

Measurement in the Social Sciences (TT 2008)

Appendix 3.1: Factor Analysis Extended Examples - Stata

Dave Armstrong
University of Oxford
Department of Politics and International Relations

e: david.armstrong@politics.ox.ac.uk
w: <http://users.ox.ac.uk/~polf0104>

May 8, 2008

1 NES Data

For this section, we will use the same NES data we have been using the past couple of weeks. However, I've made a smaller dataset such that none of the variables have any missing data. These data are in `nes2004b.dta`. Again, let's look at the correlation matrix here:

	Zss	Zds	Zin	Zjl	Zab	Zej	Zfg	Zwr
Zspendserv	1.00							
Zdefspend	0.20	1.00						
Zinsurance	0.46	0.28	1.00					
Zjobsliv	0.46	0.28	0.52	1.00				
Zgabblacks	0.34	0.26	0.32	0.49	1.00			
Zenvjobs	0.24	0.22	0.25	0.22	0.21	1.00		
Zfedgun	0.20	0.23	0.21	0.19	0.18	0.10	1.00	
Zwomrole	0.20	0.12	0.19	0.15	0.14	0.22	0.17	1.00

We see generally middle-of-the-road correlations ranging from .1 to around .5. From this, we would have no particular reason to think the factor analysis model is inappropriate. We would probably expect to see one factor and it is possible that the last two variables, federal gun laws and women's role might not fit as well as the other variables. But, we'll need to see.

1.1 Nil Uniqueness Model (aka Principal Components Factors)

The easiest thing to do is factor the original correlation matrix. We can do this through Principal Components Factors, where we are assuming that all of every variable's variance is shared with the all other variables, that is, the factors should account for all of the variance among the variables. We can do this as follows in Stata:

```
. factor Z*, pcf mineigen(0) (obs=749)

Factor analysis/correlation                Number of obs =          749
Method: principal-component factors        Retained factors =         8
Rotation: (unrotated)                     Number of params =        28
```

Beware: solution is a Heywood case
(i.e., invalid or boundary values of uniqueness)

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	2.89185	1.88301	0.3615	0.3615
Factor2	1.00884	0.08357	0.1261	0.4876
Factor3	0.92527	0.08227	0.1157	0.6032
Factor4	0.84300	0.14488	0.1054	0.7086
Factor5	0.69812	0.02435	0.0873	0.7959
Factor6	0.67377	0.13882	0.0842	0.8801
Factor7	0.53495	0.11074	0.0669	0.9470
Factor8	0.42421	.	0.0530	1.0000

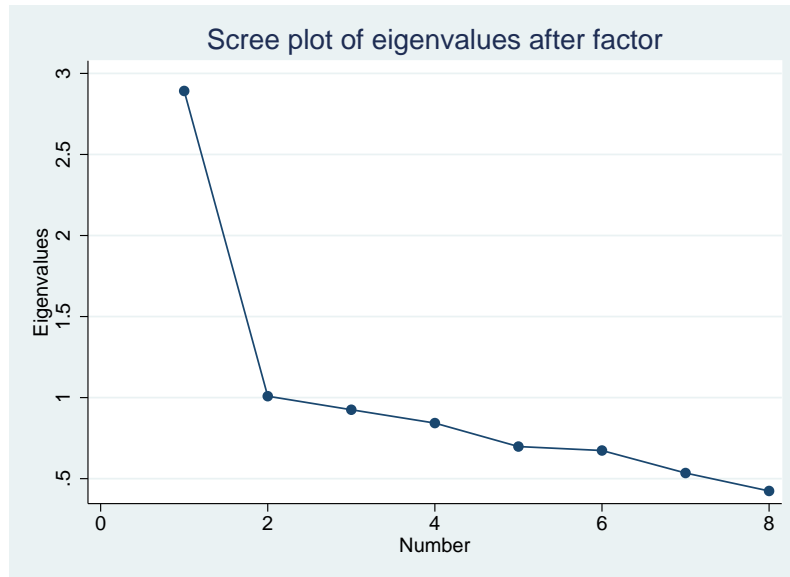
LR test: independent vs. saturated: $\chi^2(28) = 1054.96$ Prob> $\chi^2 = 0.0000$

Factor loadings (pattern matrix) and unique variances

Variable	Factor1	Factor2	Factor3	Factor4	Factor5	Factor6	Factor7	Factor8	Uniqueness
Zspendserv	0.6976	-0.1730	-0.1826	-0.2641	-0.2487	-0.0452	0.5621	0.0208	-0.0000
Zdefspend	0.5298	0.1394	0.4378	0.5393	0.1094	-0.4255	0.1549	-0.0186	0.0000
Zinsurance	0.7309	-0.1671	-0.0669	-0.1174	-0.3112	-0.2678	-0.3805	0.3260	0.0000
Zjobsliv	0.7588	-0.3439	-0.0500	-0.0742	0.0808	-0.0104	-0.2119	-0.4965	0.0000
Zgablacks	0.6550	-0.2791	0.0237	0.0785	0.5404	0.3540	0.0256	0.2613	0.0000
Zenvjobs	0.4899	0.3721	-0.3965	0.5093	-0.2707	0.3586	-0.0416	-0.0360	0.0000
Zfedgun	0.4322	0.3712	0.6613	-0.3002	-0.1826	0.3348	-0.0432	-0.0271	0.0000
Zwomrole	0.3984	0.6776	-0.3133	-0.3279	0.3496	-0.2298	-0.0327	-0.0203	0.0000

We have to use the `mineigen(0)` command to get Stata to give us all of the factors so there are zero uniqueness values. Stata defaults to retaining factors with eigenvalues greater than 1.0, regardless of whether or not this is a sound modelling decision. We can look at the scree plot in Figure 1.

