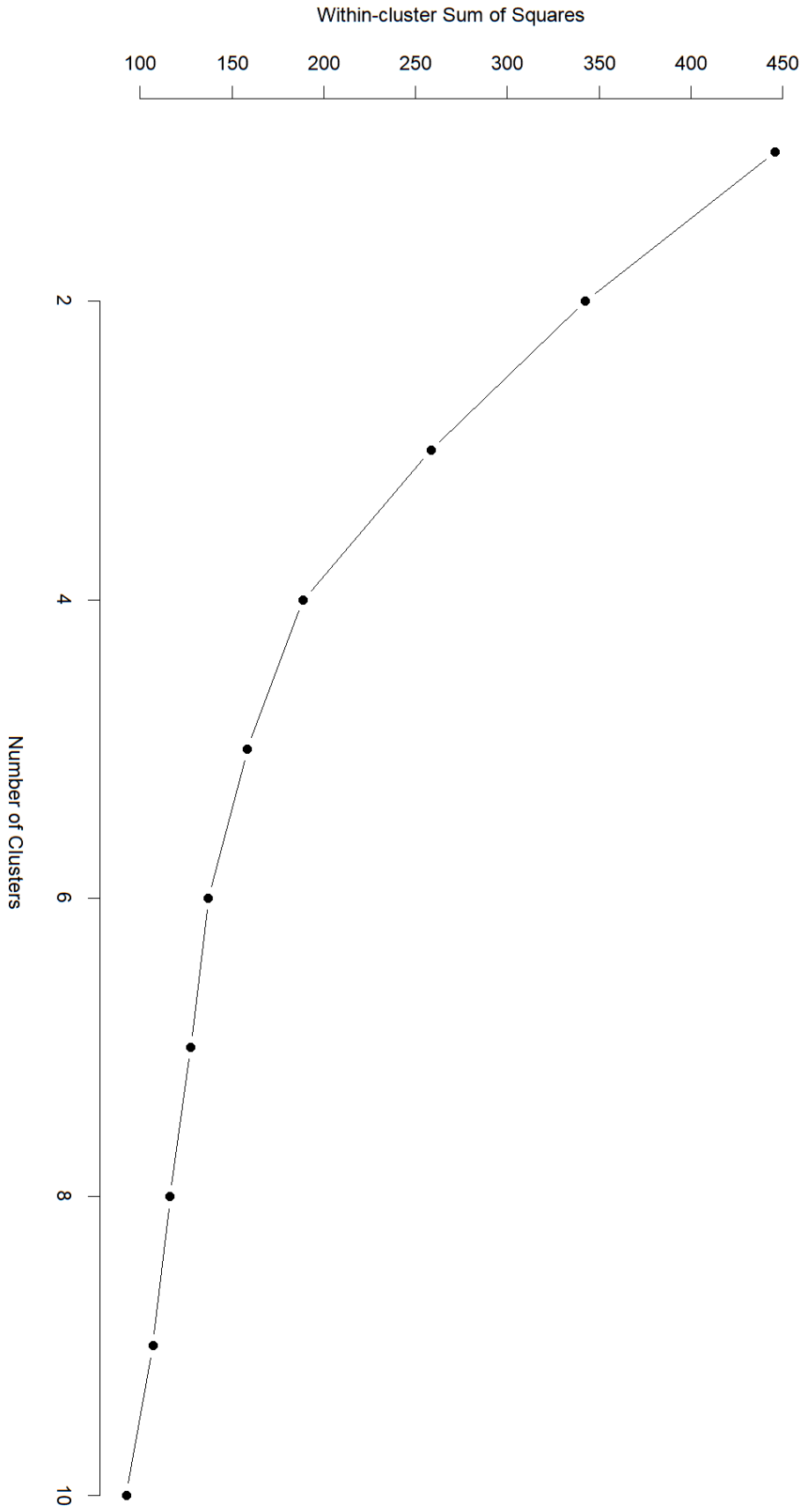

value
0
2
4
6
8
as.factor(Year)
2015
2016
2017
2018
2019
2020
2021
2022
variable
Score
Economy
Family
Health
Freedom
Trust
Generosity

