



Tutorials and worked examples for simulation,
curve fitting, statistical analysis, and plotting.

<https://simfit.org.uk>

<https://simfit.silverfrost.com>

Generalized orthomax rotation techniques can be used to simplify the interpretation of loading matrices, e.g. from canonical variates or factor analysis. These are only unique up to rotation so, by applying rotations according to stated criteria, different contributions of the original variables can be assessed.

Example 1

From the main SIMFIT menu choose [Statistics], [Multivariate], then [Varimax and Quartimax] and analyze the default test file provided, g03ba.f, to obtain the following results.

Results from Varimax rotation

Number of rows: 10

Number of columns: 3

Type: Unstandardised

Scaling: Varimax ($\gamma = 1$)

Original data Λ

0.788	-0.152	-0.352
0.874	0.381	0.041
0.814	-0.043	-0.213
0.798	-0.170	-0.204
0.641	0.070	-0.042
0.755	-0.298	0.067
0.782	-0.221	0.028
0.767	-0.091	0.358
0.733	-0.384	0.229
0.771	-0.101	0.071

Rotation matrix R

0.63347	-0.53367	-0.56029
0.75803	0.57333	0.31095
0.15529	-0.62169	0.76772

Rotated matrix $\Lambda^* = \lambda R$

0.32929	-0.28884	-0.75901
0.84882	-0.27348	-0.33974
0.44997	-0.32664	-0.63297
0.34496	-0.39651	-0.65659
0.45259	-0.27584	-0.36962
0.26278	-0.61542	-0.46424
0.33219	-0.56144	-0.48537
0.47248	-0.68406	-0.18319
0.20881	-0.75370	-0.35429
0.42287	-0.51350	-0.40888

The input loading matrix Λ has m rows and k columns and results from the analysis of an original data matrix with n rows (i.e. cases) and m columns (i.e. variables), where k factors have been calculated for $k \leq m$.

Example 2

If the input loading matrix is not standardized to unit length rows, this can be done interactively, as in the next example which also illustrates the use of Varimax rotation to simplify the interpretation of loadings containing negative values by forming a rotated loading matrix with most values positive.

Results from Varimax rotation

Number of rows: 10

Number of columns: 2

Type: Row standardized

Scaling: Varimax ($\gamma = 1$)

Original data Λ

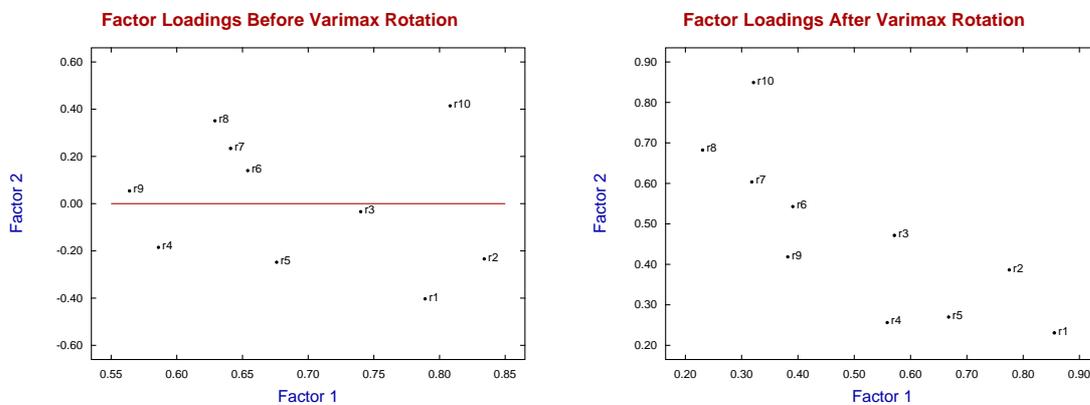
0.789	-0.403
0.834	-0.234
0.740	-0.034
0.586	-0.185
0.676	-0.248
0.654	0.140
0.641	0.234
0.629	0.351
0.564	0.054
0.808	0.414

Rotation matrix R

0.7279	0.6857
-0.6857	0.7279

Rotated matrix $\Lambda^* = \lambda R$

0.8506	0.2477
0.7675	0.4016
0.5619	0.4827
0.5534	0.2672
0.6621	0.2830
0.3800	0.5504
0.3061	0.6099
0.2171	0.6868
0.3735	0.4260
0.3042	0.8554



In these diagrams it is clear that factor 1 loads most heavily on variables 1 to 5 (i.e. as rows 1 to 5 of the loading matrix with symbols r1 to r5), while factor 2 loads most heavily on variables 6 to 10 (i.e. as rows 6 to 10 of the loading matrix with symbols r6 to r10). However it should be noted that before rotation there were both positive and negative loadings in the left hand figure as emphasized by the red line, while after rotation all loadings are now positive. So, the dependence of factors and loadings may be thought easier to appreciate after the rotation to make all values positive as in the right hand figure.

The SIMFIT Varimax procedure is made available when loading matrices are calculated and provides the facility to plot any selection of two or three columns of a loading matrix before or after rotation.

Theory

The rotated matrix Λ^* is calculated so that the elements λ_{ij}^* are either relatively large or small. This involves maximizing the function

$$V = \sum_{j=1}^k \sum_{i=1}^m (\lambda_{ij}^*)^4 - \frac{\gamma}{m} \sum_{j=1}^k \left[\sum_{i=1}^m (\lambda_{ij}^*)^2 \right]^2$$

where there were m variables originally and k factors were chosen for $k < m$.

There are several cases as follows

- Varimax rotation: $\gamma = 1$
- Quartimax rotation: $\gamma = 0$.
- Equamax rotation: $\gamma = k/2$.
- Parsimax rotation: $\gamma = m(k - 1)/(m + k + 2)$.
- User chosen rotation: γ input.

The resulting rotation matrix R satisfies $\Lambda^* = \Lambda R$ and, when the matrices have been calculated they can be viewed, written to the results log file, saved to a text file, or plotted.