faoutlier: An R Package for Detecting Outliers and Influential Cases in Exploratory and Confirmatory Factor Analysis

R. Philip Chalmers & David B. Flora
York University

Abstract

Like any statistical modeling procedure, results from both exploratory and confirmatory factor analysis are vulnerable to disproportionate influence from unusual or outlying cases. Because the common factor model is a type of multiple regression model, concepts from regression case diagnostics can be applied to factor analysis. However, extant factor analysis software does not implement these case diagnostic methods. In this paper, we briefly review these methods and present a new `R` package called `faoutlier` (Chalmers, 2011) that implements a number of case diagnostic measures, including Mahalanobis distance, factor model-based outliers, likelihood distance, and generalized Cook's distance, as well as a forward search procedure.

Keywords: exploratory factor analysis, confirmatory factor analysis, outliers, influence, forward search, R

Like any statistical modeling procedure, results from both exploratory factor analysis (EFA) and confirmatory factor analysis (CFA) are vulnerable to disproportionate influence from unusual or outlying cases. Nonetheless, the importance of data screening is often ignored or misconstrued in empirical research articles utilizing factor analysis. Perhaps some researchers have an overly indiscriminate impression that, as a large sample procedure, factor analysis is generally "robust" to the influence of unusual observations, or they may simply be unaware of this issue. Additionally, popular software packages for either EFA or CFA tend not to have built-in routines for examining outliers and potentially influential cases in the same sense that such routines are commonly available in ordinary regression modeling procedures. Nonetheless, as we describe below, principles from regression modeling diagnostics apply analogously to factor analysis. Thus, the purposes of this paper are to provide a brief review of case-diagnostic methods for factor analysis and to present an `R` (R Core Team, 2014) package called `faoutlier` (Chalmers, 2011) that implements these methods. `faoutlier` contains functions for calculating robust Mahalanobis distance statistics, case-level residuals, generalized Cook's distance, and likelihood distance, as well as a forward-search algorithm, each of which is described below. By making this set of computerized functions readily accessible to researchers in a freely available package, we hope that screening data for outlying and highly influential observations will become a more common practice in factor analysis.

Given its rapid rise in popularity, we assume that the reader has a basic familiarity with data analysis using the freely available statistical computing environment R; if not, there are numerous introductory books (e.g., Dalgaard, 2008) and web guides (e.g., Revelle, 2007) available. To use the `faoutlier` package in `R`, it must first be installed (using `install.packages(’faoutlier’)`) and loaded (using `library(faoutlier)` or `require(faoutlier)`). The `faoutlier` package comes with two data sets, `holzinger` and `holzinger.outlier`, which we use below to illustrate the package's functions. The nine-variable `holzinger` data consist of $N = 101$ simulated cases which were sampled from a standardized multivariate normal distribution. The generating correlation matrix for these simulated data matches a correlation matrix in Browne, Cudeck, Tateneni, and Mels (2008) based on data due to Holzinger that was published in Harman (1960). A three-factor EFA model fits both the original correlation matrix and our simulated data very well. The `holzinger.outlier` data were created by replacing the first case from `holzinger` with a case that equaled the original case with $Z = 2$ added to its values for five of the observed variables and $Z = 2$ subtracted from the other four observed variables. This case then shows up as an influential outlier in subsequent analyses. See Flora, LaBrish, and Chalmers (2012) for further detail and factor analyses of both data sets.

The Common Factor Model

Thurstone (1947) common factor model is the basis for modern EFA and CFA (MacCallum, 2009); Lawley and Maxwell (1963) showed that it can be expressed as a linear model with observed variables as dependent variables and factors as predictors:

$$y_j = \lambda_1 \eta_1 + \lambda_2 \eta_2 + \ldots + \lambda_m \eta_m + \epsilon_j, \tag{1}$$

where y_j is the jth observed variable from a battery of p observed variables, η_k is the kth of m common factors, λ_k is the regression coefficient, or factor loading, relating factor k to y_j, and j is the residual, or unique factor, for y_j. In matrix form the model is:

$$\mathbf{y} = \mathbf{\Lambda} \eta + \epsilon, \tag{2}$$

where y is a vector of the p observed variables, $\mathbf{\Lambda}$ is a $p \times m$ matrix of factor loadings, η is a vector of m common factors, and ϵ is a vector of unique factors. As with the general linear model, the residuals are assumed to be independent of the predictors; that is, all unique factors are uncorrelated with the common factors. Additionally, the unique factors are assumed uncorrelated with each other (although this assumption may be relaxed in CFA).