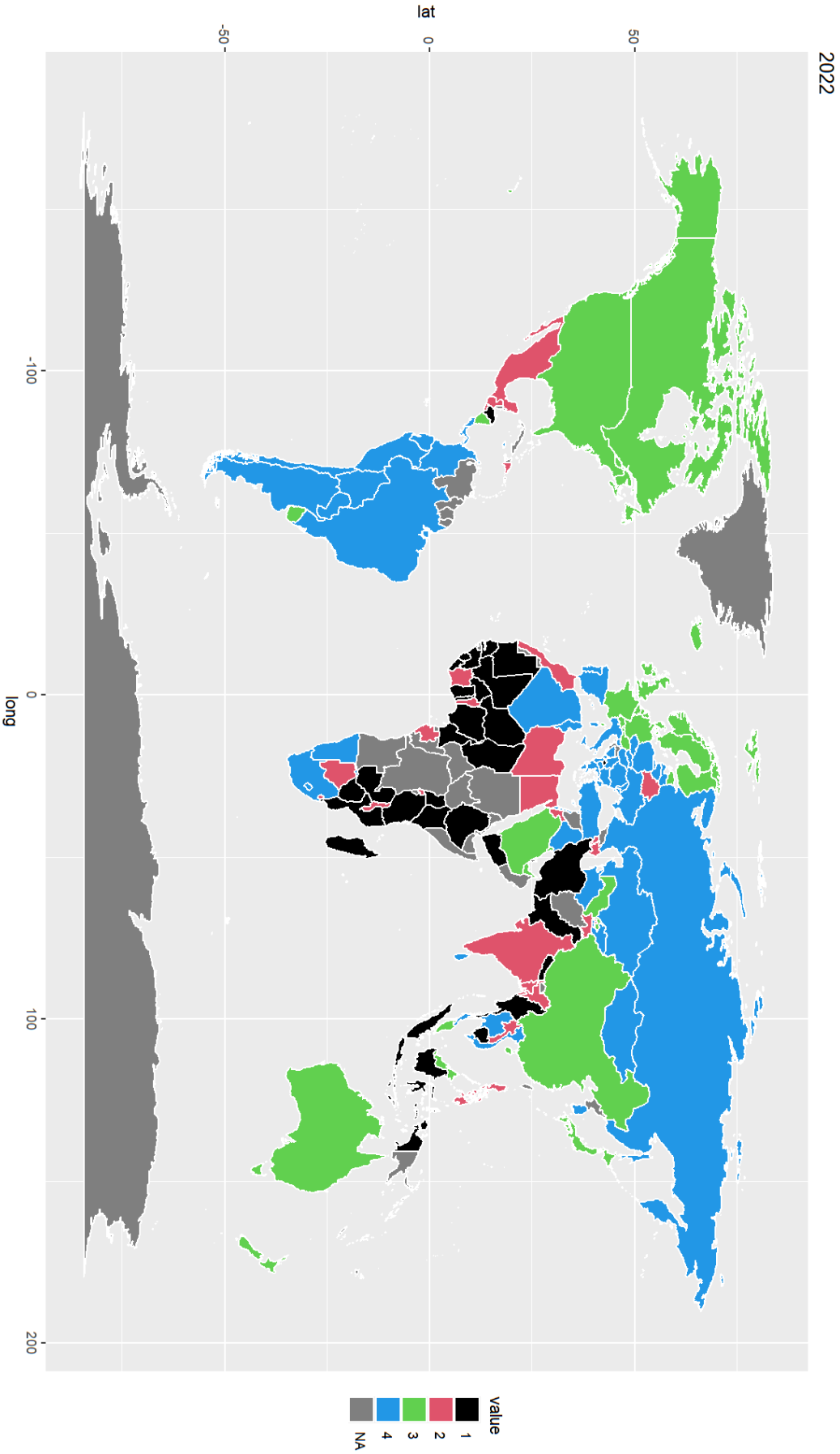

Score
Economy
Family
Health
Freedom
Trust
Generosity
Corr: 0.743***
Corr: 0.616***
Corr: 0.725***
Corr: 0.560***
Corr: 0.364***
Corr: 0.087**
Corr: 0.471***
Corr: 0.695***
Corr: 0.417***
Corr: 0.292***
Corr: -0.113***
Corr: 0.614***
Corr: 0.323***
Corr: 0.080**
Corr: 0.008
Corr: 0.343***
Corr: 0.237***
Corr: -0.004
Corr: 0.408***
Corr: 0.174***
Corr: 0.234***

