Problem-set-4

Daejin Kim

11/11/2019

Problem Set 4: Dimension Reduction

For the following questions, use the world indicators data from class (countries.csv). Be sure to prepare the data appropriately (e.g., standardize).

```
countries<-read.csv("~/Desktop/Fall Quarter/MACS 40800/MACS problem sets/Problem-Set-4/countries.csv")
```

Factor Analysis

1. How do CFA and EFA differ?

Confirmatory factor analysis (CFA) can be used to test hypotheses when a researcher already has some knowledge of possible underlying factors (Kim & Mueller, p.9). For instance, a researcher may expect how many underlying factors there exist or which variables belong to which factor from prior knowledge. Confirming these hypotheses by performing factor analysis is called confirmatory factor analysis. In contrast, a researcher might choose exploratory factor analysis (EFA) when he does not have any anticipation of the number of underlying factors. Without any hypothesis to test, EFA may enable a researcher to explore the given data to check for possible data reduction. Also, it can provide a minimum number of potential factors.

2. Fit three exploratory factor analysis models initialized at 2, 3, and 4 factors. Present the loadings from these solutions and discuss in substantive terms. How does each fit? What sense does this give you of the underlying dimensionality of the space? And so on.

Across all the fitted models, the data are loaded on the first factor the most, which is shown by the largest squared sum of loadings (SS loadings) of MR1. This follows a general rule of factor analysis that the first factor tries to capture the most variance of the data and the following factor attempts to explain most of the remaining variance. These factor models do represent this general rule except for the 4-factor-model. When four factors were used, the second factor (MR2) does not have the second-largest loadings but have a very similar amount of loadings with factor three and four. This may suggest that the data may not be the best fit for using four factors. On the other hand, using three factors can explain the variances of the data most efficiently. The loadings on each factor decrease by the factor in order (the first factor has the largest loadings and the last has the smallest). Also, the third factor is necessary and unique in that some variables (pop.wdi, milper, & cinc) are only loaded on the third factor. Therefore, the loadings of these models may justify that the data space is composed of three underlying factors.

```
library(tidyverse)
library(psych)

#scaling the data
scaled_c<-countries[,-1] %>%
  scale()

#Fitting factor analysis models initialized at 2, 3, and 4 factors
fac2<-fa(scaled_c,
```

```
        nfactors=2)
fac3<-fa(scaled_c,
        nfactors=3)
fac4<-fa(scaled_c,
        nfactors=4)

#Presenting loadings for each model
fac2$loadings
```

```
##
## Loadings:
##            MR1    MR2
## idealpoint  0.449  0.429
## polity      0.995
## polity2     0.995
## democ       0.931
## autoc      -0.969  0.159
## unreg       0.412 -0.131
## physint             0.782
## speech      0.631  0.154
## new_empinx  0.802  0.197
## wecon               0.509
## wopol       0.551
## wosoc       0.286  0.497
## elecsd      0.852
## gdp.pc.wdi          0.673
## gdp.pc.un           0.671
## pop.wdi     0.204 -0.476
## amnesty            -0.821
## statedept          -0.849
## milper      0.158 -0.468
## cinc        0.211 -0.366
## domestic9   0.288 -0.479
##
##                  MR1   MR2
## SS loadings    6.523 4.527
## Proportion Var 0.311 0.216
## Cumulative Var 0.311 0.526
```

```
fac3$loadings
```

```
##
## Loadings:
##            MR1    MR2    MR3
## idealpoint  0.432  0.468
## polity      0.992
## polity2     0.992
## democ       0.910  0.144
## autoc      -0.994  0.191
## unreg       0.413 -0.129
## physint             0.737 -0.136
## speech      0.646  0.128
```

```
## new_empinx  0.840  0.131 -0.125
## wecon              0.518
## wopol       0.552
## wosoc       0.263  0.547
## elecsd      0.858
## gdp.pc.wdi         0.856  0.158
## gdp.pc.un          0.853  0.157
## pop.wdi                   0.892
## amnesty           -0.715  0.243
## statedept         -0.803  0.144
## milper                    0.949
## cinc                      0.999
## domestic9   0.269 -0.443
##
##                  MR1   MR2   MR3
## SS loadings    6.466 4.275 2.881
## Proportion Var 0.308 0.204 0.137
## Cumulative Var 0.308 0.512 0.649
```

```
fac4$loadings
```

```
##
## Loadings:
##             MR1    MR3    MR4    MR2
## idealpoint  0.467          0.214 -0.294
## polity      0.995
## polity2     0.995
## democ       0.922          0.127
## autoc      -0.986          0.146
## unreg       0.405                 0.165
## physint     0.119                -0.761
## speech      0.658                -0.109
## new_empinx  0.855                -0.145
## wecon       0.105          0.390 -0.170
## wopol       0.555
## wosoc       0.300          0.350 -0.239
## elecsd      0.865
## gdp.pc.wdi                 0.986
## gdp.pc.un                  0.979
## pop.wdi             0.923
## amnesty             0.177 -0.197  0.602
## statedept  -0.137          -0.139  0.783
## milper              0.965
## cinc                0.981  0.111
## domestic9   0.247          0.204  0.757
##
##                  MR1   MR3   MR4   MR2
## SS loadings    6.605 2.811 2.426 2.370
## Proportion Var 0.315 0.134 0.116 0.113
## Cumulative Var 0.315 0.448 0.564 0.677
```