Jörsekog (1969) showed how the traditional EFA model can be constrained to produce a "restricted solution" that is commonly understood as the CFA model in the structural equation modeling (SEM) literature. Specifically, in the EFA model, the elements of Λ are all freely estimated; that is, each of the m factors has an estimated relationship (i.e., factor loading) with every observed variable. Factor rotation is then used to aid interpretation. But in the CFA model, depending on *a priori* hypotheses, many of the elements of Λ are constrained to equal zero, often so that each observed variable is determined by one and only one factor. Thus, EFA and CFA are variants of the same general model and the methods we present below apply equivalently to both.

In practice, parameter estimation proceeds by fitting either a model-implied covariance matrix, $\mathbf{\Sigma}_\theta$, to the sample covariance matrix, $\mathbf{S}$, among observed variables, or by fitting a model-implied correlation matrix, $\mathbf{P}_\theta$, to the sample correlation matrix, $\mathbf{R}$. The model-implied covariance structure is

$$\mathbf{\Sigma}_\theta = \mathbf{\Lambda}\mathbf{\Phi}\mathbf{\Lambda}' + \mathbf{\Theta}, \tag{3}$$

where $\mathbf{\Sigma}_\theta$ is the predicted $p \times p$ population covariance matrix for the observed variables, $\mathbf{\Phi}$ is the $m \times m$ interfactor covariance matrix, and $\mathbf{\Theta}$ is the $p \times p$ matrix unique factor covariance matrix that often contains only diagonal elements (i.e., the unique factor variances). The correlation structure is then

$$\mathbf{P}_\theta = \mathbf{\Lambda}^*\mathbf{\Phi}^*\mathbf{\Lambda}^{*\prime} + \mathbf{\Theta}^*, \tag{4}$$

where $\mathbf{P}_\theta$ is the predicted correlation matrix and, in that a correlation matrix is simply a re-scaled covariance matrix, we can view $\mathbf{\Lambda}^*$, $\mathbf{\Phi}^*$, and $\mathbf{\Theta}^*$ as re-scaled versions of $\mathbf{\Lambda}$, $\mathbf{\Phi}$, and $\mathbf{\Theta}$, respectively. There is an historical tendency to conduct EFA using correlations and CFA using covariances, but of course it is possible to use either covariances or correlations for both EFA and CFA (see Bentler, 2007; Cudeck, 1989, on the analysis of correlations vs. covariances). Because the common factors are unobserved and thus have an arbitrary scale, it is conventional to define them as standardized (i.e., with variance equal to one), thereby establishing $\mathbf{\Phi}$ as the interfactor correlation matrix. This convention imposes necessary identification restrictions that allow the model parameters to be estimated (although alternative restrictions are possible, such as the marker variable approach often used with CFA). In addition to constraining the factor variances, EFA requires a diagonal $\mathbf{\Theta}$, with the unique factor variances along the diagonal.

The goal of model estimation is thus to find the parameter estimates that optimize the match of the predicted covariance or correlation matrix to the observed sample covariance or correlation matrix. An historically popular EFA estimation method is "principal axis factor analysis," which obtains factor loading estimates from the eigenstructure of a matrix formed by $\mathbf{R} - \hat{\mathbf{\Theta}}$. But given modern computing capabilities, we agree with MacCallum (2009) that factor models should instead be estimated using an iterative algorithm to minimize a model fitting function, such as the ordinary (or unweighted) least squares (OLS) or maximum likelihood (ML) functions. The functions in `faoutlier` utilize ML estimation for both EFA and CFA.

Case-level diagnostics for factor analysis of continuous variables

Univariate plots and bivariate scatterplots of raw data can be effective for identifying outlying cases which can have distorting effects on the results of a factor analysis. Although it is always wise to graph one's data, relying on scatterplots alone for identification of outliers is not foolproof, especially when there is a large number of variables, as often occurs in applications of factor analysis. Additionally, cases that appear to be outliers in a scatterplot might not actually be *influential* in that they produce distorted or otherwise misleading factor analysis results; conversely, certain influential cases might not be apparent in a traditional bivariate scatterplot.

