PCA_Clustering

RJ

8/15/2020

PCA and Clustering

This document summarize the learning activity using the dataset provided in Planning Public Policy in Argentina. The original learning method is clearly explained in datacamp. A source in kaggle might follow the original, added with the notebook owner's improvement. The kaggle source will be the learning source for this exercise.

The provided data shows the economical dan social indicators of each province. Indicators are highly correlated, which need to be confirmed by research on socioeconomical. According to writer perspective, considering the PCA explanation on Python Machine Learning, feature reduction through extraction will help in clustering process by maintaining the relevant information in the original data. >PCA helps us to identify patterns in data based on the correlationbetween features - Python Machine Learning

Preparation

Call the required library and dataset

```
library(tidyverse)
```

```
## -- Attaching packages -----------------------------------------------------------------------

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.2     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(FactoMineR) # For PCA preprocessing
library(factoextra) # For PCA data visualization
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(ggrepel)

argentina <- read.csv(choose.files())
head(argentina)
```

```
##       province       gdp illiteracy   poverty deficient_infra school_dropout
## 1 Buenos Aires 292689868    1.38324  8.167798        5.511856      0.7661682
## 2    Catamarca   6150949    2.34414  9.234095       10.464484      0.9519631
## 3      CÃ³rdoba  69363739    2.71414  5.382380       10.436086      1.0350558
## 4   Corrientes   7968013    5.60242 12.747191       17.438858      3.8642652
## 5        Chaco   9832643    7.51758 15.862619       31.479527      2.5774621
## 6       Chubut  17747854    1.54806  8.051752        8.044618      0.5863094
##   no_healthcare birth_mortal      pop movie_theatres_per_cap doctors_per_cap
## 1       48.7947          4.4 15625084           6.015968e-06     0.004835622
## 2       45.0456          1.5   367828           5.437324e-06     0.004502104
## 3       45.7640          4.8  3308876           1.118204e-05     0.010175359
## 4       62.1103          5.9   992595           4.029841e-06     0.004495288
## 5       65.5104          7.5  1055259           2.842904e-06     0.003604802
## 6       39.5473          3.0   509108           1.571376e-05     0.004498063
```

The datasets has 11 variables which includes: 1. **province**: Argentina's provinces 2. **gdp**: a measure of the size of a province's economy 3. **illiteracy**: Adult illiteracy is defined as the percentage of the population aged 15 years and over who cannot both read and write with understanding a short simple statement on his/her everyday life. According to UNESCO 4. **poverty**: the ratio of the number of people (in a given age group) whose income falls below the poverty line 5. **deficient_infra**: 6. **school_dropout**: rate of school drop out 7. **no_healthcare**: rate of people wihtout healthcare 8. **birth_mortal**: birth mortality rate 9. **pop**: population 10. **movie_theatres_per_cap**: 11. **doctors_per_cap**: ratio of doctors and population

To measure the province economic condition, GDP per capita is more relevant. PCA feature extraction will be performed by using the factominer package. As the factominer PCA process require the input format in matrix, the current individual data point will be casted.

```
argentina_matrix <- argentina %>%
  mutate(gpd_per_capita = gdp/pop) %>%
  select_if(is.numeric) %>%
  as.matrix()

head(argentina_matrix)
```