3. Rotate the 3-factor solution using any oblique method you would like and present a visual of the unrotated and rotated versions side-by-side. How do these differ and why does this matter (or not)?

When rotated using an oblique method, the loadings explained by each factor are more spread out than when unrotated. For instance, the biplots of the oblique factor analysis include more data points that are blue or red (the blue dots are loaded on the second factor the most while the red dots are on the third factor). However, the unrotated biplots are dominated by black dots which suggest that most of the variables are loaded on the first factor. Therefore, rotating the factor axes matters because it may change components of each factor as shown in the biplots of each factor analysis.

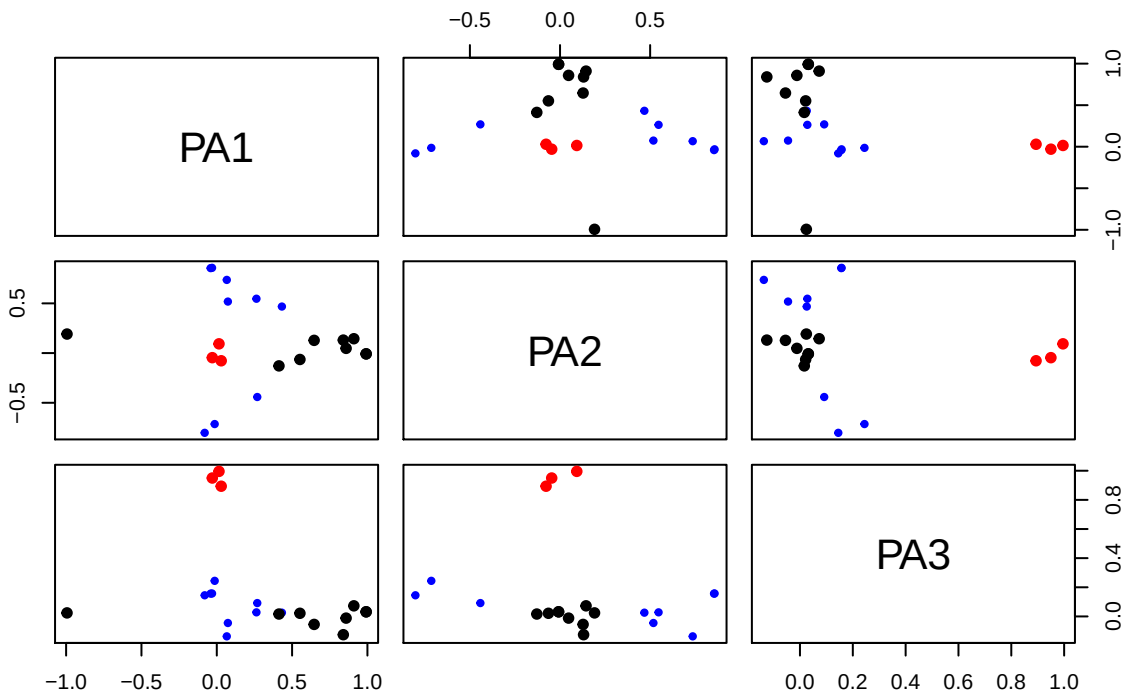
```
library(lattice)
oblique.factors <- fa(cor(scaled_c),
                      fm = "pa",
                      nfactors = 3,
                      rotate = "oblimin")
```

```
## In factor.scores, the correlation matrix is singular, an approximation is used
```

