PCA

Aissata Bah Brice Laurent Linh Vu

2023-12-09

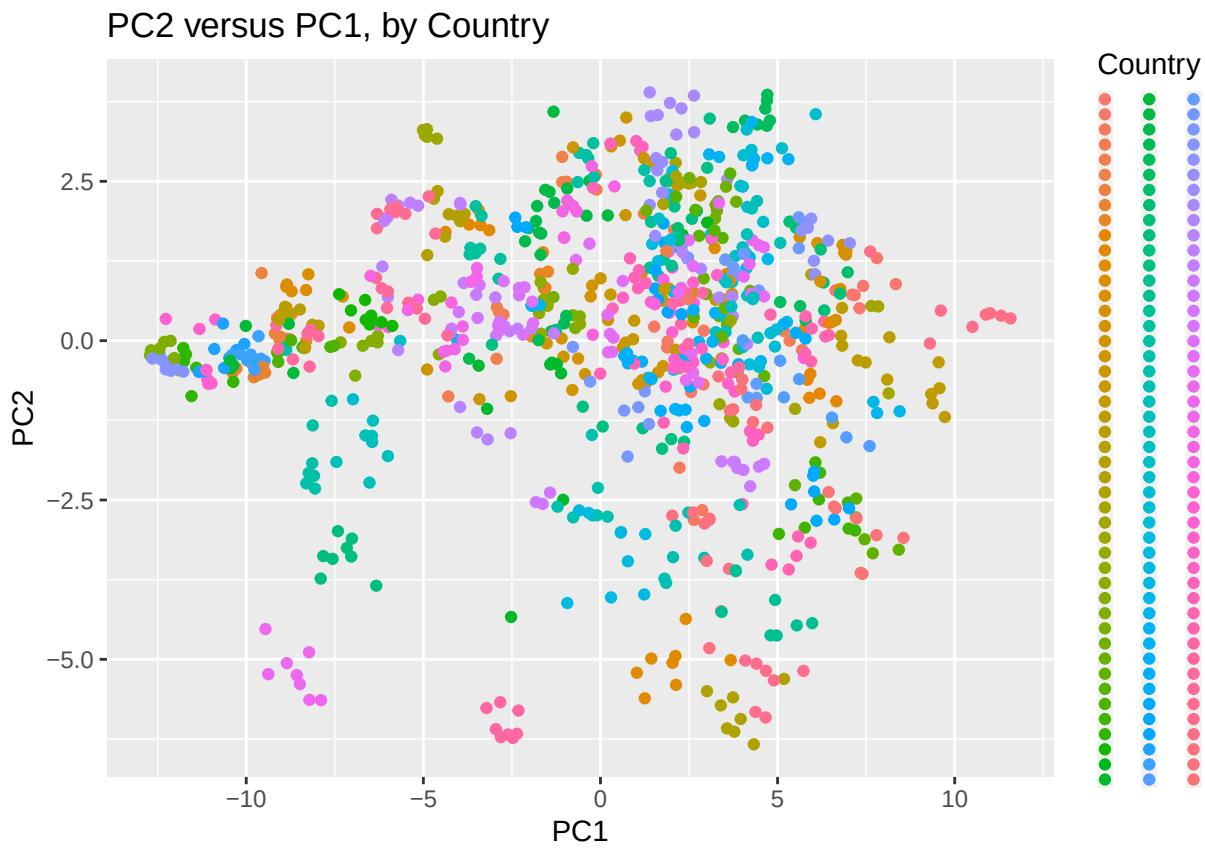
```
# Load Libraries
library(dplyr)
library(ggplot2)
library(lme4)
library(broom.mixed)
library(lmerTest)
library(tibble)
library(ggpubr)
library(knitr)

data = read.csv("./data/data_clean.csv")
```

```
# Impute missing values with of values in the same region
selected_columns_pca = select(data, c(starts_with("x"), "region"))
selected_columns_pca = selected_columns_pca %>%
  rename(x8.7=x8.7..due.process.of.the.law.and.rights.of.the.accused)
selected_columns_pca_imputed = selected_columns_pca %>%
  group_by(region) %>%
  mutate_all(~ ifelse(is.na(.), mean(., na.rm = TRUE), .))
```