Figure 1: Scree plot for PCF solution



As we saw before, this suggests a one-factor solution, which means if we're quite happy with this, we should rerun the analysis and retain one factor. We can also see this from the factor pattern coefficients. The pattern coefficients for factor 1 suggest that the first factor is positively related to everything. If we are proposing these are measures of left-right, then we would think that factor 1 would have the same direction of effect on each variable. The eigenvalues suggest we could probably ignore factor 2, but if the factor pattern coefficients for factor 2 highlight some substantively meaningful attribute of the variables included, then it might be worth estimating a two-factor solution.

```
. factor Z*, pcf fac(1) (obs=749)
```

```
Factor analysis/correlation                Number of obs =      749
Method: principal-component factors        Retained factors =    1
Rotation: (unrotated)                     Number of params =    8
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	2.89185	1.88301	0.3615	0.3615
Factor2	1.00884	0.08357	0.1261	0.4876
Factor3	0.92527	0.08227	0.1157	0.6032
Factor4	0.84300	0.14488	0.1054	0.7086
Factor5	0.69812	0.02435	0.0873	0.7959
Factor6	0.67377	0.13882	0.0842	0.8801
Factor7	0.53495	0.11074	0.0669	0.9470
Factor8	0.42421	.	0.0530	1.0000

```
LR test: independent vs. saturated:  chi2(28) = 1054.96 Prob>chi2 = 0.0000
```

```
Factor loadings (pattern matrix) and unique variances
```

Variable	Factor1	Uniqueness
Zspendserv	0.6976	0.5133

Zdefspend		0.5298		0.7193
Zinsurance		0.7309		0.4659
Zjobsliv		0.7588		0.4243
Zgablacks		0.6550		0.5709
Zenvjobs		0.4899		0.7600
Zfedgun		0.4322		0.8132
Zwomrole		0.3984		0.8413

Now, it doesn't seem like this model makes all that much sense to me. In the theoretical model of the principal components factor model, we are suggesting that there are no uniquenesses, that we can explain all of the combined variance, but by taking less than the total number of factors out, we can only explain about 36% of the variance and the uniqueness values are all greater than .4, so for every variable, at least 40% of its variance remains unexplained by the one-factor solution. This is not bad *per se*, but it does seem to contradict the theoretical model. Since we know of a theoretical model that does allow variables to have non-zero uniqueness values (both the principal factor and iterated principal factor models), then it would seem more appropriate to use one of these. Let's start with the principal factor model:

1.2 Principal Factor Model

```
. factor Z*, pf
(obs=749)
```

```
Factor analysis/correlation          Number of obs =      749
Method: principal factors           Retained factors =    4
Rotation: (unrotated)              Number of params =   26
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	2.18420	1.99315	1.1536	1.1536
Factor2	0.19105	0.11343	0.1009	1.2545
Factor3	0.07762	0.06328	0.0410	1.2955
Factor4	0.01434	0.05436	0.0076	1.3031
Factor5	-0.04001	0.09325	-0.0211	1.2819
Factor6	-0.13326	0.06435	-0.0704	1.2115
Factor7	-0.19761	0.00532	-0.1044	1.1072
Factor8	-0.20293	.	-0.1072	1.0000

```
LR test: independent vs. saturated: chi2(28) = 1054.96 Prob>chi2 = 0.0000
```

```
Factor loadings (pattern matrix) and unique variances
```

Variable	Factor1	Factor2	Factor3	Factor4	Uniqueness
Zspendserv	0.6157	-0.0628	-0.1145	-0.0259	0.6032
Zdefspend	0.4315	0.1346	0.1592	0.0054	0.7703
Zinsurance	0.6576	-0.0516	-0.0630	-0.0430	0.5591
Zjobsliv	0.7023	-0.1894	0.0170	0.0077	0.4706
Zgablacks	0.5695	-0.1019	0.0958	0.0536	0.6533
Zenvjobs	0.3936	0.1837	-0.0669	0.0676	0.8023
Zfedgun	0.3417	0.1732	0.1030	-0.0653	0.8384
Zwomrole	0.3122	0.2373	-0.1030	0.0053	0.8356

There is something strange going on here. We see negative eigenvalues, which allegedly correspond to “variance explained” by each factor. This would be like having a negative R^2 . It’s not theoretically possible. So, why does this happen? Well, that will take a bit of a digression.

