Let’s take a brief look at the data we will use for factor analysis:

> str(data1)

'data.frame': 28 obs. of 6 variables:

$ GDPPC: num 1.93 1.95 1.79 1.89 1.92 ...

$ SS : num 1.16 1.24 1.23 1.26 1.09 ...

$ HLE : num 0.774 0.777 0.728 0.775 0.776 0.779 0.79 0.787 ...

$ FMLC : num 0.623 0.719 0.689 0.736 0.585 0.627 0.7 0.651 ...

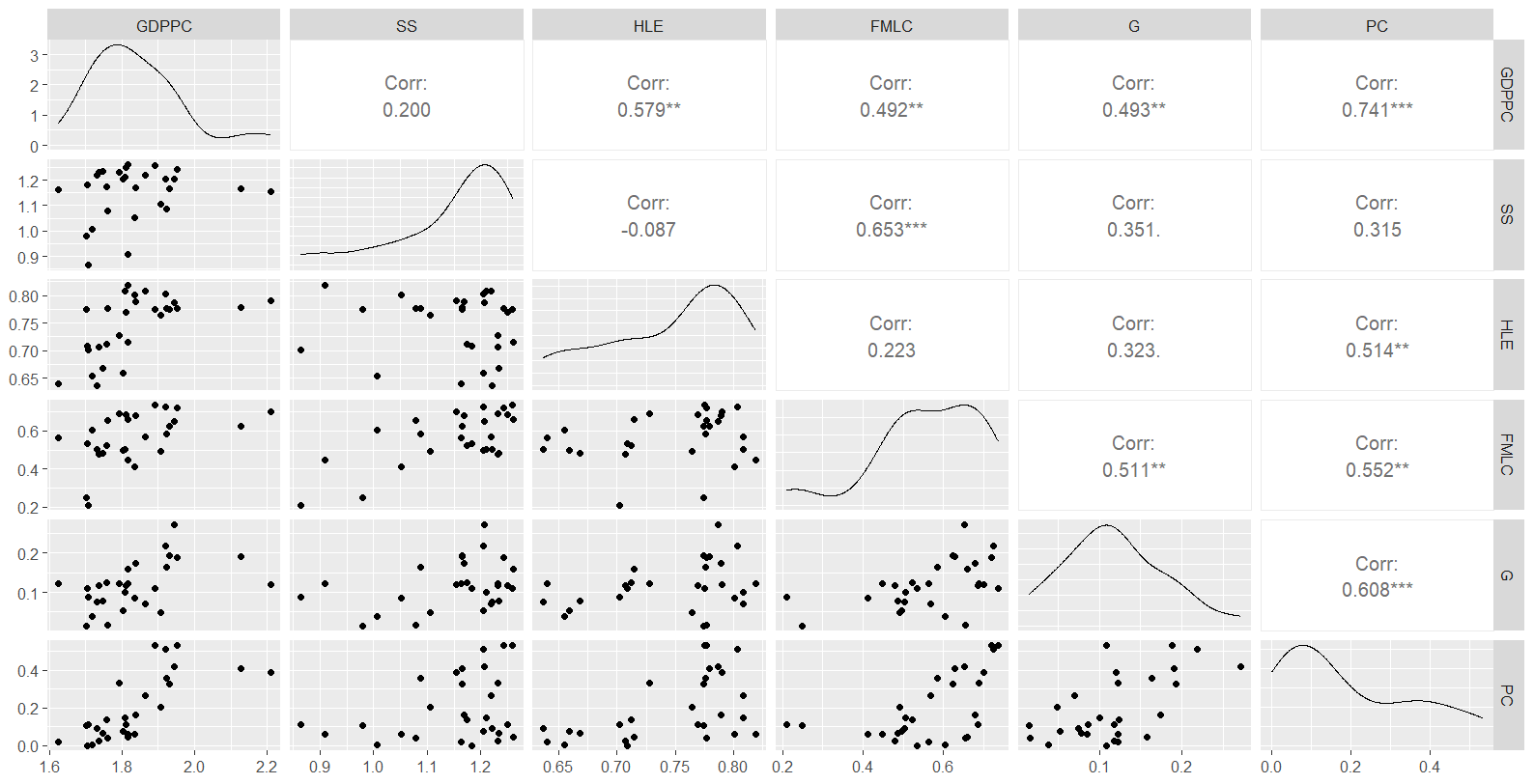
$ G : num 0.193 0.188 0.123 0.109 0.163 0.19 0.12 0.271 ...

