PS4

Rui He

11/11/2019

EFA & CFA

Exploratory factor analysis provides a method to examine the latent dimensions to account for the variance in a dataset. It is exploited at the early stage of data analysis to learn about the factor stucture of dimension and give an intuitive sense of what are some major factors that can be used to decompose the data.

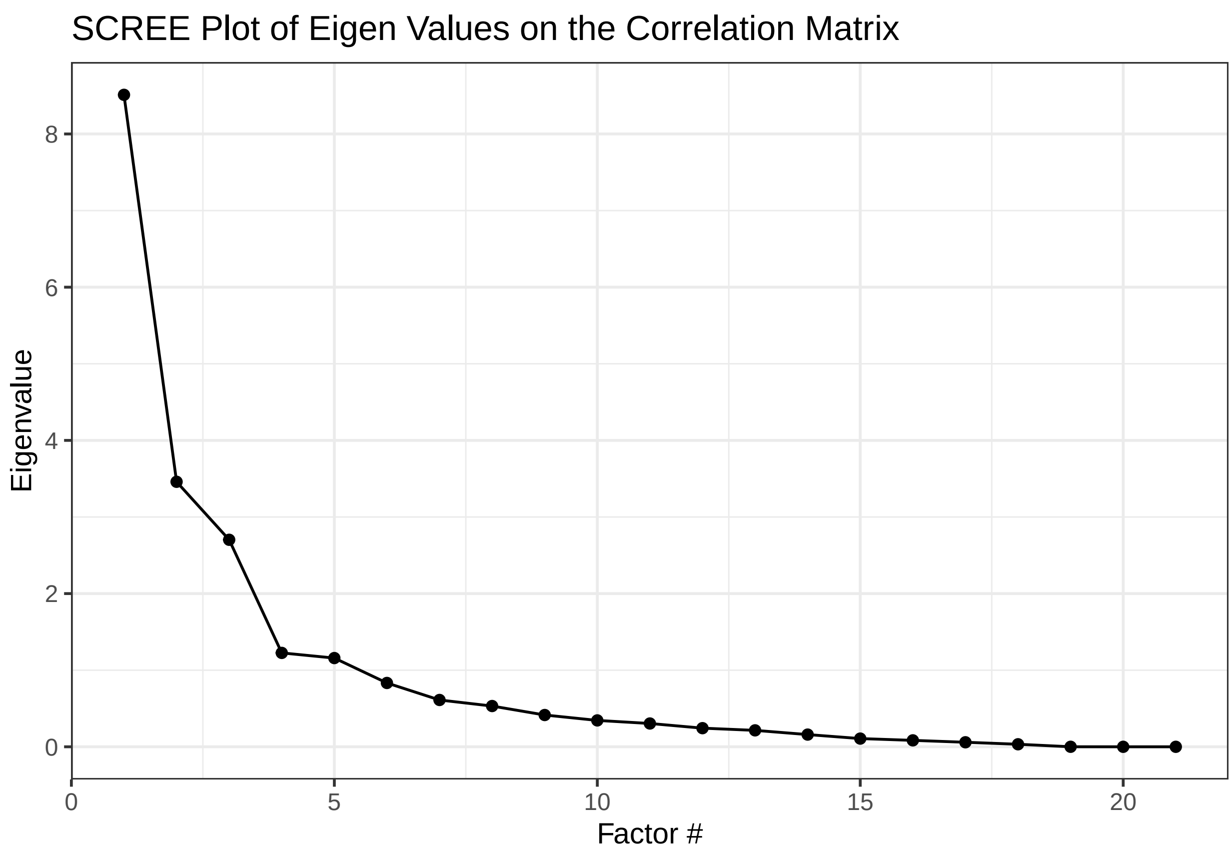
Confirmatory factor analysis is usually based on some a priori assumption and pre-established theory and examine if the number of factors and input feature loadings are conform to the expectations generated from the apriori assumptions.