1.2.1 How can a model explain more than 100% of the Variance

First, let’s remind ourselves what the factor model is trying to do: it is trying to reproduce the $k \times k$ correlation matrix with m -eigenvalues and eigenvectors. To the extent that we can effectively reproduce the correlation matrix with $m = 1$ or $m = 2$ we will have generally made a big reduction in the dimensionality of our data. Now, let’s remember back to what the principal factor model is doing. It is putting squared multiple correlations on the diagonal of the correlation matrix and is then factoring that matrix. So, in our data, the squared multiple correlations are as follows:

Table 1: Squared Multiple Correlations for NES Data

variable	SMC
Zspendserv	0.31
Zdefspend	0.16
Zinsurance	0.36
Zjobsliv	0.42
Zgablacks	0.28
Zenvjobs	0.13
Zfedgun	0.10
Zwomrole	0.09

That gives us the following adjusted correlation matrix $\tilde{\mathbf{R}}_{XX}$:

	Zss	Zds	Zin	Zjl	Zab	Zej	Zfg	Zwr
Zspendserv	0.31							
Zdefspend	0.20	0.16						
Zinsurance	0.46	0.28	0.36					
Zjobsliv	0.46	0.28	0.52	0.42				
Zgablacks	0.34	0.26	0.32	0.49	0.28			
Zenvjobs	0.24	0.22	0.25	0.22	0.21	0.13		
Zfedgun	0.20	0.23	0.21	0.19	0.18	0.10	0.10	
Zwomrole	0.20	0.12	0.19	0.15	0.14	0.22	0.17	0.09

We know that we’re trying to explain as much variance as is in the trace (sum of the diagonal elements) of $\tilde{\mathbf{R}}_{XX}$. In this case, that number is 1.89.

Now, we can factor this matrix using the SVD (eigen decomposition) as follows:

$$\mathbf{\Lambda}^2 = \begin{bmatrix} 2.184 & 0.000 & 0.000 & 0.000 & 0.00 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.191 & 0.000 & 0.000 & 0.00 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.078 & 0.000 & 0.00 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.014 & 0.00 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & -0.04 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.00 & -0.133 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.00 & 0.000 & -0.198 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.00 & 0.000 & 0.000 & -0.203 \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} 0.417 & 0.144 & 0.411 & -0.216 & 0.156 & -0.709 & 0.250 & 0.039 \\ 0.292 & -0.308 & -0.572 & 0.045 & -0.430 & -0.147 & 0.531 & -0.052 \\ 0.445 & 0.118 & 0.226 & -0.359 & -0.487 & 0.302 & -0.259 & -0.462 \\ 0.475 & 0.433 & -0.061 & 0.064 & 0.049 & 0.404 & 0.132 & 0.629 \\ 0.385 & 0.233 & -0.344 & 0.448 & 0.448 & -0.057 & -0.186 & -0.490 \\ 0.266 & -0.420 & 0.240 & 0.564 & -0.293 & -0.164 & -0.435 & 0.274 \\ 0.231 & -0.396 & -0.370 & -0.546 & 0.325 & -0.064 & -0.440 & 0.226 \\ 0.211 & -0.543 & 0.370 & 0.044 & 0.399 & 0.433 & 0.392 & -0.146 \end{bmatrix}$$

The eigenvalues are the diagonal elements of $\mathbf{\Lambda}^2$. You can see that the first factor explains a variance of 2.184, which is actually more than 1.89, the total variance we were trying to explain. Since this is the case, some of the eigenvalues have to be negative. If you sum all of the eigenvalues, they will sum to 1.89. We can see how this works if we actually do the matrix calculation for the one-factor solution. Remember, the following result:

$$\tilde{\mathbf{R}}_{XX} = \mathbf{V}\mathbf{\Lambda}^2\mathbf{V}'$$

In a one-factor solution, we take the first column of \mathbf{V} (we'll call that \mathbf{v}_1 , and the first eigenvalue λ_1^2 , where:

$$\mathbf{v}_1 = \begin{bmatrix} 0.417 \\ 0.292 \\ 0.445 \\ 0.475 \\ 0.385 \\ 0.266 \\ 0.231 \\ 0.211 \end{bmatrix}$$

We can now calculate:

$$\hat{\mathbf{R}}_{XX} = \mathbf{v}_1\lambda_1^2\mathbf{v}_1'$$

and we get:

$$\widehat{\mathbf{R}}_{XX} = \begin{bmatrix} 0.379 & 0.266 & 0.405 & 0.432 & 0.351 & 0.242 & 0.210 & 0.192 \\ 0.266 & 0.186 & 0.284 & 0.303 & 0.246 & 0.170 & 0.147 & 0.135 \\ 0.405 & 0.284 & 0.432 & 0.462 & 0.374 & 0.259 & 0.225 & 0.205 \\ 0.432 & 0.303 & 0.462 & 0.493 & 0.400 & 0.276 & 0.240 & 0.219 \\ 0.351 & 0.246 & 0.374 & 0.400 & 0.324 & 0.224 & 0.195 & 0.178 \\ 0.242 & 0.170 & 0.259 & 0.276 & 0.224 & 0.155 & 0.134 & 0.123 \\ 0.210 & 0.147 & 0.225 & 0.240 & 0.195 & 0.134 & 0.117 & 0.107 \\ 0.192 & 0.135 & 0.205 & 0.219 & 0.178 & 0.123 & 0.107 & 0.097 \end{bmatrix}$$

where the sum of the diagonal elements of $\widehat{\mathbf{R}}_{XX}$ equals 2.18. This tells us that the one-dimensional solution explains more than what we had calculated the total amount of variance would be. We will see shortly that the iterated principal factor model gets us out of this seemingly anomalous result.