$ PC : num 0.329 0.532 0.333 0.534 0.358 0.408 0.388 0.419 ...

The data consists of 6 numeric variables. Categorical variables that were not suitable for factor analysis were previously removed from the data. The explanations of these 6 variables can be found in the file "Criteria Explanations.pdf".

To examine the variables in more detail, we can use the ggpairs function to show correlation scatter plots and density plots:

> GGally::ggpairs(data1)



By looking at the scatter plots, we can see that the relationships are generally linear, indicating that the linearity assumption is met. When we look at the correlation values, we observe a 0.74 correlation between PC and GDPPC. This might indicate multicollinearity. Apart from that, our correlation matrix seems to differ significantly from the identity matrix. We can apply VIF and Bartlett's sphericity tests for these.

> cortest.bartlett(cor(data1),n=28)

$p.value

[1] 6.759564e-09

A p-value less than 0.05 in Bartlett's test indicates that the correlation matrix is significantly different from the identity matrix. Now, let's apply the VIF test to check for multicollinearity:

> qwe = 1:28

> formula <- as.formula(paste("qwe ~", paste(names(data1), collapse = " + ")))

> model <- lm(formula, data = data1)

> car::vif(model)

GDPPC SS HLE FMLC G PC

2.629527 1.975275 1.712474 2.458635 1.722245 2.905439

When we look at the results of the test, we see that none of the VIF values exceed the critical value of 10 for multicollinearity.

To see if the sample is adequate for factor analysis, we can apply the KMO test:

> KMO(cor(data1))

Kaiser-Meyer-Olkin factor adequacy

Call: KMO(r = cor(data1))

Overall MSA = 0.77

MSA for each item =

GDPPC SS HLE FMLC G PC

0.79 0.61 0.77 0.75 0.89 0.80

An overall MSA value of 0.77 indicates that our sample is sufficiently adequate. Additionally, when we look at the individual MSA values for each variable, we see that all of them are above 0.6, indicating they are suitable for factor analysis.

After performing the necessary tests, we need to decide how many factors to use before starting the factor analysis. To make this decision, we can use the eigenvalues scree test plot and parallel method plots:

> data.frame(bilesen=1:6,ozdeger=scree(data1,pc=F)$fv)

bilesen ozdeger

1 1 2.788252984

2 2 0.639451874

3 3 0.009412939

4 4 -0.035281259

5 5 -0.202160048

6 6 -0.411421413

The eigenvalues of our variables are as shown above. It appears that only one component has an eigenvalue greater than 1. Although using the number of components with eigenvalues greater than 1 is an option, it is better not to make a decision without looking at the scree and parallel method plots.

Scree test grafiği:

> scree(data1,pc = F)

A graph with numbers and points

Description automatically generated

When we look at the plot, it seems more likely that the elbow point is at the second factor. We can reach a definitive conclusion by also looking at the parallel method plot:

> fa.parallel(data1,fa = "fa")

A graph of a number

Description automatically generated with medium confidenceParallel analysis suggests that the number of factors = 2

The red dashed lines in the plot show the eigenvalues obtained with the boost technique. According to this plot, it can be definitively said that the elbow starts from the second factor. It should not be overlooked that the output of the function also suggests using 2 factors.