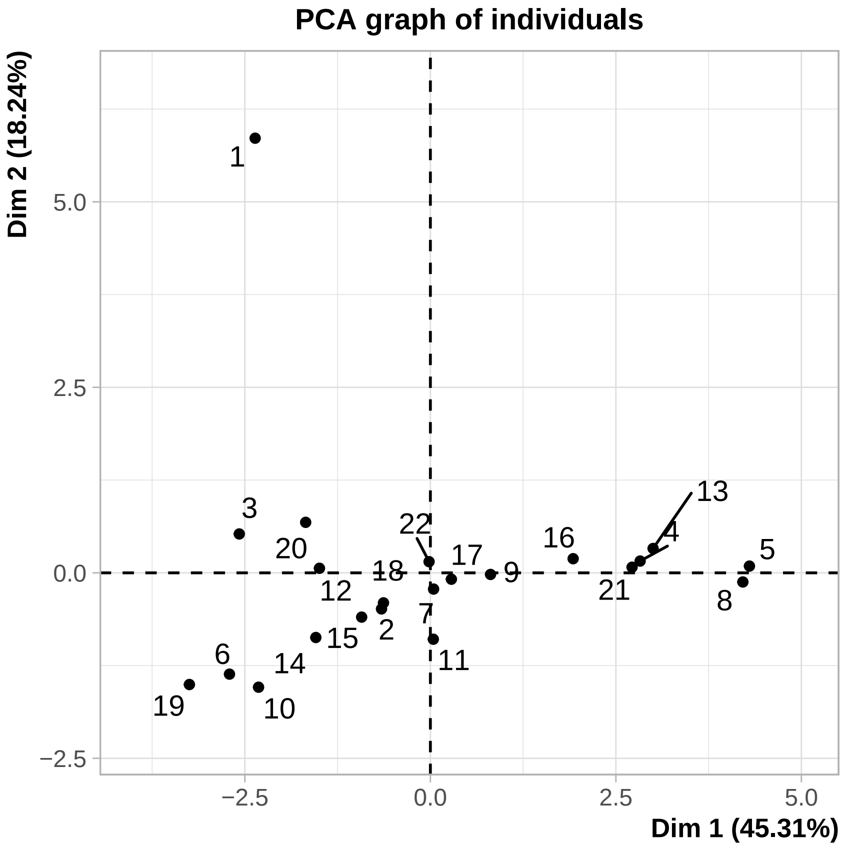
```
##            gdp illiteracy   poverty deficient_infra school_dropout
## [1,] 292689868    1.38324  8.167798        5.511856      0.7661682
## [2,]   6150949    2.34414  9.234095       10.464484      0.9519631
## [3,]  69363739    2.71414  5.382380       10.436086      1.0350558
## [4,]   7968013    5.60242 12.747191       17.438858      3.8642652
## [5,]   9832643    7.51758 15.862619       31.479527      2.5774621
## [6,]  17747854    1.54806  8.051752        8.044618      0.5863094
##      no_healthcare birth_mortal      pop movie_theatres_per_cap doctors_per_cap
## [1,]       48.7947          4.4 15625084           6.015968e-06     0.004835622
## [2,]       45.0456          1.5   367828           5.437324e-06     0.004502104
## [3,]       45.7640          4.8  3308876           1.118204e-05     0.010175359
## [4,]       62.1103          5.9   992595           4.029841e-06     0.004495288
## [5,]       65.5104          7.5  1055259           2.842904e-06     0.003604802
## [6,]       39.5473          3.0   509108           1.571376e-05     0.004498063
##      gpd_per_capita
## [1,]      18.732051
## [2,]      16.722352
## [3,]      20.962931
## [4,]       8.027456
```

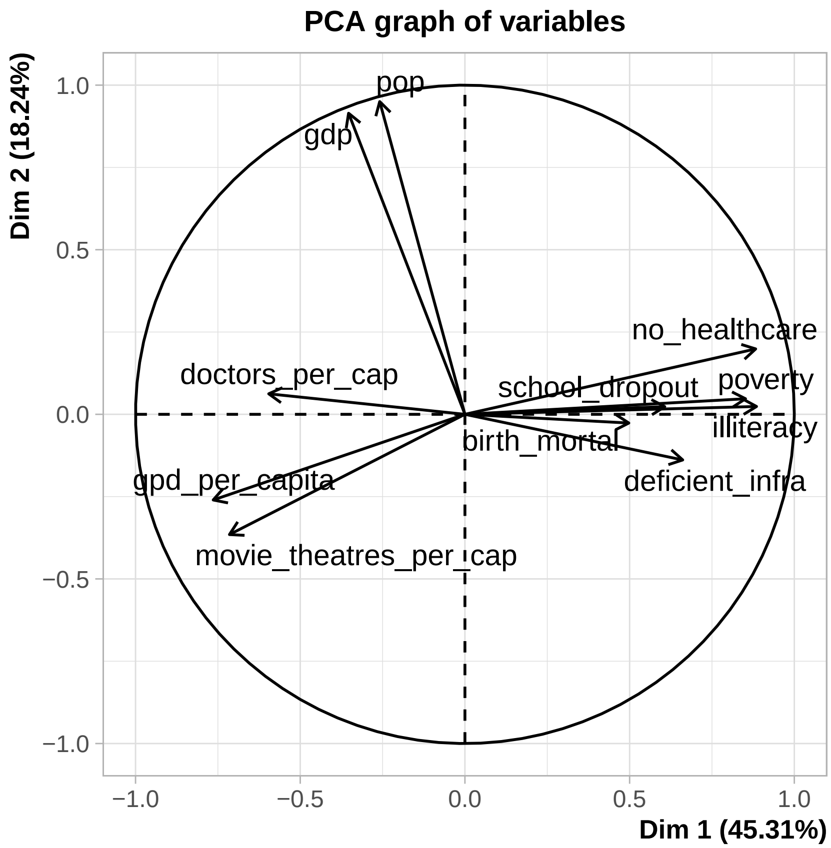
```
## [5,]       9.317753
## [6,]      34.860686
```