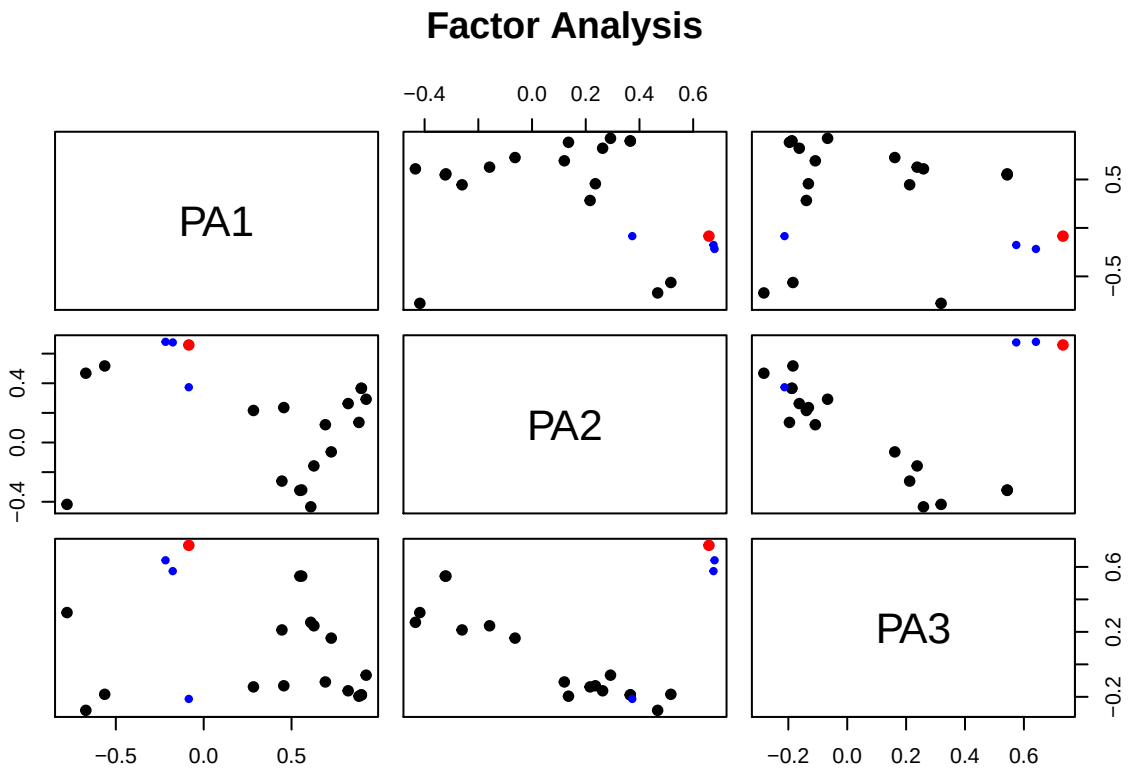
```
non.factors <- fa(cor(scaled_c),
                        fm = "pa",
                        nfactors = 3,
                        rotate = "none",
                        residuals = TRUE)
```

```
## In factor.scores, the correlation matrix is singular, an approximation is used
```

```
plot(oblique.factors)
```




```
plot(non.factors)
```



Principal Components Analysis

1. What is the statistical difference between PCA and FA? Describe the basic construction of each approach using equations and then point to differences that exist across these two widely used methods for reducing dimensionality.

Equations for factor analysis

$$X_1 = b_1 F + d_1 U_1$$

$$X_2 = b_2 F + d_2 U_2$$

Equation for principal component analysis (PCA)

$$Z_{ij} = \phi_{11} X_{i1} + \phi_{21} X_{i2} + ... + \phi_{p1} X_{ip}$$

Factor analysis tries to identify a latent factor that is not observed from the data when a principal component can be derived from the observed data in PCA. This difference can be statistically shown. In the equations for factor analysis, factor (F) potentially contributes to the observed variables (X1, X2) with weights (b1, b2). On the other hand, the variables (X) contribute to the principal component (Z), which is the opposite process to factor analysis.

2. Fit a PCA model. Present the proportion of explained variance across the first 10 components. What do these values tell you substantively (e.g., how many components likely characterize these data?)?

The variance in the data can be explained largely by the first three components, which make up approximately 70% of the total variance (PC1=40.53%, PC2=16.48%, PC3=12.87%). After the third component, the explained variance gets as low as 5%. Therefore, it seems that there are three principal components (PC1, PC2, & PC3) that characterize the data.