After performing the necessary tests and determining the number of factors, we can now start the factor analysis. We can use the fa() function from the psych package for this. Let's do our first factor analysis without rotation:

> f1 = fa(scale(data1),nfactors = 2,rotate = "none")

> f1

Factor Analysis using method = minres

Call: fa(r = scale(data1), nfactors = 2, rotate = "none")

Standardized loadings (pattern matrix) based upon correlation matrix

MR1 MR2 h2 u2 com

GDPPC 0.80 -0.31 0.73 0.27 1.3

SS 0.52 0.73 0.80 0.20 1.8

HLE 0.51 -0.49 0.50 0.50 2.0

FMLC 0.75 0.35 0.68 0.32 1.4

G 0.67 0.02 0.45 0.55 1.0

PC 0.86 -0.18 0.77 0.23 1.1

MR1 MR2

SS loadings 2.91 1.02

Proportion Var 0.49 0.17

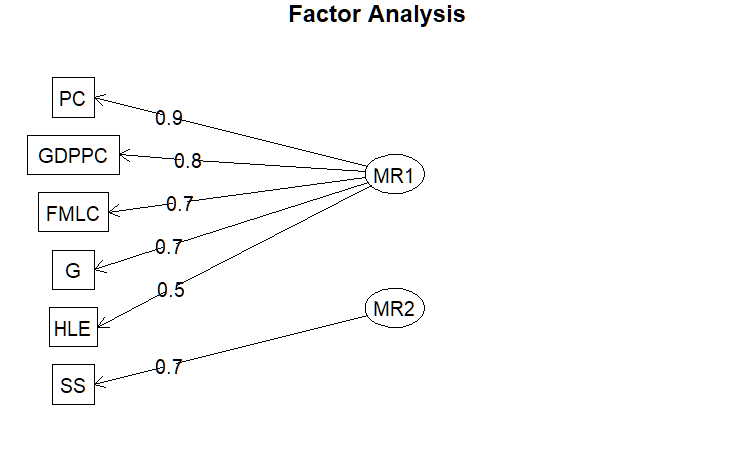
Cumulative Var 0.49 0.66

Proportion Explained 0.74 0.26

Cumulative Proportion 0.74 1.00

Mean item complexity = 1.4

The h2 (communality), u2 (uniqueness), and com (complexity) values are shown. When we look at the variance ratios, we see that the first factor explains 49% of the variance and the second factor explains 17%. Additionally, it is noteworthy that the variables G and HLE have low h2 values, also, SS and HLE have high com values.

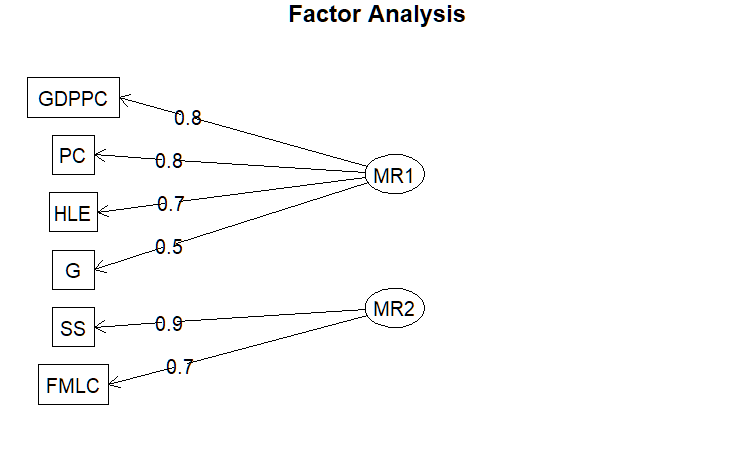
> fa.diagram(f1)

The factor diagram on the side visualizes the result of the factor analysis. Accordingly, while the first five variables are grouped under the first factor, we see only the SS variable in the second factor. Based on the explanations of these variables, new names can be considered for the MR1 and MR2 factors

Finally, when we calculate the factor scores by multiplying the factor scores with the proportion of variance explained by the factors and rank the countries accordingly:

> data.frame(ulkeler = asd$Country,skor = f1$scores %\*% f1$Vaccounted[4,] ) %>% arrange(desc(skor))

|  |  |
| --- | --- |
| ulkeler skor  1 Denmark 1.16161855  2 Finland 1.07283983  3 Sweden 1.04013778  4 Netherla 0.90398350  5 Luxembou 0.85955537  6 Ireland 0.80947927  7 Estonia 0.56524331  8 Austria 0.50192526  9 Slovenia 0.36924109  10 Czechia 0.31299897  11 France 0.29361968  12 Malta 0.28498679  13 Germany 0.19672045  14 Spain 0.01126947 | 15 Poland -0.15184312  16 Hungary -0.21123646  17 Slovakia -0.21563316  18 Latvia -0.22398407  19 Lithuani -0.24699671  20 Belgium -0.25096555  21 Croatia -0.35460615  22 Bulgaria -0.44203799  23 Portugal -0.47874137  24 Italy -0.73350409  25 Romania -0.88732613  26 Cyprus -1.07987592  27 Greece -1.38070763  28 Turkey -1.72616096 |

**Varimax rotation:**

|  |
| --- |
| > f2 = fa(scale(data1),nfactors = 2,rotate = "varimax")  > f2  Factor Analysis using method = minres  Call: fa(r = scale(data1), nfactors = 2, rotate = "varimax")  Standardized loadings (pattern matrix) based upon correlation matrix  MR1 MR2 h2 u2 com  GDPPC 0.82 0.24 0.73 0.27 1.2  SS -0.02 0.89 0.80 0.20 1.0  HLE 0.71 -0.08 0.50 0.50 1.0  FMLC 0.38 0.73 0.68 0.32 1.5  G 0.52 0.42 0.45 0.55 1.9  PC 0.79 0.37 0.77 0.23 1.4  MR1 MR2  SS loadings 2.21 1.72  Proportion Var 0.37 0.29  Cumulative Var 0.37 0.66  Proportion Explained 0.56 0.44  Cumulative Proportion 0.56 1.00  Mean item complexity = 1.3  > fa.diagram(f2) |
|  |
| |  | | --- | |  | |

We observe that the number of variables in the factors has changed compared to the factor analysis without any rotation. Additionally, it is noted that the complexity values are smaller, although the variable G still has a high complexity value of 1.9.

> data.frame(ulkeler = asd$Country,skor = f2$scores %\*% f2$Vaccounted[4,] ) %>% arrange(desc(skor))

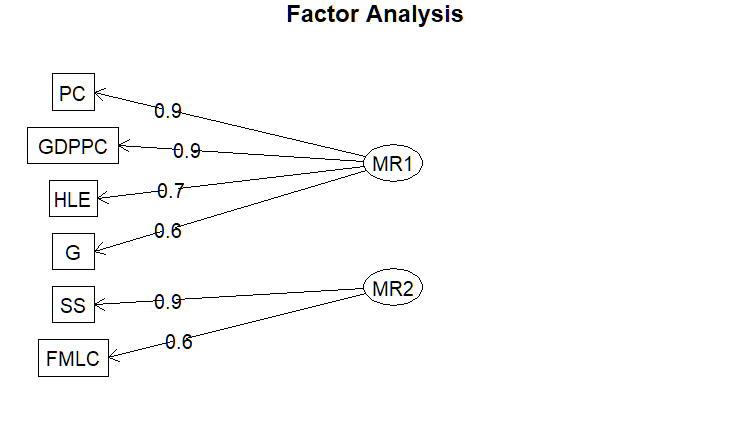
|  |  |
| --- | --- |
| ulkeler skor  1 Denmark 1.096992357  2 Luxembou 1.067126507  3 Sweden 1.042469033  4 Ireland 0.977510499  5 Finland 0.948088432  6 Netherla 0.905363984  7 Austria 0.551931343  8 Germany 0.404103417  9 Estonia 0.366668234  10 France 0.259924442  11 Malta 0.236322263  12 Slovenia 0.137227861  13 Czechia 0.008206128  14 Spain -0.039956126 | 15 Belgium -0.086268171  16 Poland -0.258709154  17 Portugal -0.427095891  18 Lithuani -0.430490999  19 Slovakia -0.448710788  20 Hungary -0.465768168  21 Italy -0.481058535  22 Latvia -0.493467820  23 Croatia -0.542518622  24 Cyprus -0.585008405  25 Bulgaria -0.702960819  26 Romania -0.826907818  27 Greece -1.010601339  28 Turkey -1.202411846 |