Outlying status

An important property of statistics such as *Mahalanobis distance (MD)* is that they are based on the full multivariate distribution of observed variables, and as such can uncover outlying observations that may not easily appear as outliers in a univariate distribution or bivariate scatterplot. The *MD* for a given observation can be measured with

$$MD_i = (\mathbf{y}_i - \bar{\mathbf{y}})'\mathbf{S}^{-1}(\mathbf{y}_i - \bar{\mathbf{y}}) \tag{5}$$

where $\mathbf{y}_i$ is the vector of observed variable scores for case i, $\bar{\mathbf{y}}$ is the vector of observed variable means, and $\mathbf{S}$ is the sample covariance matrix. However, this *MD* measure is itself sensitive to outliers, in that any potential outlying case is included in the calculation of $\bar{\mathbf{y}}$ and $\mathbf{S}$. Thus, robust Mahalanobis distance measures are based on estimates of $\bar{\mathbf{y}}$ and $\mathbf{S}$ which are resistant to outliers. `faoutlier` includes a function `robustMD()` which uses the `MASS` package to calculate *MD* using the traditional product-moment estimates of $\bar{\mathbf{y}}$ and $\mathbf{S}$ as well as two robust *MD* measures, namely the *minimum volume ellipsoid* (MVE) and *minimum covariance determinant* (MCD) methods for obtaining robust *MDs* (see Venables & Ripley, 2002).

Commands to produce *MD* values from the `holzinger` data are:

```
data(holzinger)
MD <- robustMD(holzinger)
```

By default, `robustMD()` produces MVE-based *MD*s. MCD-based or non-robust, classical *MD*s can be obtained with the method argument, specifically `robustMD(data, method = 'mcd')` or `robustMD(data, method = 'classical')`. Because factor analysis is typically conducted with large samples, it is tedious and error-prone to inspect a simple list of numeric values of *MD*s. Thus, it is preferable to plot the results; the command `plot(MD)` will create an *index plot* of the *MD*s from the `MD` object created above (see left panel of Figure 1). This plot shows that there are no extremely aberrant *MD* values from `holzinger`, although Case 76 seems somewhat outlying. Because holzinger was generated from a multivariate normal distribution we do not expect any extreme *MD*s, but in practice the data generation process is unknown and subjective judgment is needed to determine whether a *MD* is extreme enough to warrant concern. Assessing *influence* (see below) can aid that judgment. When we repeat the above commands using the `holzinger.outlier` data set, the single extreme outlying case, Case 1 with $MD \approx 150$, is obvious (see right panel of Figure 1; note the different scale of its y-axis).

These *MD* statistics are model-free measures of case-level outlying status in that they are calculated without regard to the actual factor analysis model. As such, they provide no conclusive evidence as to whether an extreme observation is likely to have an excessive effect on the fit of a model to data or on the estimates of a model's parameters (but see Yuan & Zhong, 2008, for factor model-based MD measures). Thus, it is also important to assess model-based outliers and the *influence* of cases on model fit and parameter estimation.

Factor model outliers. Because the common factor model is a linear regression model (Equation 2), many of the well-known concepts about regression diagnostics generalize to factor analysis. Regression diagnostics are a set of methods that can be used to