1.3 Iterated Principal Factor Model

We just saw that the Principal Factor model generates a somewhat anomalous result, so since there is a model (IPF) that gets us out of that situation, there is no particularly good reason not to use that model. Here is a thumbnail sketch of how the IPF model works:

1. Generate the correlation matrix \mathbf{R}_{XX} .
2. Replace the diagonal elements of \mathbf{R}_{XX} with squared multiple correlations and re-name this matrix $\tilde{\mathbf{R}}_{XX}$.
3. Factor $\tilde{\mathbf{R}}_{XX} = \mathbf{V}\mathbf{\Lambda}^2\mathbf{V}'$
4. Take m columns of \mathbf{V} and the $m \times m$ diagonal matrix of the first m eigenvalues $(\lambda_1^2, \lambda_2^2, \dots, \lambda_m^2)$ and generate $\widehat{\mathbf{R}}_{XX} = \mathbf{v}_m\mathbf{\Lambda}_m^2\mathbf{v}_m'$.
5. Take the diagonal elements of $\widehat{\mathbf{R}}_{XX}$ and put them on the diagonal of \mathbf{R}_{XX} making a new $\tilde{\mathbf{R}}_{XX}$.
6. Repeat steps 3-5 until there are minimal changes in the diagonal elements between iterations.

We can see the results of this model:

```
. factor Z*, ipf fac(1) (obs=749)

Factor analysis/correlation          Number of obs =      749
Method: iterated principal factors   Retained factors =    1
Rotation: (unrotated)               Number of params =    8
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	2.25013	2.02929	1.0000	1.0000
Factor2	0.22084	0.10241	0.0981	1.0982
Factor3	0.11843	0.07131	0.0526	1.1508
Factor4	0.04712	0.04844	0.0209	1.1717
Factor5	-0.00132	0.07197	-0.0006	1.1711
Factor6	-0.07329	0.05119	-0.0326	1.1386
Factor7	-0.12448	0.06287	-0.0553	1.0833
Factor8	-0.18734	.	-0.0833	1.0000

LR test: independent vs. saturated: $\chi^2(28) = 1054.96$ Prob> $\chi^2 = 0.0000$

Factor loadings (pattern matrix) and unique variances

Variable	Factor1	Uniqueness
Zspendserv	0.6294	0.6039
Zdefspend	0.4281	0.8168
Zinsurance	0.6767	0.5421
Zjobsliv	0.7253	0.4739
Zgablacks	0.5732	0.6714
Zenvjobs	0.3892	0.8486
Zfedgun	0.3365	0.8868
Zwomrole	0.3058	0.9065

This model suggests that one factor can explain all of the shared variance, because we're defining the shared variance as the amount of variance explainable by m factors, 1 in this case. This gets us out of the problem of explaining more than the total amount of variance. We can see from the table below that the communality estimates generally get better. Remember, squared multiple correlations are the theoretical bound of communality estimates if the model is perfectly specified (i.e., you have the right number of factors and each variable belongs in the model).

Table 2: Initial and Final Communality Estimates

variable	Initial	Final	Final - Initial
Zspendserv	0.316	0.396	0.080
Zdefspend	0.163	0.183	0.020
Zinsurance	0.363	0.458	0.095
Zjobsliv	0.424	0.526	0.102
Zgablacks	0.283	0.329	0.046
Zenvjobs	0.138	0.151	0.013
Zfedgun	0.108	0.113	0.005
Zwomrole	0.098	0.094	-0.005

The women's role variable doesn't get any better and remains pretty low. This might suggest that this variable doesn't belong in the model. I would probably try removing this variable and see what happens. I don't suspect that it will change the nature of the solution that much.

```
Factor analysis/correlation          Number of obs =      749
Method: iterated principal factors   Retained factors =    1
Rotation: (unrotated)                Number of params =    7
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	2.16281	2.00069	1.0000	1.0000
Factor2	0.16212	0.06376	0.0750	1.0750
Factor3	0.09836	0.08138	0.0455	1.1205
Factor4	0.01699	0.03376	0.0079	1.1283
Factor5	-0.01678	0.09550	-0.0078	1.1205
Factor6	-0.11228	0.03617	-0.0519	1.0686
Factor7	-0.14844	.	-0.0686	1.0000

LR test: independent vs. saturated: $\chi^2(21) = 978.19$ Prob> $\chi^2 = 0.0000$

Factor loadings (pattern matrix) and unique variances

Variable	Factor1	Uniqueness
Zspendserv	0.6243	0.6102
Zdefspend	0.4273	0.8174
Zinsurance	0.6776	0.5408
Zjobsliv	0.7435	0.4472
Zgablacks	0.5801	0.6635
Zenvjobs	0.3700	0.8631
Zfedgun	0.3241	0.8950

As you can see, this doesn't change the solution that much, so removing the women's role variable probably makes sense. Now, one thing to remember is that this model, even though it looks good, is only explaining $2.16/7 = .308$ of the total variance. That means that this model suggests most of each variable's variance is a function of something other

than this underlying factor. This is neither bad nor good, but we should not lose sight of this figure just because we have rather large factor pattern coefficients.