**Quartimax rotation:**

> f3 = fa(scale(data1),nfactors = 2,rotate = "quartimax")

> f3

> fa.diagram(f3)

 MR1 MR2 h2 u2 com

GDPPC 0.85 0.07 0.73 0.27 1.0

SS 0.15 0.88 0.80 0.20 1.1

HLE 0.68 -0.21 0.50 0.50 1.2

FMLC 0.52 0.64 0.68 0.32 1.9

G 0.59 0.31 0.45 0.55 1.5

PC 0.85 0.21 0.77 0.23 1.1

MR1 MR2

SS loadings 2.55 1.38

Proportion Var 0.42 0.23

Cumulative Var 0.42 0.66

Proportion Explained 0.65 0.35

Cumulative Proportion 0.65 1.00

Mean item complexity = 1.3

Despite the variables in the factors being the same as in the factor analysis we performed with varimax, we see that the factor loadings of the variables in quartimax have slightly increased

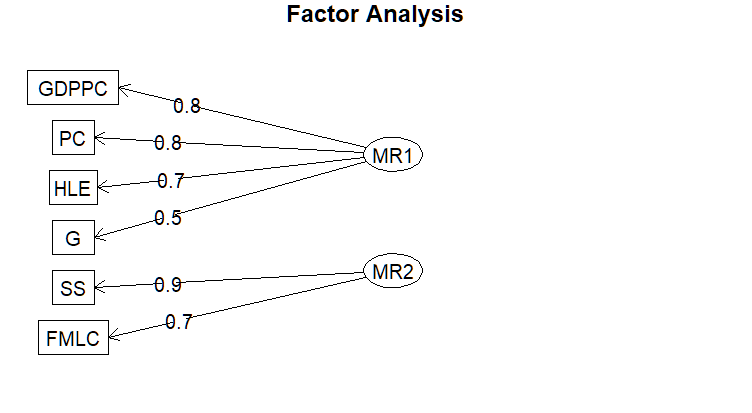
> data.frame(ulkeler = asd$Country,skor = f3$scores %\*% f3$Vaccounted[4,] ) %>% arrange(desc(skor))

|  |  |
| --- | --- |
| ulkeler skor  1 Denmark 1.13619770  2 Luxembou 1.07660308  3 Sweden 1.07296994  4 Finland 0.98927396  5 Ireland 0.98853432  6 Netherla 0.93192170  7 Austria 0.56291103  8 Estonia 0.39852851  9 Germany 0.39404384  10 France 0.27116207  11 Malta 0.24845020  12 Slovenia 0.16586454  13 Czechia 0.04075055  14 Spain -0.03570568 | 15 Belgium -0.10626754  16 Poland -0.25501405  17 Lithuani -0.42374149  18 Slovakia -0.43724377  19 Portugal -0.44516669  20 Hungary -0.45253053  21 Latvia -0.47946255  22 Italy -0.52200206  23 Croatia -0.53860516  24 Cyprus -0.65471019  25 Bulgaria -0.69604191  26 Romania -0.85770097  27 Greece -1.07963377  28 Turkey -1.29338507 |

**Equamax rotation:**

> f4 = fa(scale(data1),nfactors = 2,rotate = "equamax")