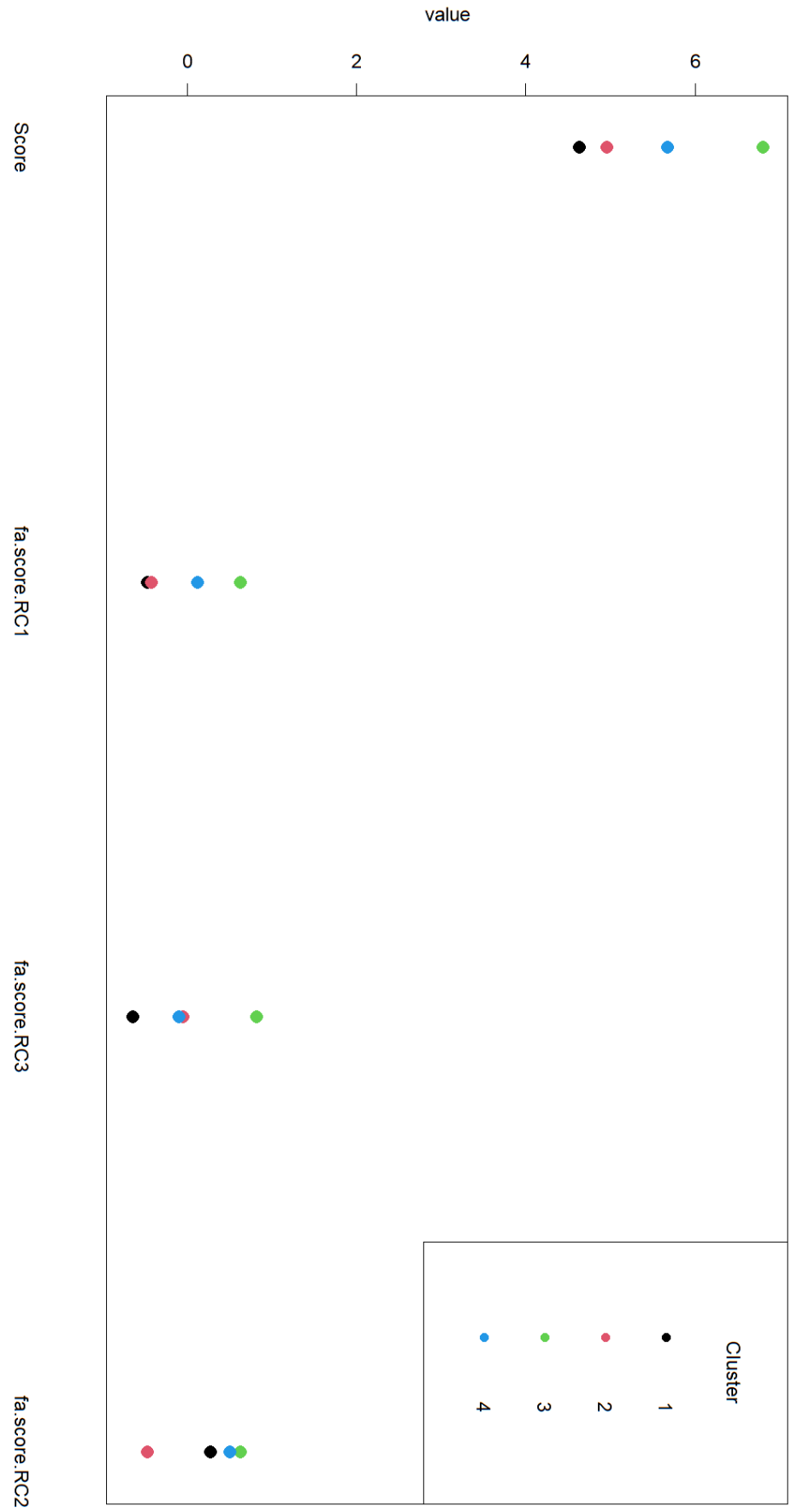

Scree plot
Percentage of explained variances
0
10
20
30
40
Dimensions
1
2
3
4
5
6