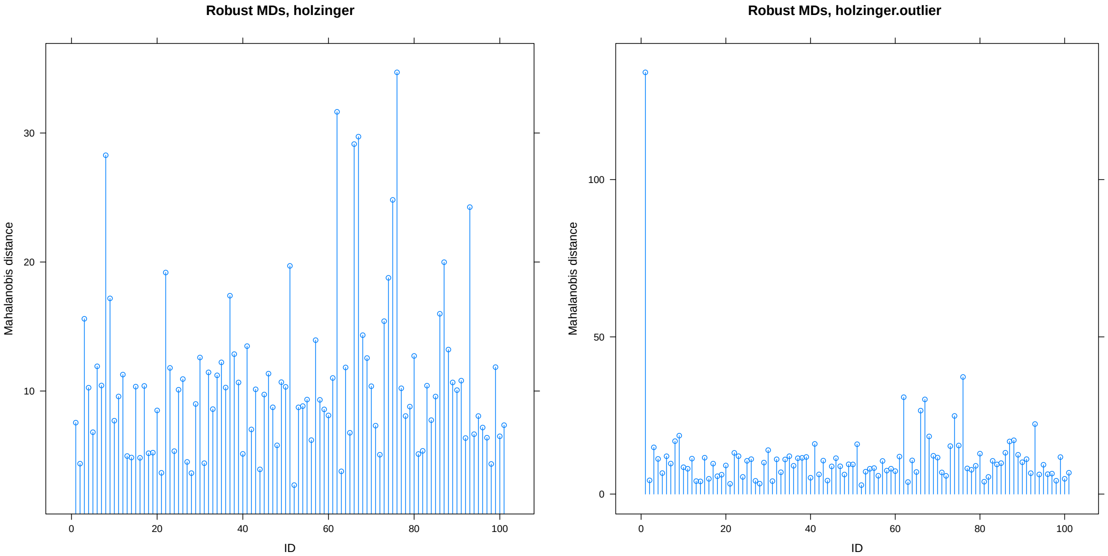

Figure 1. Robust Mahalanobis distances for `holzinger` (left panel) and `holzinger.outlier` (right panel) data sets. Note different y-axis scales for the two panels.

reveal aspects of the data that are problematic for a model that has been fitted to that data (e.g., Fox, 1991, 2008). Many data characteristics that are problematic for ordinary multiple regression are also problematic for factor analysis; the trick is that in the common factor model, the predictors (the factors) are unobserved.

In that regression (and hence factor analysis) is a procedure for modeling a dependent variable conditional on one or more predictors, a regression outlier is a case whose dependent variable value is unusual relative to its predicted, or modeled, value given its scores on the predictors (Fox, 2008). Thus, regression outliers are cases with large residuals. The factor analysis analog to a regression outlier is a case whose value for a particular observed variable is extremely different from its predicted value given its scores on the factors. In other words, cases with large (absolute) values for one or more unique factors, that is, scores on residual terms in ϵ, are factor model outliers. Equation 2 obviously defines the residuals as

$$\epsilon = \mathbf{y} - \mathbf{\Lambda}\eta. \tag{6}$$

But because the factor scores (i.e., scores on latent variables in η) are unobserved and cannot be calculated precisely, so too the residuals cannot be calculated precisely, even with known population factor loadings, $\mathbf{\Lambda}$. Thus, to obtain estimates of the residuals, $\hat{\epsilon}$, it is necessary first to estimate the factor scores, η, which themselves must be based on sample estimates of $\mathbf{\Lambda}$ and $\mathbf{\Theta}$. Bollen and Arminger (1991) show how a least-squares regression method for estimating factor scores can be applied to obtain $\hat{\epsilon}$. As implied by Equation 6, the estimated residuals $\hat{\epsilon}$ are unstandardized, in that they are in the metric of the observed variables, $\mathbf{y}$. Bollen and Arminger thus present multiple formulas to convert unstandardized residuals into standardized residuals.

`faoutlier` includes a function `obs.resid()` which calculates both unstandardized and standardized factor-model residuals. This function calculates residuals from either an

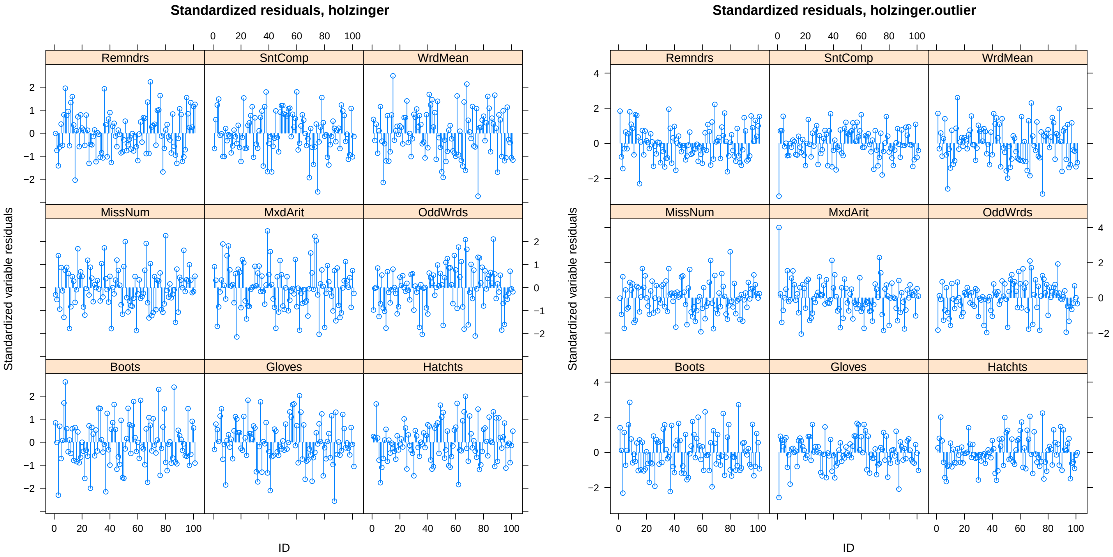

Figure 2. Standardized factor model residuals for `holzinger` (left panel) and `holzinger.outlier` (right panel) data sets from a three-factor EFA model. Note different y-axis scales for the two panels.