Feature Reduction

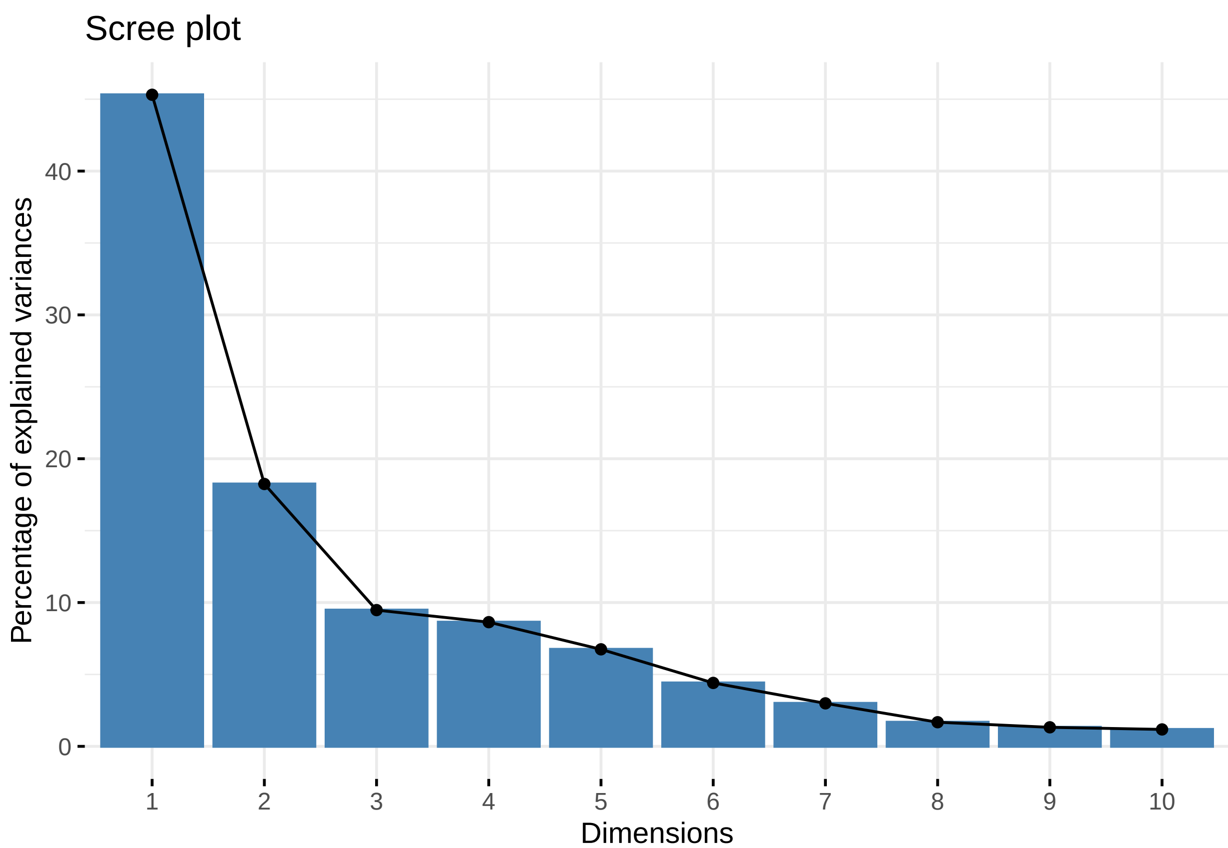
Factominer::PCA function ease the research by automatically shows the PCA biplot. The first two principal components represent around 63% of the original data variance. The cummulative

```
argentina_pca <- PCA(argentina_matrix, scale.unit = T)
```






```r
fviz_eig(argentina_pca)
```