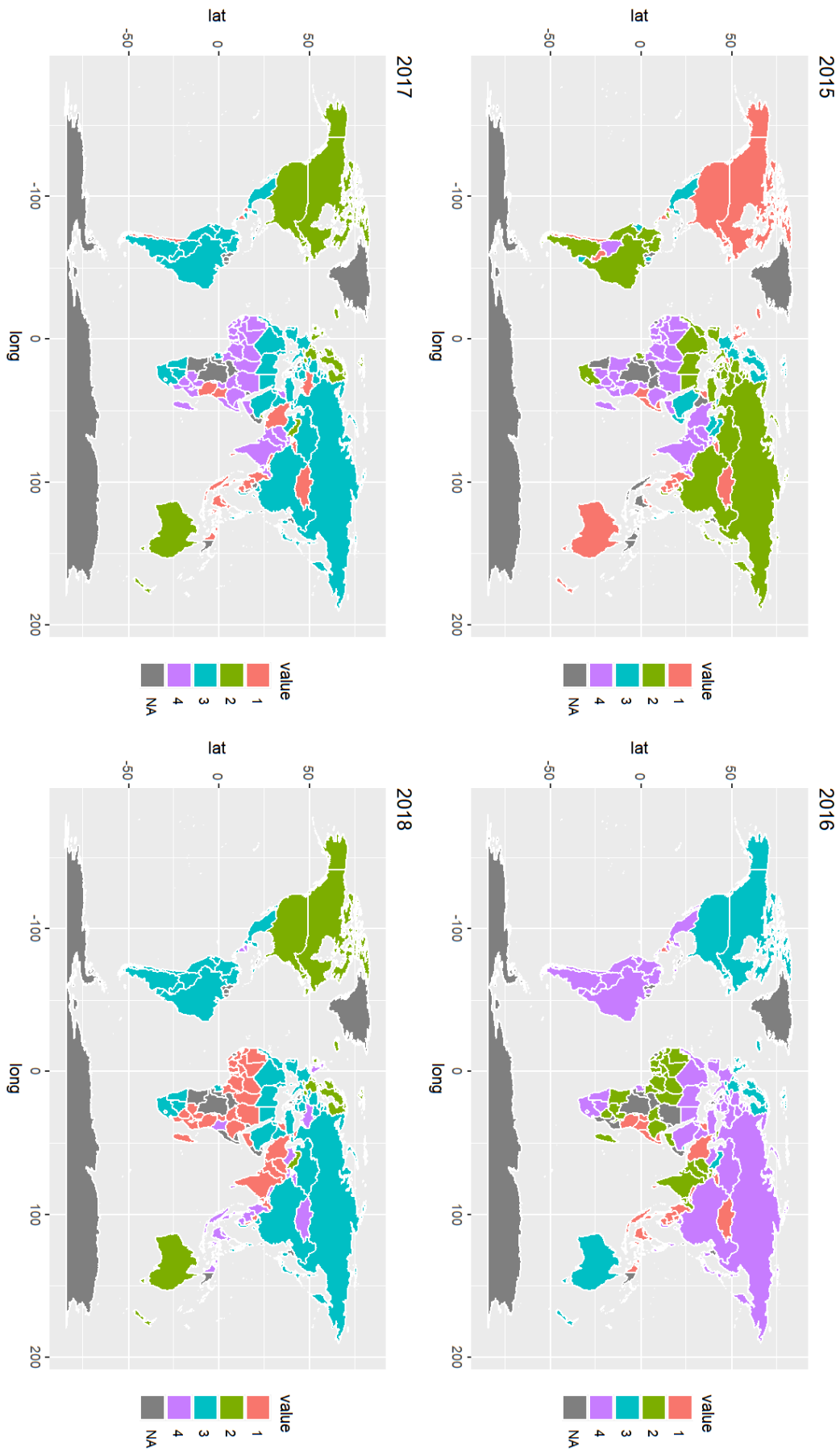

PC
Economy
Family
Health
Freedom
Trust
Generosity
RC1
RC3
RC2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1