1.4 Comparing Results

So far, we have been making decisions about which model to use based on which one best fits our theoretical understanding of the relationships we're trying to model. However, lest we lose sight of the similarities here, we should probably take a look to see how similar these results are. The table below shows the factor pattern coefficients for each model.

Table 3: Comparing Factor Pattern Coefficients

Variable	PCF	PF	IPF
Zspendserv	0.70	0.61	0.62
Zdefspend	0.53	0.43	0.42
Zinsurance	0.73	0.65	0.67
Zjobsliv	0.77	0.71	0.74
Zgablacks	0.66	0.57	0.58
Zenvjobs	0.47	0.37	0.37
Zfedgun	0.42	0.32	0.32

We tend to see the same types of trends here, so it would be difficult to suggest that any one of these methods is going to lead you very far astray. Let's take a look at the correlation matrix between the factor pattern coefficients from each model.

	PCF	PF	IPF
PCF	1.000	0.999	0.996
PF	0.999	1.000	0.999
IPF	0.996	0.999	1.000

We can also see the same sort of thing happening if we ask Stata to give us the estimated factor scores, which we can do with the following set of commands:

```
. factor Zspendserv-Zfedgun, pcf fac(1)
. predict pcfF1
. factor Zspendserv-Zfedgun, pf fac(1)
. predict pfF1
. factor Zspendserv-Zfedgun, ipf fac(1)
. predict ipfF1

. cor pcfF1-ipfF1 (obs=749)
```

	pcfF1	pfF1	ipfF1
pcfF1	1.0000		
pfF1	0.9951	1.0000	
ipfF1	0.9908	0.9993	1.0000

So this tells us that even though these are somewhat different models, they are producing roughly the same thing. Now, this can be interpreted in two different ways. The first, and I think less attractive, is to use whatever model you want because the results will generally be similar. The second, and I think more attractive, is to use the model that makes the most theoretical sense, which I would suggest is IPF. This way, you'll never be caught in the situation where you're asked why you chose a certain extraction method and you're left to say I chose X because it was easiest and didn't give substantively different results from a more sensible model.

2 Two Factor Model of Democracy and Repression

I will take the advice from above with respect to factor extraction and will apply that to a pretty clear two-factor solution. I do this to illustrate the points you need to be aware of with factor rotation. Open the data demrep.dta

```
factor Z*, ipf (obs=2961)
```

```
Factor analysis/correlation          Number of obs =      2961
Method: iterated principal factors   Retained factors =    9
Rotation: (unrotated)                Number of params =   45
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	4.91326	3.27781	0.6667	0.6667
Factor2	1.63545	1.20009	0.2219	0.8886
Factor3	0.43536	0.22598	0.0591	0.9477
Factor4	0.20938	0.11964	0.0284	0.9761
Factor5	0.08973	0.02304	0.0122	0.9883
Factor6	0.06670	0.05021	0.0091	0.9974
Factor7	0.01648	0.01360	0.0022	0.9996
Factor8	0.00288	0.00251	0.0004	1.0000
Factor9	0.00038	0.00064	0.0001	1.0000
Factor10	-0.00026	.	-0.0000	1.0000