Factor Analysis

```r
data<- rawdata%>%
  select(-X)
scaled_df <- scale(data)
dfcor<-cor(scaled_df)


# Next, generate the eigenvalues
ev <- eigen(dfcor) # store EVs on the correlation matrix
ev$values
```

```
##  [1]  8.510913e+00  3.459670e+00  2.702458e+00  1.225737e+00  1.158450e+00
##  [6]  8.333739e-01  6.112287e-01  5.321391e-01  4.150076e-01  3.446022e-01
## [11]  3.043095e-01  2.434553e-01  2.147082e-01  1.595778e-01  1.073515e-01
## [16]  8.416652e-02  5.927696e-02  3.317877e-02  3.962048e-04  1.230485e-15
## [21] -9.239694e-18
```

```r
# Next, generate Scree plot
qplot(y = ev$values,
      main = 'SCREE Plot of Eigen Values on the Correlation Matrix',
      xlab = 'Factor #',
      ylab = 'Eigenvalue') +
    geom_line() +
    theme_bw()
```




```r
factan.2 <- fa(scaled_df,nfactors = 2)
```

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

```r
factan.3 <- fa(scaled_df,nfactors = 3)
```

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

```r
factan.4 <- fa(scaled_df,nfactors = 4)
```

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

```r
factan.2$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
```

```r
factan.3$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
```

```r
factan.4$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
```

```r
nonrotated.factors <- fa(cor(scaled_df),
        fm = "pa", # communalities along the diagonal (total variation across features)
        nfactors = 3,
        rotate = "none",
        residuals = TRUE)
```

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

```r
nonrotated.factors$loadings
```

```
##
## Loadings:
##            PA1    PA2    PA3
## idealpoint  0.726         0.162
## polity      0.897  0.366 -0.188
## polity2     0.897  0.366 -0.188
## democ       0.925  0.292
## autoc      -0.778 -0.418  0.319
## unreg       0.283  0.216 -0.139
## physint     0.610 -0.434  0.259
```

```
## speech      0.693  0.120 -0.108
## new_empinx  0.884  0.136 -0.196
## wecon       0.445 -0.260  0.212
## wopol       0.456  0.236 -0.132
## wosoc       0.627 -0.158  0.238
## elecsd      0.822  0.263 -0.163
## gdp.pc.wdi  0.558 -0.321  0.543
## gdp.pc.un   0.548 -0.323  0.543
## pop.wdi    -0.176  0.676  0.574
## amnesty    -0.563  0.517 -0.184
## statedept  -0.671  0.468 -0.283
## milper     -0.217  0.680  0.641
## cinc               0.659  0.733
## domestic9          0.373 -0.213
##
##                  PA1   PA2   PA3
## SS loadings    8.258 3.202 2.512
## Proportion Var 0.393 0.152 0.120
## Cumulative Var 0.393 0.546 0.665
```

```r
nonrot.pattern <- as.data.frame(nonrotated.factors$loadings[1:8,])

nonrot <- xyplot(PA2 ~ PA1, data = nonrot.pattern,
       aspect = 1,
       xlim = c(-.1, 1.2),
       ylim = c(-.5, .8),
       panel = function (x, y) {
         panel.segments(c(0, 0), c(0, 0),
             c(1, 0), c(0, 1), col = "gray")
         panel.text(1, 0, labels = "Initial\n(unrotated)\nfactor 1",
                    cex = .65, pos = 3, col = "gray")
         panel.text(0, .7, labels = "Initial\n(unrotated)\nfactor 2",
                    cex = .65, pos = 4, col = "gray")
         panel.segments(rep(0, 8), rep(0, 8), x, y,
            col = "black")
         panel.text(x[-7], y[-7], labels = rownames(nonrot.pattern)[-7],
            pos = 4, cex = .75)
         panel.text(x[7], y[7], labels = rownames(nonrot.pattern)[7],
                    pos = 1, cex = .75)
       },
       main = "Unrotated Factor Pattern",
       xlab = "",
       ylab = "",
       scales = list(x = list(at = c(0, 1)),
                     y = list(at = c(-.4, 0, .6)))
)



oblique.factors <- fa(cor(scaled_df),
        fm = "pa", # communalities along the diagonal (total variation across features)
        nfactors = 3,
        rotate = "oblimin",
        residuals = TRUE)
```

