

FA*i*R: A Package for FAiR

Ben Goodrich (goodrich@fas.harvard.edu)

The primary objective of FA*i*R is to provide functionality that is not available in closed-source factor analysis software, but FA*i*R also strives to integrate the various tools for factor analysis that are already available in R packages and to provide a reasonably user-friendly GUI (based on gWidgets) so that people who have more experience with factor analysis than with R can readily estimate their models. The first version of FA*i*R was released in February 2008, and the second version will have been released in April 2008.

FA*i*R is unique in that it utilizes a genetic algorithm (rgenoud) for constrained optimization, which permits new approaches to exploratory and confirmatory factor analysis (EFA and CFA) and also straightforwardly leads to a new estimator of the common factor analysis model called semi-exploratory factor analysis (SEFA). The common factor analysis model in the population can be written as $\Sigma = \Lambda\Phi\Lambda' + \Psi$, where Σ is a covariance matrix among n observable variables, Φ is a correlation matrix among r common factors, Λ is a $n \times r$ matrix of factor loadings, and Ψ is a (typically diagonal) covariance matrix among n unique variances. However, Λ and Φ are not separately identified unless additional restrictions are imposed. For example, CFA requires the analyst to specify which cells of Λ are zero *a priori*. SEFA differs by requiring the analyst to specify the *number* of zeros in each column of Λ but does not require the analyst to specify *where* the zeros occur. SEFA thus uses a genetic algorithm to maximize the fit to the data over the locations of these exact zeros in Λ and the values of the corresponding non-zero parameters.

FA*i*R also differs from all other factor analysis software in that the analyst can impose a wide variety of (non-linear) inequality restrictions on (functions of) parameters in SEFA and CFA models and also during the transformation stage of EFA models. For example, Louis Thurstone — who was the father of exploratory factor analysis with multiple factors — proposed a criterion for factor transformation in 1935 that had never been implemented by any factor analysis software, largely due to its perceived computational difficulty. Optimizing with respect to Thurstone's criterion is implemented in FA*i*R, which is fairly easy and very reliable due to power of the underlying genetic algorithm.

FA*i*R utilizes S4 classes, which not only facilitates post-estimation analysis but also provides a framework that allows rapid development and easy integration of other R packages with FA*i*R. My goals for useR are to attract the interest of additional developers and to expose attendees (and their colleagues at home) to the new ways of thinking that are embodied in FA*i*R but are unavailable in traditional factor analysis software.