```
LR test: independent vs. saturated:  chi2(45) = 2.1e+04 Prob>chi2 = 0.0000
```

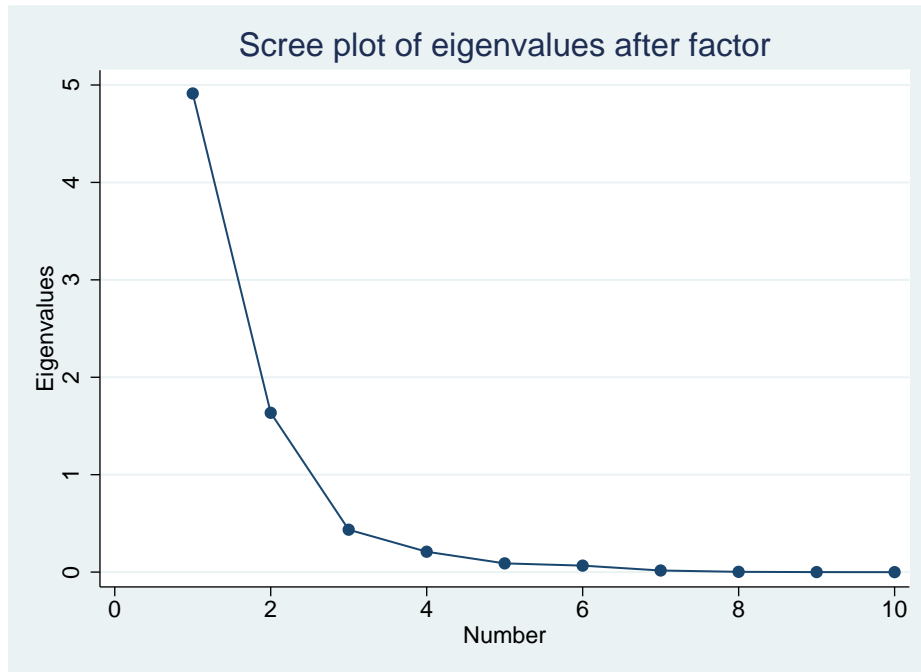
Factor loadings (pattern matrix) and unique variances

Variable	Factor1	Factor2	Factor3	Factor4	Factor5	Factor6
Zpolconii	0.8752	-0.1877	-0.2578	-0.1286	0.0259	0.0093
Zpolconv	0.8972	-0.0626	-0.2364	-0.1225	0.0266	-0.0908
Zchecks	0.7463	-0.1832	-0.0594	0.0591	0.0061	0.1891
Zeiec	0.8214	-0.2839	0.3711	0.0512	-0.0305	0.0492
Zliec	0.7673	-0.3289	0.3284	-0.0138	0.0970	-0.1004
Zxconst	0.8688	-0.0675	-0.1604	0.0609	-0.0521	0.0013
Zdisap	0.3268	0.5967	0.1559	-0.2232	-0.1344	0.0418
Zkill	0.3642	0.7530	0.0951	-0.1183	0.1427	0.0187
Zpolpris	0.6218	0.3520	0.0076	0.1739	-0.1630	-0.0892
Ztort	0.3754	0.5678	-0.0684	0.2716	0.1013	0.0072

Variable	Factor7	Factor8	Factor9	Uniqueness
Zpolconii	0.0622	-0.0148	-0.0094	0.1108
Zpolconv	-0.0149	0.0333	0.0041	0.1099
Zchecks	0.0214	-0.0020	0.0091	0.3662
Zeiec	-0.0171	0.0188	-0.0104	0.1002
Zliec	0.0208	-0.0139	0.0079	0.1748
Zxconst	-0.0942	-0.0182	0.0000	0.1992
Zdisap	0.0111	0.0077	0.0035	0.4430
Zkill	-0.0258	-0.0146	-0.0027	0.2557
Zpolpris	0.0342	-0.0121	0.0005	0.4234
Ztort	0.0195	0.0160	-0.0008	0.4472

Now, we can look at the scree plot to see whether the two-factor solution makes sense.

Figure 2: Scree plot for Democracy-Repression Factor Analysis



This is a pretty clear two-factor solution, so we can go back and estimate this with just two factors:

```
. factor Z*, ipf fac(2) (obs=2961)
```

```
Factor analysis/correlation          Number of obs=      2961
Method: iterated principal factors    Retained factors =    2
Rotation: (unrotated)                Number of params =   19
```

Factor	Eigenvalue	Difference	Proportion	Cumulative
Factor1	4.81098	3.27173	0.7576	0.7576
Factor2	1.53925	1.24972	0.2424	1.0000
Factor3	0.28953	0.18164	0.0456	1.0456
Factor4	0.10788	0.09297	0.0170	1.0626
Factor5	0.01491	0.03362	0.0023	1.0649
Factor6	-0.01871	0.01672	-0.0029	1.0620
Factor7	-0.03543	0.05597	-0.0056	1.0564
Factor8	-0.09140	0.00306	-0.0144	1.0420
Factor9	-0.09447	0.07788	-0.0149	1.0271
Factor10	-0.17234	.	-0.0271	1.0000

```
LR test: independent vs. saturated:  chi2(45) = 2.1e+04 Prob>chi2 = 0.0000
```

```
Factor loadings (pattern matrix) and unique variances
```

Variable	Factor1	Factor2	Uniqueness
Zpolconii	0.8661	-0.1896	0.2140
Zpolconv	0.8897	-0.0689	0.2036
Zchecks	0.7479	-0.1904	0.4043
Zeiec	0.7954	-0.2606	0.2993
Zliec	0.7465	-0.3055	0.3495
Zxconst	0.8720	-0.0761	0.2338
Zdisap	0.3223	0.5648	0.5772
Zkill	0.3667	0.7561	0.2938
Zpolpris	0.6192	0.3405	0.5007
Ztort	0.3712	0.5373	0.5735

With one factor, there are lots of concerns that you don't have to worry about. You don't have to worry about simple structure or rotation. Here, however, we do have to worry about these things. Remember, simple structure is a situation where factor pattern coefficients are as close to zero or one as possible. Often times people want to maintain the orthogonality of the solution because it is a bit easier to interpret. However, if it is not true to reality, then you've got a choice between an inferior solution that is easier to interpret or a superior solution that is a bit more tricky to interpret. In general, the latter trumps the former - just think of the difference between logit and OLS for binary dependent variables.

2.1 Varimax Rotation

The varimax rotation is supposed to create something close to simple structure while maintaining the uncorrelated nature of the factors. We can get Stata to do the rotation the following way:

```
. rotate, varimax
```

```
Factor analysis/correlation          Number of obs =      2961
Method: iterated principal factors    Retained factors =      2
Rotation: orthogonal varimax (Horst off) Number of params =     19
```

Factor	Variance	Difference	Proportion	Cumulative
Factor1	4.41492	2.47961	0.6952	0.6952
Factor2	1.93531	.	0.3048	1.0000