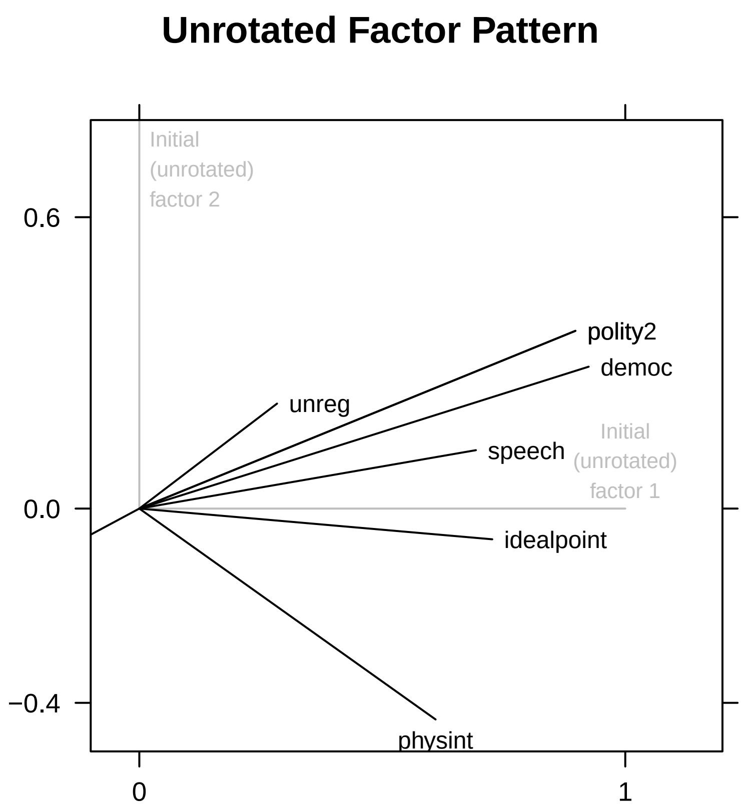
```
## In factor.scores, the correlation matrix is singular, an approximation is used
```

```r
oblique.factors$loadings
```

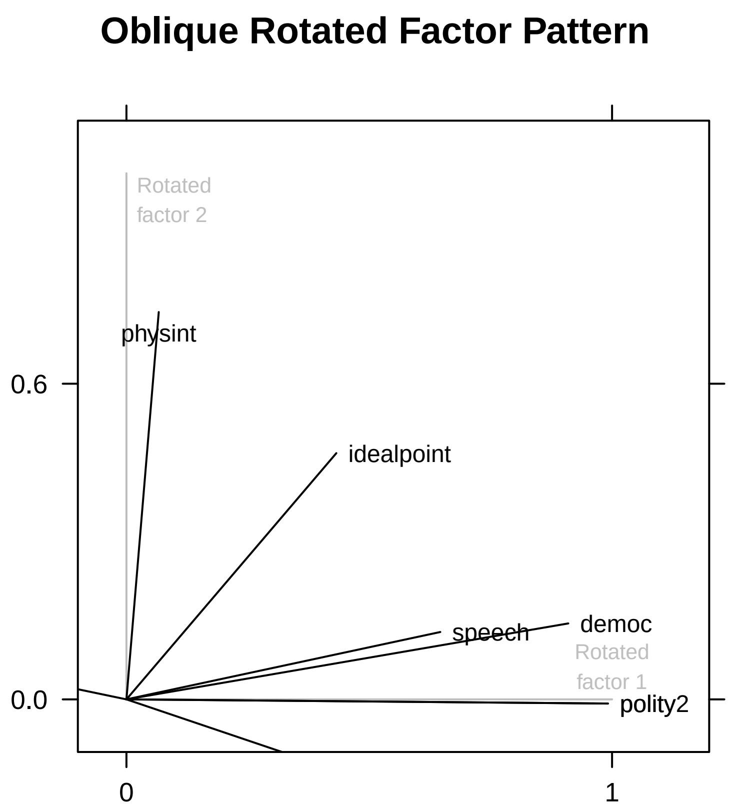
```
##
## Loadings:
##            PA1    PA2    PA3
## idealpoint  0.432  0.468
## polity      0.992
## polity2     0.992
## democ       0.910  0.144
## autoc      -0.994  0.191
## unreg       0.413 -0.129
## physint            0.736 -0.137
## speech      0.646  0.128
## new_empinx  0.840  0.131 -0.125
## wecon              0.518
## wopol       0.552
## wosoc       0.263  0.547
## elecsd      0.858
## gdp.pc.wdi         0.857  0.157
## gdp.pc.un          0.855  0.156
## pop.wdi                   0.894
## amnesty           -0.714  0.244
## statedept         -0.802  0.145
## milper                    0.950
## cinc                      0.996
## domestic9   0.269 -0.442
##
##                  PA1   PA2   PA3
## SS loadings    6.467 4.274 2.880
## Proportion Var 0.308 0.204 0.137
## Cumulative Var 0.308 0.511 0.649
```

```r
oblique.pattern <- as.data.frame(oblique.factors$loadings[1:8,])

obliq <- xyplot(PA2 ~ PA1, data = oblique.pattern,
       aspect = 1,
       xlim = c(-.1, 1.2),
       ylim = c(-.1, 1.1),
       panel = function (x, y) {
         panel.segments(c(0, 0), c(0, 0),
                        c(1, 0), c(0, 1), col = "gray")
         panel.text(1, 0, labels = "Rotated\nfactor 1",
                    cex = .65, pos = 3, col = "gray")
         panel.text(0, .95, labels = "Rotated\nfactor 2",
                    cex = .65, pos = 4, col = "gray")
         panel.segments(rep(0, 8), rep(0, 8), x, y,
                        col = "black")
         panel.text(x[-7], y[-7], labels = rownames(oblique.pattern)[-7],
                    pos = 4, cex = .75)
         panel.text(x[7], y[7], labels = rownames(oblique.pattern)[7],
                    pos = 1, cex = .75)
```

```
        },
        main = "Oblique Rotated Factor Pattern",
        xlab = "",
        ylab = "",
        scales = list(x = list(at = c(0, 1)),
                      y = list(at = c(-.4, 0, .6)))
)

nonrot
```




```
obliq
```