Clustering

At this point, clustering is performed by using 2 first principal component.

```
argentina_component <- tibble(pca_1 = argentina_pca$ind$coord[,1],
                              pca_2 = argentina_pca$ind$coord[,2])

# Clustering via kmeans algorithm
argentina_kmeans <- kmeans(argentina_component, centers = 4, nstart = 20, iter.max = 50)
argentina_kmeans
```

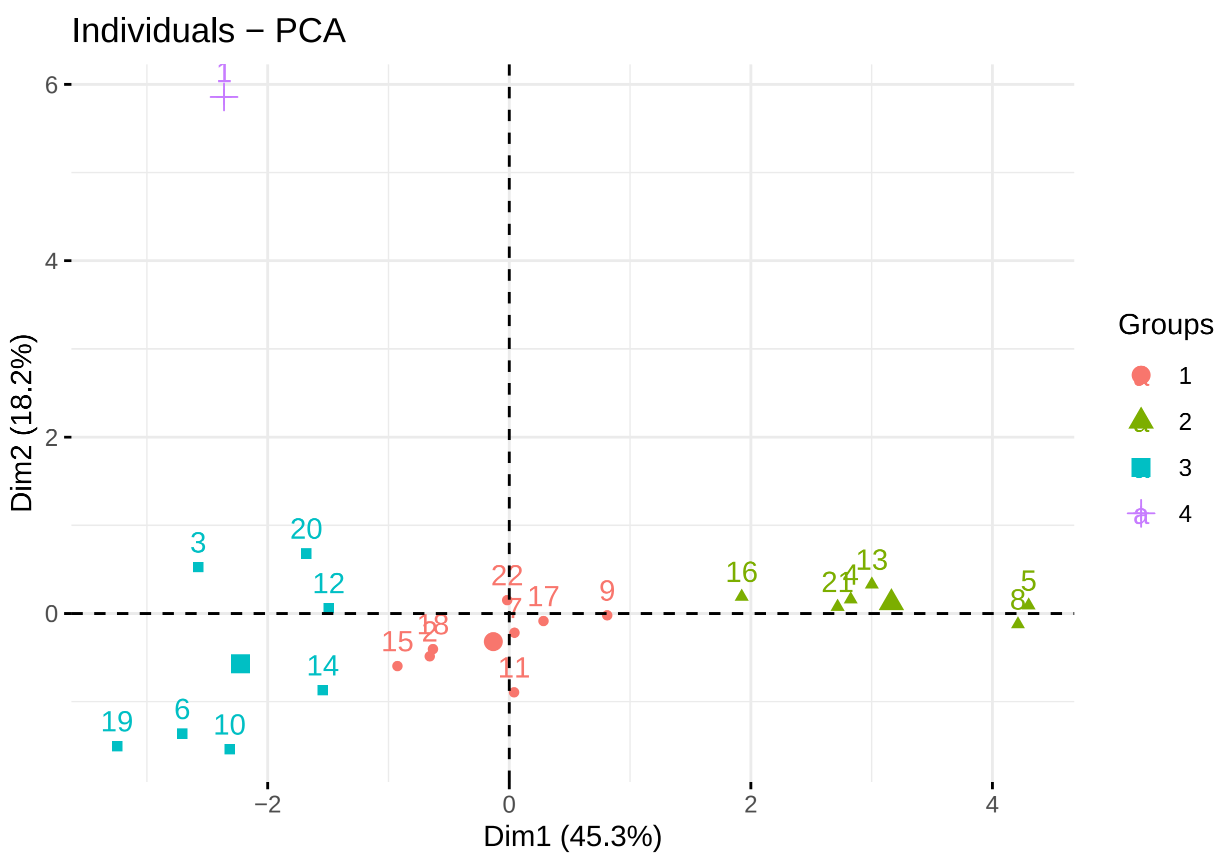
```
## K-means clustering with 4 clusters of sizes 8, 6, 7, 1
##
## Cluster means:
##        pca_1      pca_2
## 1 -0.1320515 -0.3199319
## 2  3.1637648  0.1200775
## 3 -2.2235295 -0.5740342
## 4 -2.3614699  5.8572297
##
## Clustering vector:
##  [1] 4 1 3 2 2 3 1 2 1 3 1 3 2 3 1 2 1 1 3 3 2 1
##
## Within cluster sum of squares by cluster:
## [1] 3.109136 4.375350 8.403846 0.000000
```