```
LR test: independent vs. saturated:  chi2(45) = 2.1e+04 Prob>chi2 = 0.0000
```

Rotated factor loadings (pattern matrix) and unique variances

Variable	Factor1	Factor2	Uniqueness
Zpolconii	0.8779	0.1236	0.2140
Zpolconv	0.8581	0.2450	0.2036
Zchecks	0.7674	0.0817	0.4043
Zeiec	0.8364	0.0324	0.2993
Zliec	0.8061	-0.0266	0.3495
Zxconst	0.8440	0.2321	0.2338
Zdisap	0.1057	0.6416	0.5772
Zkill	0.0807	0.8365	0.2938
Zpolpris	0.4620	0.5346	0.5007
Ztort	0.1611	0.6329	0.5735

Factor rotation matrix

	Factor1	Factor2
Factor1	0.9375	0.3479
Factor2	-0.3479	0.9375

I argue that you shouldn't really stop here unless you're either 1) sure or 2) completely wedded to the assumption that these two concepts are uncorrelated. If you're not, you don't lose anything from looking at the promax rotation (the one that allows the factors to be correlated). Here, you can see that the factor pattern coefficients are getting closer to one and zero, but there are still some sizeable coefficients in the blocks that we might like to be 0.

2.2 Promax Rotation

The promax rotation starts with a varimax rotation and created a "target matrix" by raising those coefficients to some power (the default is usually 3 and there isn't a particularly good reason to pick something other than that). This should be an odd

number to retain the sign of the pattern coefficients. Then, Stata tries to find a rotation matrix by which it can multiply the current pattern coefficients to get to the target matrix.

```
. rotate, promax
```

```
Factor analysis/correlation          Number of obs
=      2961
Method: iterated principal factors   Retained factors =      2
Rotation: oblique promax (Horst off) Number of params =    19
```

Factor	Variance	Proportion	Rotated factors are correlated
Factor1	4.67062	0.7355	
Factor2	2.44314	0.3847	

```
LR test: independent vs. saturated:  chi2(45) = 2.1e+04 Prob>chi2 = 0.0000
```

Rotated factor loadings (pattern matrix) and unique variances

Variable	Factor1	Factor2	Uniqueness
Zpolconii	0.8887	-0.0065	0.2140
Zpolconv	0.8427	0.1242	0.2036
Zchecks	0.7824	-0.0333	0.4043
Zeiec	0.8646	-0.0959	0.2993
Zliec	0.8454	-0.1532	0.3495
Zxconst	0.8307	0.1128	0.2338
Zdisap	-0.0240	0.6580	0.5772
Zkill	-0.0908	0.8667	0.2938
Zpolpris	0.3695	0.4902	0.5007
Ztort	0.0355	0.6402	0.5735

Factor rotation matrix

	Factor1	Factor2
Factor1	0.9783	0.5256
Factor2	-0.2071	0.8507

2.3 Comparing Rotation Schemes

One thing we can do, is compare the factor pattern coefficients:

Table 4: Comparison of Original, Varimax and Promax Rotated Factor Pattern Coefficients

Variable	Factor 1			Factor 2		
	Orig	Varimax	Promax	Orig	Varimax	Promax
Zpolconii	0.866	0.877	0.888	-0.189	0.123	-0.006
Zpolconv	0.889	0.858	0.842	-0.068	0.245	0.124
Zchecks	0.747	0.767	0.782	-0.190	0.081	-0.033
Zeiec	0.795	0.836	0.864	-0.260	0.032	-0.095
Zliec	0.746	0.806	0.845	-0.305	-0.026	-0.153
Zxconst	0.872	0.844	0.830	-0.076	0.232	0.112
Zdisap	0.322	0.105	-0.024	0.564	0.641	0.658
Zkill	0.366	0.080	-0.090	0.756	0.836	0.866
Zpolpris	0.619	0.462	0.369	0.340	0.534	0.490
Ztort	0.371	0.161	0.035	0.537	0.632	0.640

What I would take away from this is that the promax rotation seems to have done a nice job on the first factor, but there is less strong evidence for the second factor. We can also look to see what the correlation is between the factors:

```
. estat common
```

Correlation matrix of the promax(3) rotated common factors

```
-----
      Factors |  Factor1  Factor2
-----+-----
      Factor1 |          1
      Factor2 |        .338          1
-----
```

So, when we allow the factors to be correlated, the model estimates correlation at 0.34. This is not a particularly strong correlation, but it's definitely not 0. This is, again, a judgement call. If you really think these two things are uncorrelated, you can use the varimax rotation. However, if you are willing to submit that they might be correlated, then you can use the promax rotation.

One more thing you can look at are the residuals - that is the difference between the adjusted correlation matrix and the reproduced correlation matrix. To the extent that the residuals are a lot bigger (in absolute value) than zero, it means the model is having a hard time reproducing the original correlations and possibly that the variable does not belong in your solution.