2. With a 2-factor model, we can see the three variables polity, polity2, and democ have the highest loading on factor 1, suggesting factor 1 is a main indicator of the political regime on a spectrum of autocracy to democracy. physin, gdp.pc.un and gdp.pc.wdi have higher loadings on factor 2, suggesting that the physical integrity and per capita income contribute to the second factor and probably an indication of human right. The two factors explained about 53% of variance in the data.

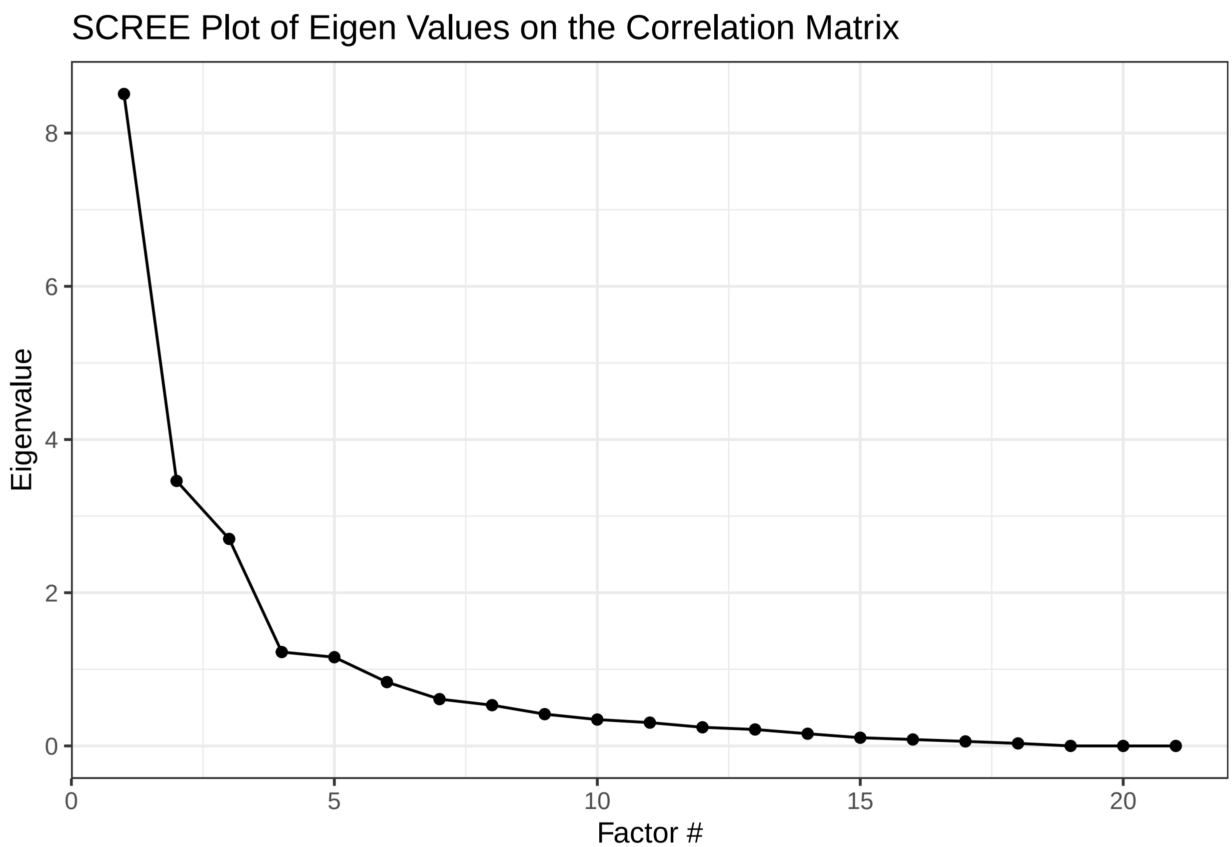
With a 3-factor model, the first factor is still the political regime. The second factor is still human right, And the third factor have highest loading on milper and cinc, which indicates high military power. And the additional factor explained about 14% of the variance in the data.