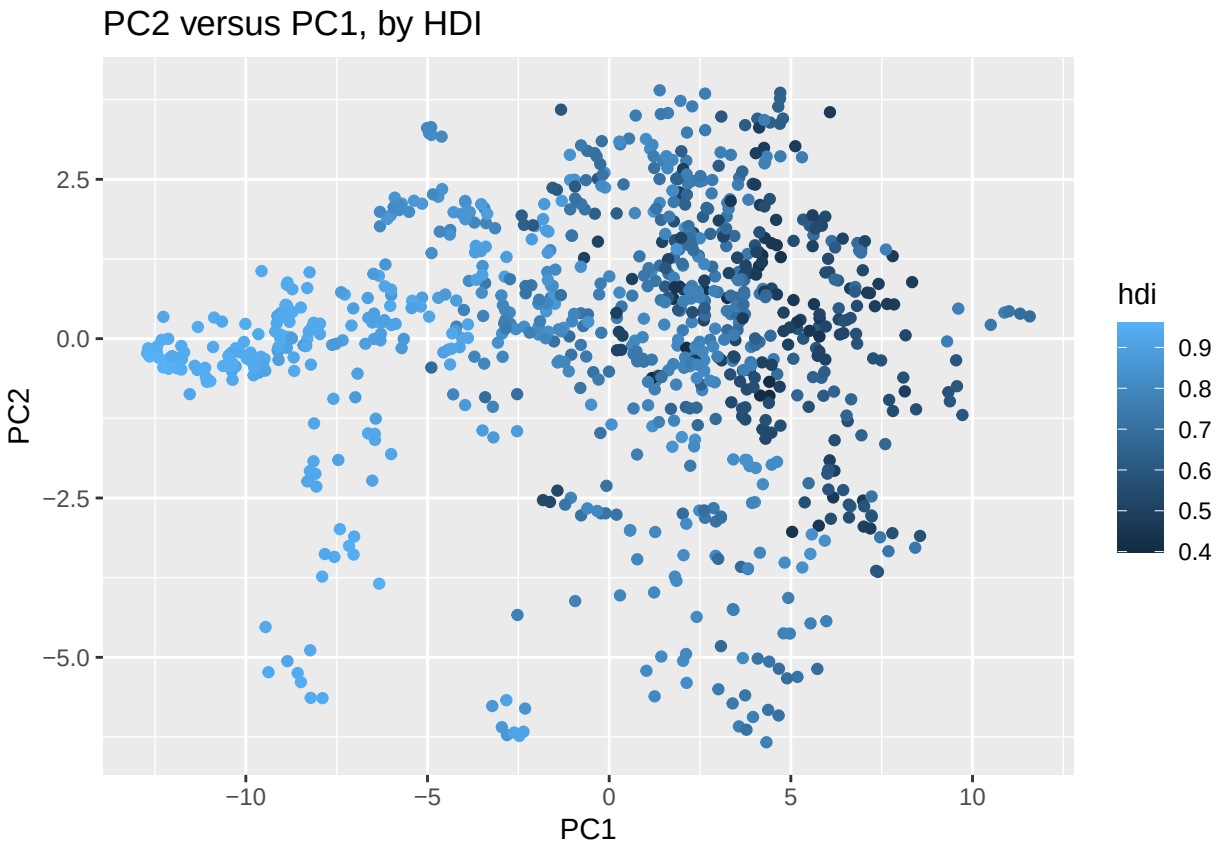
```
## `mutate_all()` ignored the following grouping variables:
## * Column `region`
## i Use `mutate_at(df, vars(-group_cols()), myoperation)` to silence the message.
```

```
selected_columns_pca_imputed = selected_columns_pca_imputed[, -c(1,ncol(selected_columns_pca_imputed))]

# Extract Principle Components of Rule of Law Variables
pca = prcomp(selected_columns_pca_imputed, scale. = TRUE)
variance_explained = (pca$sdev^2) / sum(pca$sdev^2)
```

```
# Visualize Principle Components
PC1and2 = as.data.frame(cbind(pca$x[,c(1,2)], data[, c("country", "year", "hdi")]))
PC1and2$PC1 = as.numeric(PC1and2$PC1)
PC1and2$PC2 = as.numeric(PC1and2$PC2)
PC1and2$Country = PC1and2$country
p1 = ggplot(data = PC1and2,aes(x=PC1, y=PC2)) + geom_point(aes(color = Country)) + theme(legend.text = e
p2 = ggplot(data = PC1and2,aes(x=PC1, y=PC2)) + geom_point(aes(color = hdi)) + labs(title="PC2 versus PC

p1
```