```
. estat residuals
```

```
Raw residuals of correlations (observed-fitted)
```

Variable	Zpolc~i	Zpolc~v	Zchecks	Zeiec	Zliec	Zxconst	Zdisap	Zkill	Zpolp~s	Ztort
Zpolconii	0.0000									
Zpolconv	0.0885	0.0000								
Zchecks	0.0146	-0.0081	0.0000							
Zeiec	-0.0699	-0.0694	0.0100	0.0000						
Zliec	-0.0510	-0.0412	-0.0218	0.1629	0.0000					
Zxconst	0.0301	0.0324	0.0051	-0.0341	-0.0457	0.0000				
Zdisap	-0.0121	-0.0087	-0.0141	0.0424	0.0236	-0.0272	0.0000			
Zkill	-0.0036	-0.0005	-0.0053	0.0168	0.0335	-0.0237	0.0456	0.0000		
Zpolpris	-0.0206	-0.0119	-0.0061	0.0185	-0.0028	0.0172	0.0023	-0.0385	0.0000	
Ztort	-0.0113	-0.0067	0.0233	-0.0222	-0.0287	0.0251	-0.0459	-0.0028	0.0506	0.0000

These actually look pretty good. The one correlation we're not able to account for is the one between eiec and liec. That is a bit of a strange result and it might be worth considering taking one of those variables out and see what the solution looks like. I would probably take out liec because it loads on both dimensions and we're trying to create simple structure. I'll leave that exercise for the interested reader.

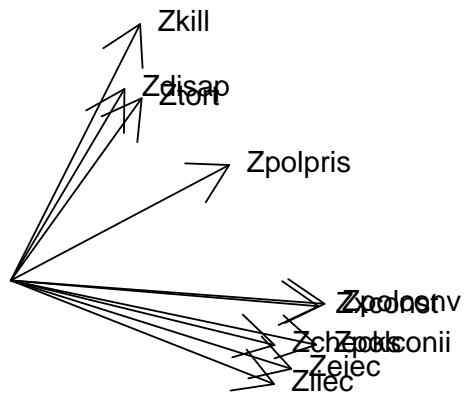
2.4 Graphical Representation of Rotation

We can also present our factor solution graphically. First, let's set a couple of parameters about what the graph will look like.

- Arrows represent variables in 2-dimensional factor space. The length of the arrow is equal to the variable's communality.
- The arrows will be arranged in such a way that the angle between each arrow represents the 2-dimensional correlation between each variable. The particular orientation is not important, only the relative positioning of the variable arrows (vectors) is important.
- The smaller the angle between any two variables (or factors), the bigger the correlation. Arrows at 90 degree angles are uncorrelated ($r=0$).

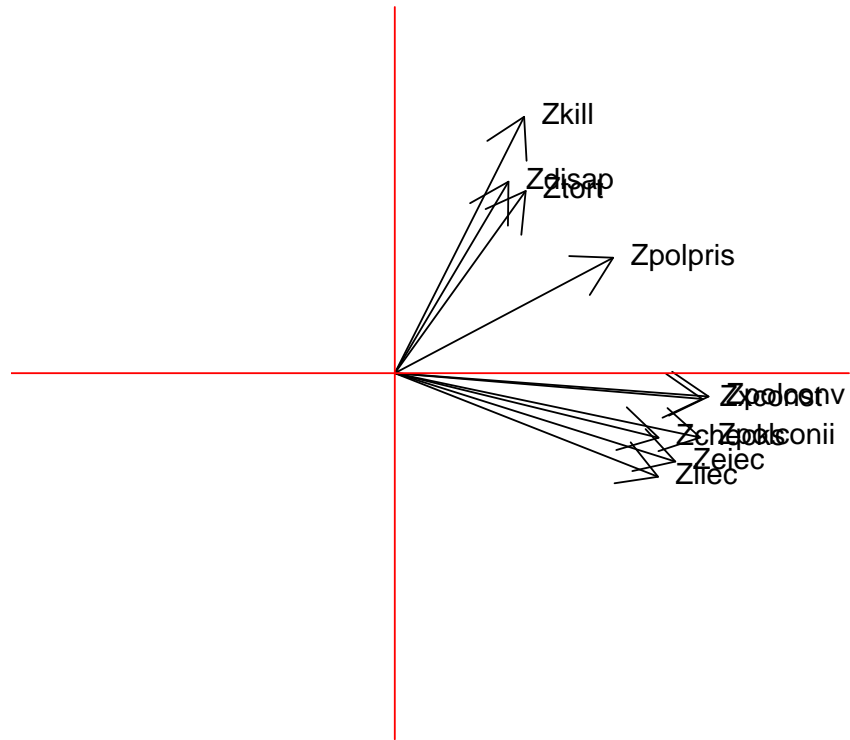
So, we can plot these variable vectors as follows:

Figure 3: Variable vectors in 2-dimensional space



What we ultimately want is for one factor to be running through each cluster of arrows. Right now, the factors are horizontal and vertical lines, we can impose those factors like this:

Figure 4: Original factor solution



You can see, the factors are not running through either cluster. The varimax rotation will make this happen as much as possible subject to the constraint that the factors remain uncorrelated. We can plot the varimax solution like this:

Figure 5: Varimax rotated factor solution



Now, the first factor is running through the bottom cluster, but the second factor is still not running through the top cluster. This means that we might gain something by allowing the factors to be correlated. We can see this as follows:

Figure 6: Promax rotated factor solution