With a 4-factor model, the second factor only have high loadings from gdp.pc.wdi and gdp.pc.un but not physint. The fourth factor have highest loading from physint. Adding another factor may in fact separate out the factor that indicate the financial status of the countries from the physical integrity in the latent structure.

PCA

1. Difference between PCA & FA: For factor analysis, the factors are assumed to be the cause of the observed outcome, and the latent variables are assumed to follow a Gaussian distribution. Whereas for principal component analysis, the component are the outcomes built from the combination of the items with no assumptions. PCA assuming no correlation between the latent variables and therefore allowing for true statistical independence.

```r
dfcov <- cov(scaled_df)
ev <- eigen(dfcov)

qplot(y = ev$values,
      main = 'SCREE Plot of Eigen Values on the Correlation Matrix',
      xlab = 'Factor #',
      ylab = 'Eigenvalue') +
  geom_line() +
  theme_bw()
```




```r
ph1_perc = ev$values[1] / sum(ev$values)
ph2_perc = ev$values[2] / sum(ev$values)
ph1_perc
```

```
## [1] 0.4052816
```

```r
ph2_perc
```

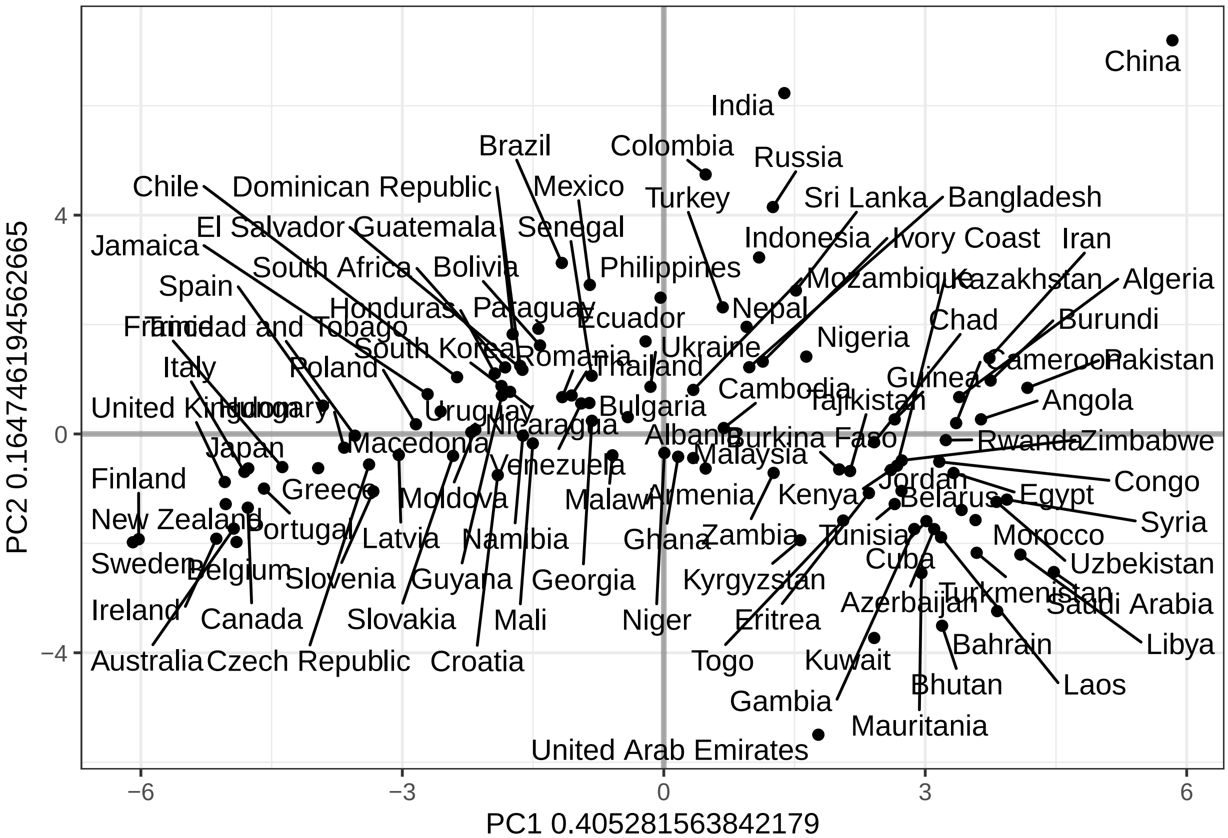
```
## [1] 0.1647462
```

```r
# extract first two loadings
phi <- ev$vectors[ , 1:2]
row.names(phi) <- names(data)
```

```r
colnames(phi) <- c("PC1", "PC2")

# Calculate scores
Z1 <- as.matrix(select_if(as_tibble(scaled_df), is.numeric)) %*% phi[,1]
Z2 <- as.matrix(select_if(as_tibble(scaled_df), is.numeric)) %*% phi[,2]

# Create data frame with Principal Components scores
(PC <- tibble(
  Geography = rawdata$X,
  PC1 = Z1[,1],
  PC2 = Z2[,1]
))
```

```
## # A tibble: 107 x 3
##    Geography               PC1    PC2
##    <fct>                 <dbl>  <dbl>
##  1 Angola                3.64   0.267
##  2 Albania              -0.413  0.308
##  3 United Arab Emirates  1.77  -5.50
##  4 Armenia               0.480 -0.632
##  5 Australia            -4.93  -1.73
##  6 Azerbaijan            3.10  -1.74
##  7 Burundi               3.75   0.977
##  8 Belgium              -4.90  -1.97
##  9 Burkina Faso          2.01  -0.648
## 10 Bangladesh            0.981  1.22
## # ... with 97 more rows
```

```r
ggplot(PC, aes(PC1, PC2)) +
  geom_vline(xintercept = 0, size = 1, alpha = .3) +
  geom_hline(yintercept = 0, size = 1, alpha = .3) +
  geom_point() +
  ggrepel::geom_text_repel(aes(label = Geography)) +
  labs(x = paste("PC1",toString(ph1_perc)),
       y = paste("PC2",toString(ph2_perc))) +
  theme_bw()
```