> f4

 MR1 MR2 h2 u2 com

GDPPC 0.84 0.17 0.73 0.27 1.1

SS 0.04 0.89 0.80 0.20 1.0

HLE 0.70 -0.13 0.50 0.50 1.1

FMLC 0.43 0.70 0.68 0.32 1.7

G 0.55 0.38 0.45 0.55 1.8

PC 0.82 0.31 0.77 0.23 1.3

MR1 MR2

SS loadings 2.35 1.58

Proportion Var 0.39 0.26

Cumulative Var 0.39 0.66

Proportion Explained 0.60 0.40

Cumulative Proportion 0.60 1.00

Mean item complexity = 1.3

> data.frame(ulkeler = asd$Country,skor = f4$scores %\*% f4$Vaccounted[4,] ) %>% arrange(desc(skor))

|  |  |
| --- | --- |
| ulkeler skor  1 Denmark 1.10922245  2 Luxembou 1.07216773  3 Sweden 1.05247525  4 Ireland 0.98268884  5 Finland 0.96040506  6 Netherla 0.91407051  7 Austria 0.55599267  8 Germany 0.40274775  9 Estonia 0.37524261  10 France 0.26328829  11 Malta 0.23983772  12 Slovenia 0.14443463  13 Czechia 0.01601220  14 Spain -0.03904329 | 15 Belgium -0.09127637  16 Poland -0.25849789  17 Lithuani -0.42999572  18 Portugal -0.43252888  19 Slovakia -0.44713443  20 Hungary -0.46381262  21 Latvia -0.49140067  22 Italy -0.49210306  23 Croatia -0.54299299  24 Cyprus -0.60320225  25 Bulgaria -0.70313341  26 Romania -0.83642350  27 Greece -1.02974157  28 Turkey -1.22729907 |

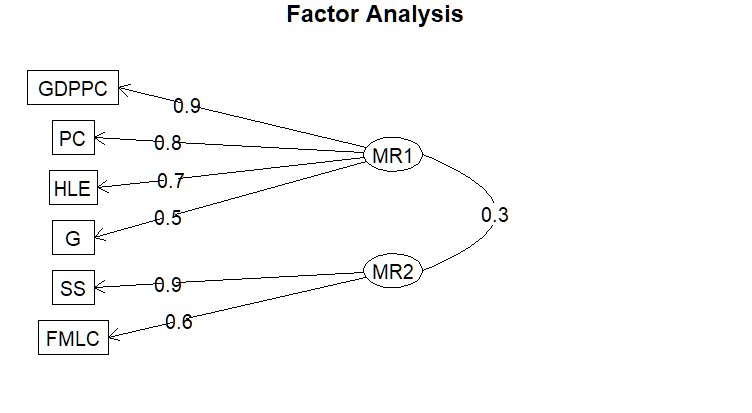
**Oblimin rotation:**

> f5 = fa(scale(data1),nfactors = 2,rotate = "oblimin")

> f5

MR1 MR2 h2 u2 com

GDPPC 0.85 0.00 0.73 0.27 1.0

SS -0.06 0.91 0.80 0.20 1.0

HLE 0.74 -0.28 0.50 0.50 1.3

FMLC 0.37 0.63 0.68 0.32 1.6

G 0.53 0.28 0.45 0.55 1.5

PC 0.82 0.15 0.77 0.23 1.1

MR1 MR2

SS loadings 2.44 1.49

Proportion Var 0.41 0.25

Cumulative Var 0.41 0.66

Proportion Explained 0.62 0.38

Cumulative Proportion 0.62 1.00

With factor correlations of

MR1 MR2

MR1 1.00 0.31

MR2 0.31 1.00

Mean item complexity = 1.2

Since Oblimin is an oblique rotation, we see that the correlation between factors is also included in the results. According to these results, the correlation between the MR1 and MR2 factors is 0.31. Additionally, because the average complexity is lower compared to previous rotations, we can say that the Oblimin rotation yields better results.