```
##  (between_SS / total_SS =  89.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Visualize the clustering result

1. In the selected principal component

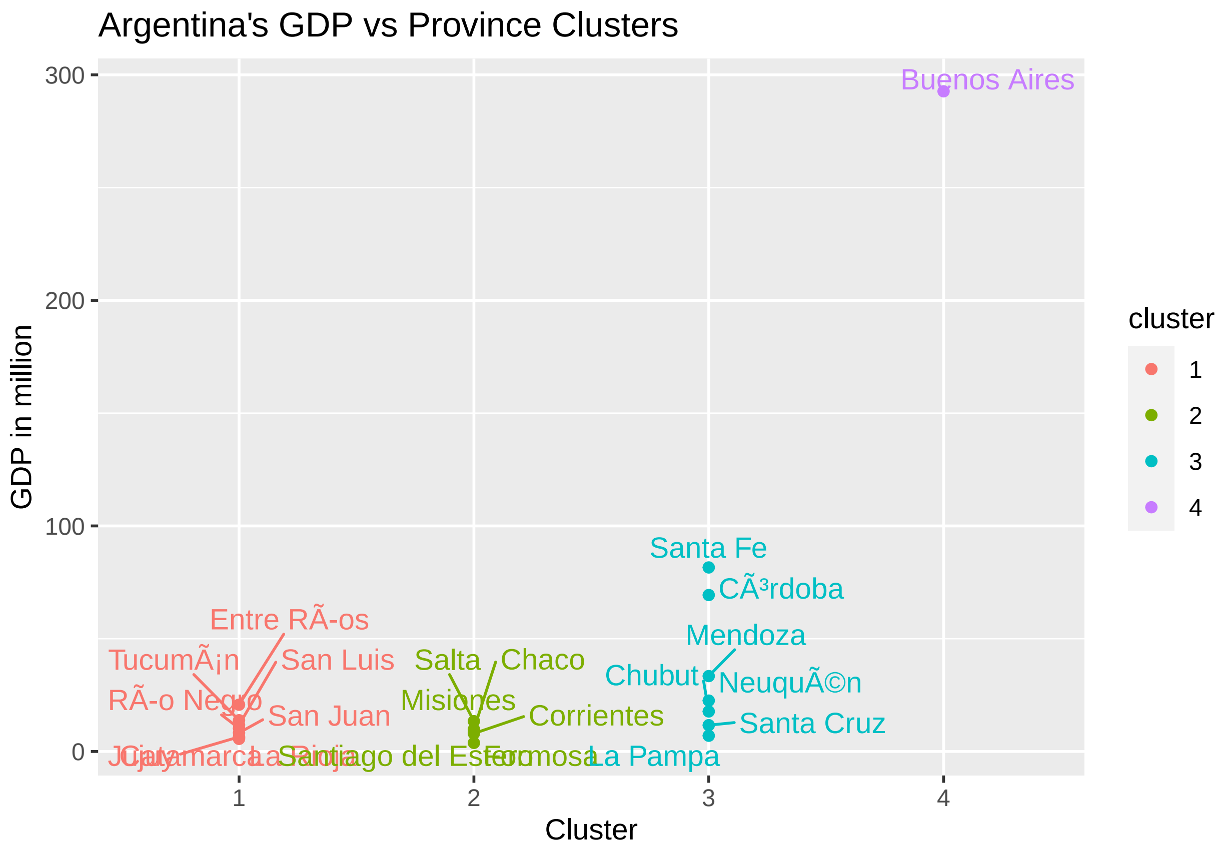
```
fviz_pca_ind(argentina_pca,
             habillage = as.factor(argentina_kmeans$cluster))
```



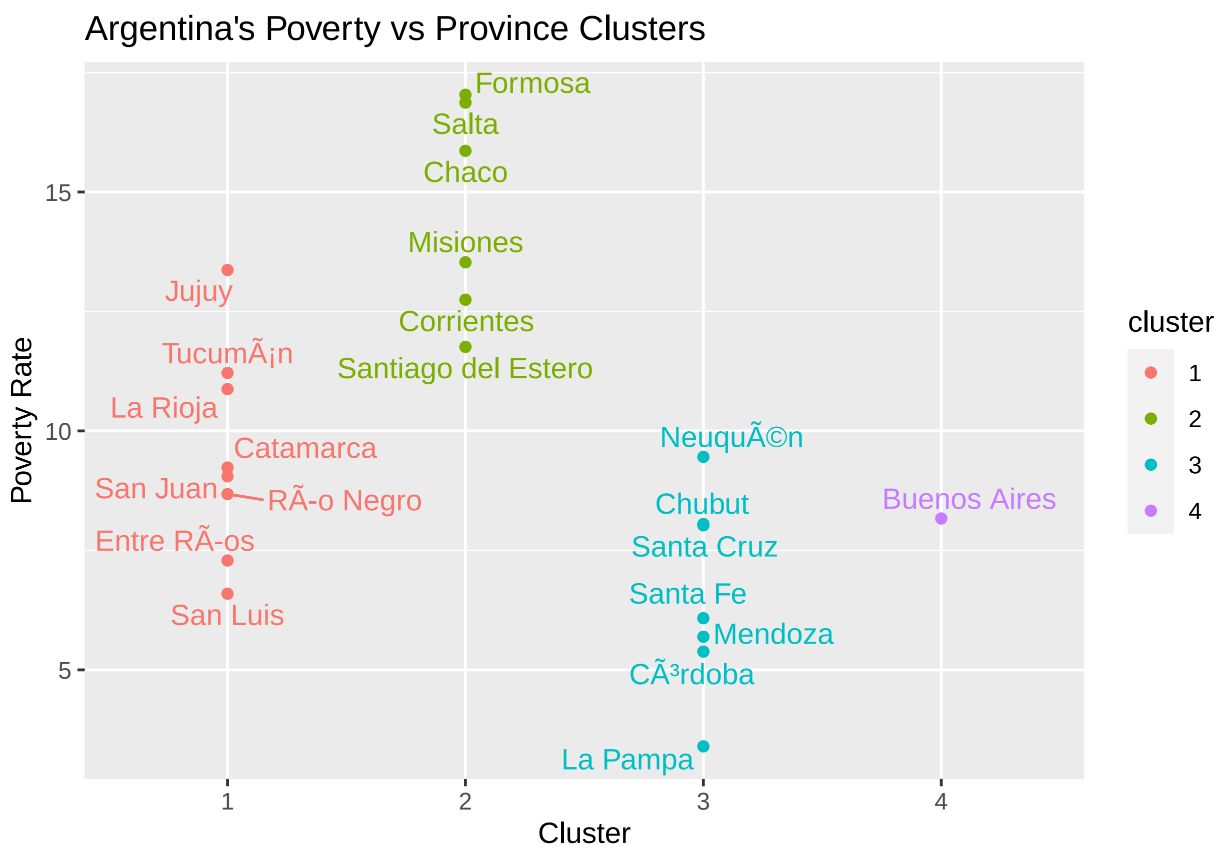
2. In the initial original data As can be seen in PCA biplot, in PCA-2, GDP and Population are in the same direction. Plotting will be performed by using GDP, Population and GPD per capita information.

```
argentina %>%
  mutate(cluster = as.factor(argentina_kmeans$cluster)) %>%
  ggplot(aes(x = cluster,
             y = gdp/1000000,
             color = cluster)) +
  geom_point() +
  geom_text_repel(aes(label = province), show.legend = FALSE) + #to use this, aes should be in ggplot, 
```

```
  labs(x = "Cluster",
       y = "GDP in million",
       title = "Argentina’s GDP vs Province Clusters")
```




```
argentina %>%
  mutate(cluster = as.factor(argentina_kmeans$cluster)) %>%
  ggplot(aes(x = cluster,
                y = poverty,
                color = cluster)) +
  geom_point() +
  geom_text_repel(aes(label = province), show.legend = FALSE) + #to use this, aes should be in ggplot,
  labs(x = "Cluster",
       y = "Poverty Rate",
       title = "Argentina’s Poverty vs Province Clusters")
```



Findings:

1. PCA-1 describes the economic condition of the province. Negative direction means the province has a good economic condition, while the positive means the province tends to have a low GPD per capita or high poverty rate