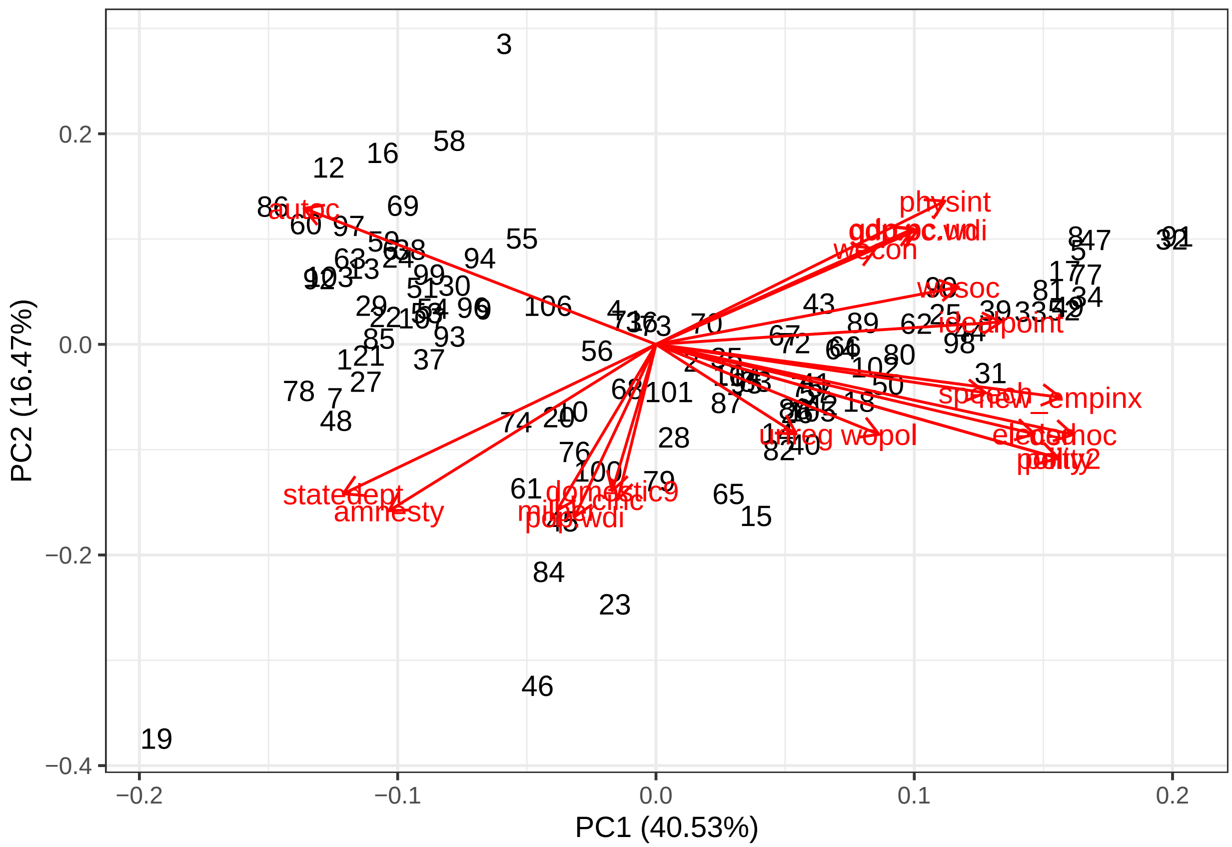

```r
pca.out <- prcomp(scaled_df, scale = T)

summary(pca.out)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6
## Standard deviation     2.9173 1.8600 1.6439 1.10713 1.07631 0.91289
## Proportion of Variance 0.4053 0.1648 0.1287 0.05837 0.05516 0.03968
## Cumulative Proportion  0.4053 0.5700 0.6987 0.75708 0.81225 0.85193
##                            PC7     PC8     PC9    PC10    PC11    PC12
## Standard deviation     0.78181 0.72948 0.64421 0.58703 0.55164 0.49341
## Proportion of Variance 0.02911 0.02534 0.01976 0.01641 0.01449 0.01159
## Cumulative Proportion  0.88104 0.90638 0.92614 0.94255 0.95704 0.96864
##                           PC13   PC14    PC15    PC16    PC17    PC18
## Standard deviation     0.46337 0.3995 0.32765 0.29011 0.24347 0.18215
## Proportion of Variance 0.01022 0.0076 0.00511 0.00401 0.00282 0.00158
## Cumulative Proportion  0.97886 0.9865 0.99157 0.99558 0.99840 0.99998
##                           PC19      PC20      PC21
## Standard deviation     0.01990 5.378e-16 2.786e-16
## Proportion of Variance 0.00002 0.000e+00 0.000e+00
## Cumulative Proportion  1.00000 1.000e+00 1.000e+00
```