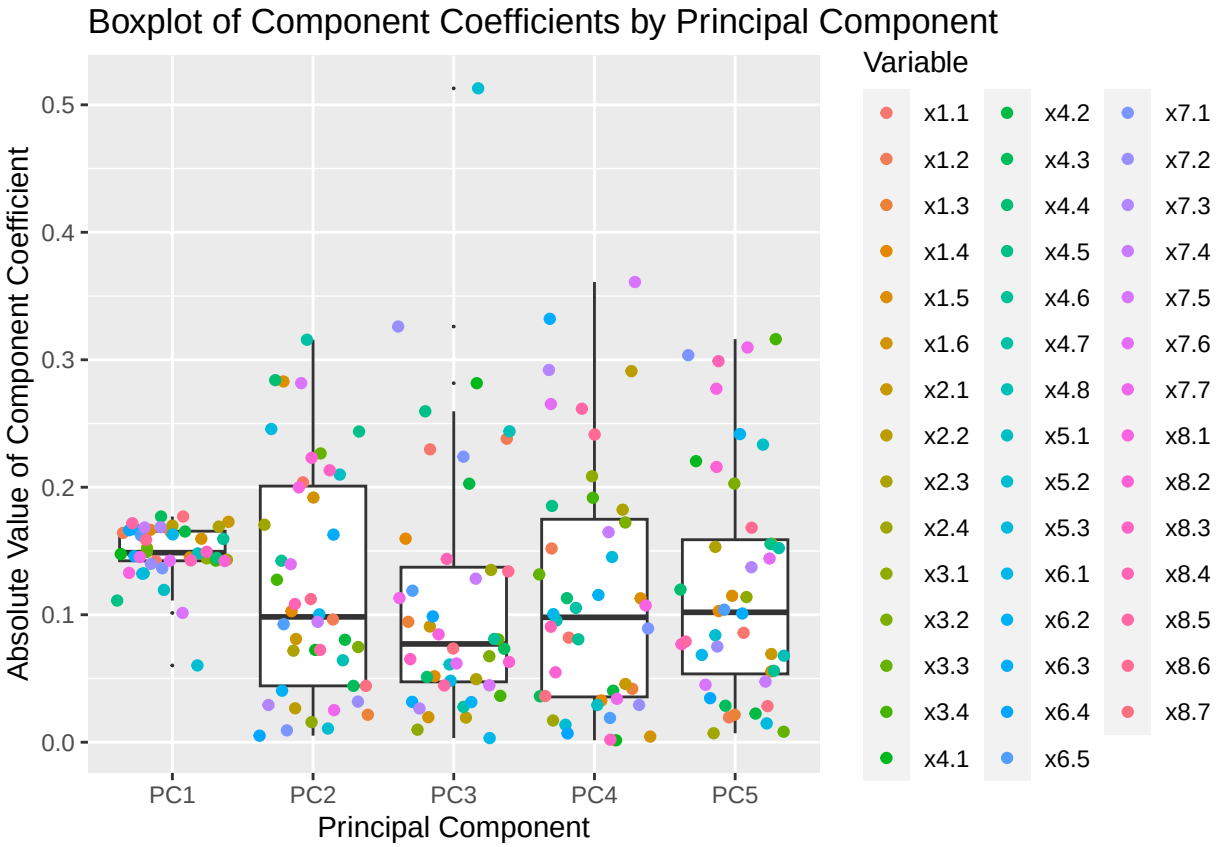



```
p2
```


PC2 versus PC1, by HDI
2.5
0.0
−2.5
−5.0
PC2
−10
−5
0
5
10
PC1
hdi
0.9
0.8
0.7
0.6
0.5
0.4