> data.frame(ulkeler = asd$Country,skor = f5$scores %\*% f5$Vaccounted[4,] ) %>% arrange(desc(skor))

|  |  |
| --- | --- |
| ulkeler skor  1 Denmark 1.26646376  2 Sweden 1.18942362  3 Luxembou 1.17219209  4 Finland 1.10978882  5 Ireland 1.07864080  6 Netherla 1.03313359  7 Austria 0.61895279  8 Estonia 0.46244151  9 Germany 0.41541786  10 France 0.30414372  11 Malta 0.28051233  12 Slovenia 0.20793693  13 Czechia 0.07675456  14 Spain -0.03428279 | 15 Belgium -0.13488530  16 Poland -0.27167940  17 Lithuani -0.45081991  18 Slovakia -0.46065458  19 Hungary -0.47538159  20 Portugal -0.49893229  21 Latvia -0.50369390  22 Croatia -0.57771839  23 Italy -0.60492411  24 Bulgaria -0.74471502  25 Cyprus -0.77717677  26 Romania -0.95724212  27 Greece -1.23539120  28 Turkey -1.48830501 |

**Promax rotation:**

> f6 = fa(scale(data1),nfactors = 2,rotate = "promax")

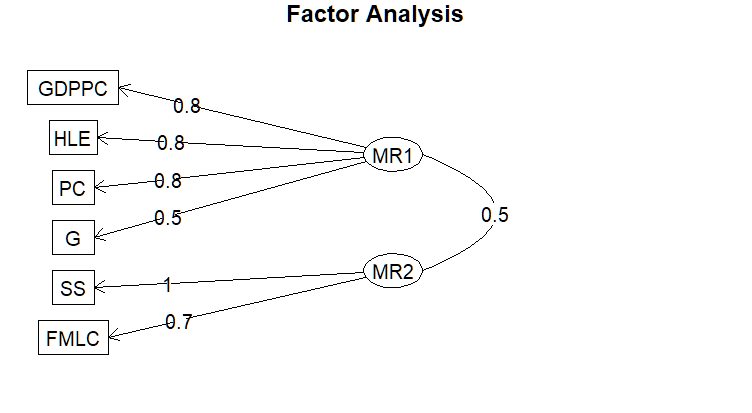
> f6

Factor Analysis using method = minres

Call: fa(r = scale(data1), nfactors = 2, rotate = "promax")

Standardized loadings (pattern matrix) based upon correlation matrix

MR1 MR2 h2 u2 com

GDPPC 0.84 0.03 0.73 0.27 1.0

SS -0.27 0.99 0.80 0.20 1.1

HLE 0.80 -0.28 0.50 0.50 1.2

FMLC 0.22 0.70 0.68 0.32 1.2

G 0.45 0.32 0.45 0.55 1.8

PC 0.77 0.19 0.77 0.23 1.1

MR1 MR2

SS loadings 2.25 1.68

Proportion Var 0.38 0.28

Cumulative Var 0.38 0.66

Proportion Explained 0.57 0.43

Cumulative Proportion 0.57 1.00

With factor correlations of

MR1 MR2

MR1 1.00 0.47

MR2 0.47 1.00

Mean item complexity = 1.2

Although the average complexity is the same as with oblimin, it is observed that the correlation between factors is higher. Additionally, the high complexity value of the G variable is also noticeable.

**Genel Sonuç:**

Aside from the factor analysis where we did not apply any rotation, in all other factor analyses, the variables GDPPC, HLE, PC, and G are seen under the factor named MR1, and the variables SS and FMLC are seen under the factor named MR2. Looking at the explanations of these variables, it is observed that the ones in the MR1 factor are generally related to the economy, while the MR2 factor is related to social life. Accordingly, we can name the MR1 factor as economy and the MR2 factor as social life. When we look at the overall performance of the rotations, we see that the lowest complexity value is 1.2 in the oblimin and promax rotations. Additionally, the factor loadings are higher in these two rotations. However, when looking at the complexity values of the variables, the more homogeneous distribution in the oblimin rotation and the lower correlation between factors indicate better performance of this rotation. Since these two rotations are oblique, the correlation between factors should also be considered when interpreting, which is a complicating factor. Therefore, a non-oblique rotation might also be preferred. Considering the factor loadings and complexity values, it can be said that the varimax rotation gives the best result. A choice should be made between varimax and oblimin rotation based on the need.