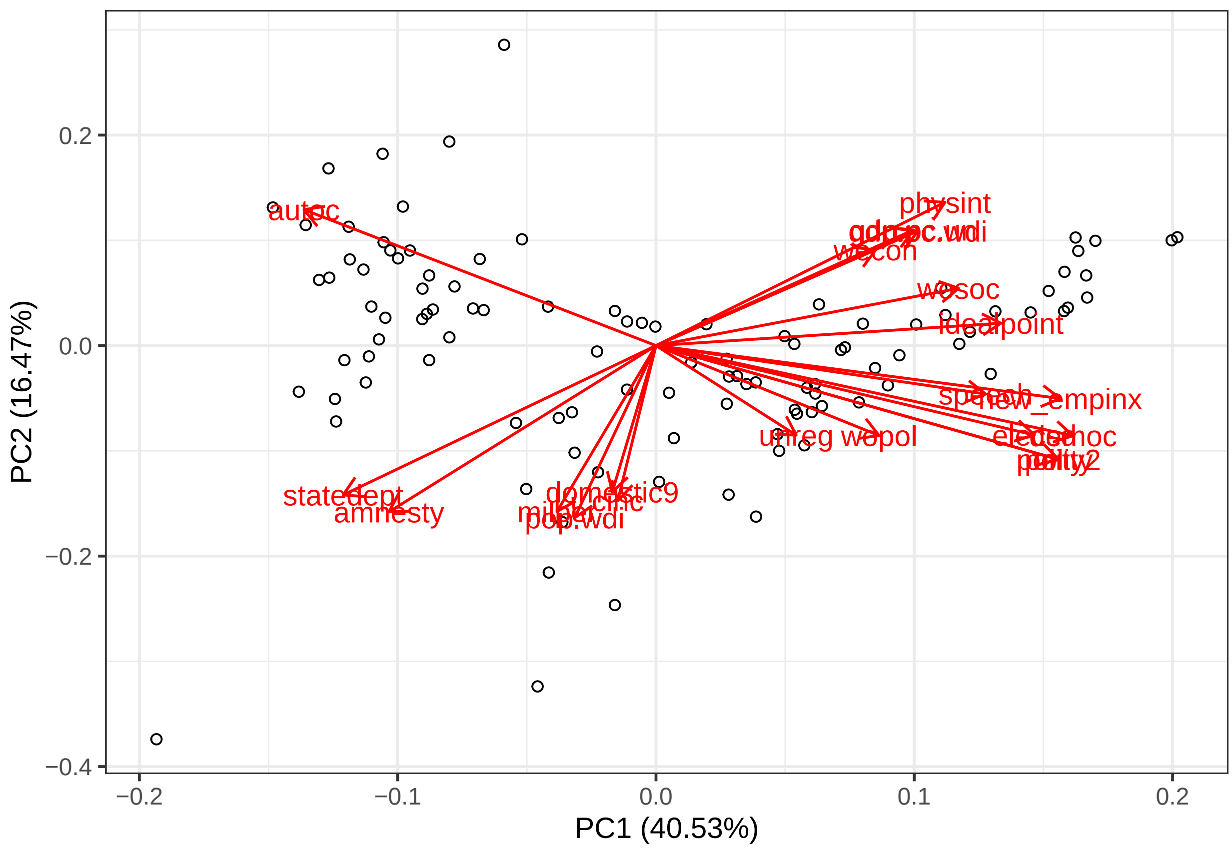
```r
names(pca.out)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

```r
# visualize
autoplot(pca.out,
         shape = F,
         loadings.label = T) +
  theme_bw()
```




```r
autoplot(pca.out,
         shape = T,
         loadings.label = T) +
  theme_bw()
```



2. As we can tell from the summary of pca output,~70% of variance can be explained by the first 3 principal componenets, suggesting that there are probably 3 componenets that are most likely to characterize the data

3. We can see some geospatial pattern that PC1 roughly corresponding to an axis of the political regime on a spectrum from democracy on the left to autocracy on the right, which is consistent to the factor analysis.