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.7969102	0.0114316	69.711233	0.0000000
years_from_2000	-0.0007155	0.0006028	-1.186964	0.2355544
PC1	-0.0140302	0.0004144	-33.858146	0.0000000
PC2	-0.0017414	0.0010436	-1.668669	0.0955301
PC3	0.0100197	0.0016650	6.017831	0.0000000
PC4	-0.0114123	0.0016648	-6.854863	0.0000000
PC5	0.0263945	0.0020210	13.059999	0.0000000
regionEastern Europe & Central Asia	0.0240525	0.0070248	3.423964	0.0006449
regionEU + EFTA + North America	0.0231869	0.0064870	3.574354	0.0003697
regionLatin America & Caribbean	-0.0444531	0.0077871	-5.708545	0.0000000
regionMiddle East & North Africa	-0.0143941	0.0079197	-1.817522	0.0694689
regionSouth Asia	-0.0760998	0.0094195	-8.078933	0.0000000
regionSub-Saharan Africa	-0.1786519	0.0063769	-28.015510	0.0000000

```r
# Determine importance of each variable
TopPCs_loadings_abs <- abs(pca$rotation[,seq(1,5)])
TopPCs_loadings_abs_long <- as.data.frame(as.table(TopPCs_loadings_abs))
colnames(TopPCs_loadings_abs_long) <- c("Variable", "Principal_Component", "Loading")
ggplot(TopPCs_loadings_abs_long, aes(x = Principal_Component, y = Loading)) +
  geom_boxplot(outlier.size = 0) +
  geom_jitter(aes(color = Variable)) +
  labs(title = "Boxplot of Component Coefficients by Principal Component",
       x = "Principal Component",
       y = "Absolute Value of Component Coefficient")
```




```
rowSums(TopPCs_loadings_abs)[order(-rowSums(TopPCs_loadings_abs))]
```

```
##      x4.5      x7.5      x8.6      x3.4      x8.1      x8.4      x6.3      x7.1
## 0.9553316 0.8339229 0.8147289 0.8141922 0.8019538 0.7843124 0.7735867 0.7625431
##      x7.6      x1.1      x4.7      x4.1      x4.4      x1.5      x8.2      x5.2
## 0.7531998 0.7434732 0.7335235 0.7236037 0.7125175 0.7074593 0.6989774 0.6815854
##      x1.2      x3.2      x2.2      x7.3      x5.1      x8.5      x7.7      x2.3
## 0.6700069 0.6696940 0.6625806 0.6536230 0.6530090 0.6293128 0.6272086 0.6260249
##      x4.8      x7.4      x7.2      x3.3      x6.2      x5.3      x1.4      x4.6
## 0.6195439 0.6034731 0.6021162 0.5964080 0.5959562 0.5861729 0.5646912 0.5195635
##      x4.2      x8.3      x6.5      x3.1      x2.4      x2.1      x6.1      x1.6
## 0.5113497 0.5067469 0.4965893 0.4922770 0.4729180 0.4595358 0.4354793 0.4312512
##      x8.7      x4.3      x1.3      x6.4
## 0.3593491 0.3593013 0.3445003 0.3116982
```

```
TopPCs_loadings_abs_grouped = as.data.frame(TopPCs_loadings_abs)
TopPCs_loadings_abs_grouped$Row_Labels = row.names(TopPCs_loadings_abs)
TopPCs_loadings_abs_grouped = TopPCs_loadings_abs_grouped %>%
  mutate(First_Digit = as.numeric(gsub("\\D*(\\d).*", "\\1", TopPCs_loadings_abs_grouped$Row_Labels))) %
  select(-Row_Labels)
TopPCs_loadings_abs_grouped = TopPCs_loadings_abs_grouped %>%
  group_by(First_Digit) %>%
  summarise(across(everything(), sum, na.rm = TRUE)) %>%
  column_to_rownames(var = "First_Digit")
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(everything(), sum, na.rm = TRUE)`.
## i In group 1: `First_Digit = 1`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```
TopPCs_loadings_abs_grouped = as.data.frame(t(TopPCs_loadings_abs_grouped)) %>% rename("1 (Constraints

df = as.data.frame(colSums(TopPCs_loadings_abs_grouped)[order(-colSums(TopPCs_loadings_abs_grouped))])

names(df) = "Sum of Absolute Values of Component Coefficients"

kable(df)
```

	Sum of Absolute Values of Component Coefficients
4 (Fundamental Rights)	5.134735
7 (Civil Justice)	4.836087
8 (Criminal Justice)	4.595381
1 (Constraints on Government Powers)	3.461382
6(Regulatory Enforcement)	2.613310
3 (Open Government)	2.572571
2 (Absence of Corruption)	2.221059
5 (Order and Security)	1.920767

The squared loadings represent the proportion of the variance of the variable that is accounted for by the corresponding principal component.