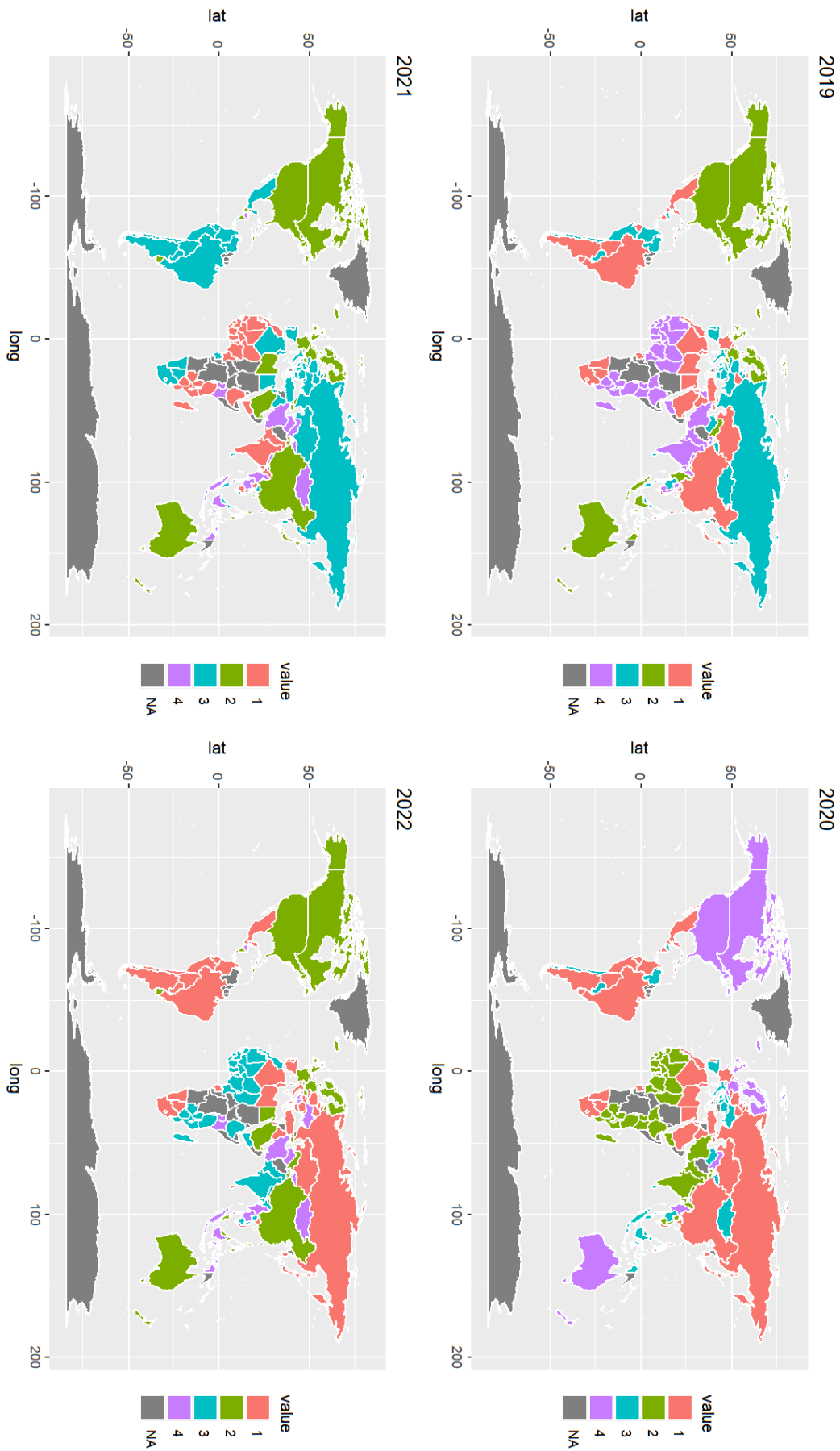

MLE
Economy
Family
Health
Freedom
Trust
Generosity
Factor1
Factor2
Factor3
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1


Within-cluster Sum of Squares
100
150
200
250
300
350
400
450
Number of Clusters
2
4
6
8
10


lat
-50
0
50
2022
long
-100
0
100
200
value
NA
4
3
2
1


value
0
2
4
6
Score
fa.score.RC1
fa.score.RC3
fa.score.RC2
Cluster
1
2
3
4


2015
2016
2017
2018
lat
long
value
1
2
3
4
NA


2019
2020
2021
2022
lat
long
value
1
2
3
4
NA
-50
0
50
-100
0
100
200