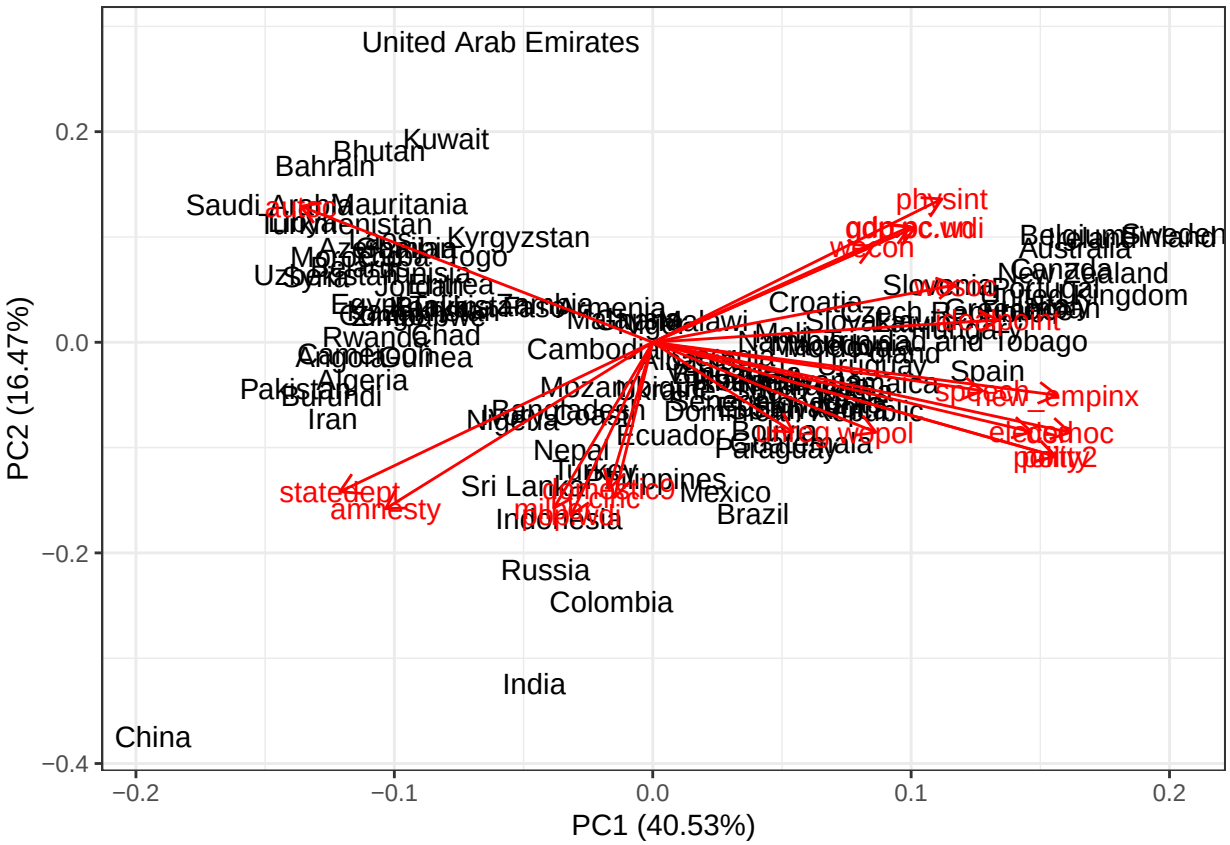
```
library(tidyverse)
library(ggfortify)
#changing country variable as an attribute
countries_new<-countries[,-1]
nation<-countries$X
row.names(countries_new) <- nation
pca_count <- prcomp(countries_new, scale = TRUE)

#The proportion of variance across the first 10 components
s<-summary(pca_count)
s<-s$importance
s[2,1:10]
```

```
##     PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9
## 0.40528 0.16475 0.12869 0.05837 0.05516 0.03968 0.02911 0.02534 0.01976
##    PC10
## 0.01641
```

3. Present a biplot of the PCA fit from the previous question. Describe what you see (e.g., which countries are clustered together? Which input features are doing the bulk of the explaining? How do you know this?)

In the biplot, many Middle Eastern countries, such as Saudi Arabia, Iran, Pakistan, seem to be clustered together driving the "autoc" variable. Many other countries are widely spread along the line that connects China to Sweden on the far right. These widely spread countries largely explain the variance of the PC1.

```
autoplot(pca_count,
         shape = F,
         loadings.label = T) +
theme_bw()
```



Bonus Question (5 points):

1. Fit a sparse PCA model and a probabilistic PCA model. Compare these results substantively. What does each tell you and why do these distinctions matter in terms of inference (or not)?

I could not find any consistent pattern in each model when I tried to fit them using "do.ppca()" and "do.pca()" functions.

```
library(Rdimtools)
```

```
## ** Rdimtools
```

```
## **  - Dimension Reduction and Estimation Toolbox
```

```
## ** Version    : 0.4.2 (2019)
```

```
## ** Maintainer : Kisung You (kyou@nd.edu)
```

```
## **
```

```
## ** Please share any bugs or suggestions to the maintainer.
```

```r
PPCA<-do.ppca(countries_new, ndim=3, preprocess="center")
PCA <- do.pca(countries_new, ndim=3, preprocess="center")
sqrPPCA<-(PPCA$Y)^2
colSums(sqrPPCA)
```

```
## [1] 5.664244e-27 2.089636e+07 1.228684e+16
```

```r
sqrPCA<-(PCA$Y)^2
colSums(sqrPCA)
```

```
## [1] 2.696422e+18 1.398941e+10 2.032219e+08
```