EFA model, in which case it is necessary to indicate the number of factors, or a CFA model, in which case it is necessary to indicate the model specification using syntax based on either the `sem` (Fox, Nie, & Byrnes, 2012), `lavaan` (Rosseel, 2012), or `OpenMx` (Boker et al., 2011) packages. The first argument of `obs.resid()` names the data, while the second argument gives either the number of factors for the EFA model or a model specification for CFA.

Now, because each case has a residual for every observed variable, it can be especially valuable to plot the residuals rather than attempting to inspect a list of individual numeric values. The commands to calculate and subsequently plot the residuals of the `holzinger` data from a three-factor EFA model are:

```
nfact <- 3 #set the number of factors for EFA model
resids1 <- obs.resid(holzinger, nfact)
plot(resids1, restype = 'std_res')
```

Figure 2 illustrates the standardized residuals; unstandardized residuals can be plotted using the argument `restype = ’res’` in the plot command above. Because the data were drawn from a standard normal distribution conforming to a known population model, these residuals themselves are approximately normally distributed with no extreme outliers. When we repeat the commands above using the `holzinger.outlier` data, Case 1 is a clear factor model outlier for variables 2 and 5 in particular (see right panel of Figure 2; again note the different scale). We can also obtain a general multivariate statistic computed by summing the squared standardized residual values over each variable with the option `restype = ’obs’` (this is the default option). This statistic may be useful since it has more power for detecting multivariate outliers than the univariate counterparts.

Suppose that instead a three-factor CFA model is to be fitted to `holzinger` such that the first three observed variables have freely estimated loadings on the first factor, the next three variables load on the second factor, and the last three variables load on the third factor, with all other potential factor loadings constrained to zero (i.e., each observed variable loads on only one factor) while the factors are freely intercorrelated. Using syntax of the `sem` package and the `holzinger` variable names, we can assign this model specification to an object called `CFAmodel` as below (see Fox et al., 2012):

```
CFAmodel <- specifyModel()
   F1 -> Remndrs, lam11
   F1 -> SntComp, lam21
   F1 -> WrdMean, lam31
   F2 -> MissNum, lam41
   F2 -> MxdArit, lam52
   F2 -> OddWrds, lam62
   F3 -> Boots, lam73
   F3 -> Gloves, lam83
   F3 -> Hatchts, lam93
   F1 <-> F1, NA, 1
   F2 <-> F2, NA, 1
   F3 <-> F3, NA, 1
```

We then calculate and plot standardized residuals from this three-factor CFA model by again using the `obs.resid()` function:

```
resids1 <- obs.resid(holzinger, CFAmodel)
plot(resids1, restype = 'std_res')
```

An additional benefit when fitting a CFA model is that the ML estimation method used is full-information, meaning that cases with missing data can be included in the analysis and even evaluated as to whether they are influential outliers given their valid responses. By default in `faoutlier`, all functions have an `na.rm = TRUE` option which can be over-written to `FALSE` when the incomplete cases are of interest.

Although the `holzinger` data do not contain any notable outliers from this three-factor CFA model, it is important to recognize that the presence of outliers or influential observations can be indicative of model misspecification (Pek & MacCallum, 2011). Here, although a three-factor EFA model fits the `holzinger` data very well, this three-factor CFA model does not actually have a good fit (RMSEA = .175; CFI = .85) because constraining the secondary loadings from the EFA model to zero for the CFA model is overly restrictive (see van Prooijen & van der Kloot, 2001).

Influence

Just as in ordinary regression analysis, in the context of factor analysis an outlying case (or set of cases) may or may not have considerable *influence* on modeling results (see Flora et al., 2012; Pek & MacCallum, 2011; Yuan & Zhong, 2008). Case influence refers to the extent that an individual case (or a set of cases) impacts modeling results and can exert itself with respect to overall model fit or estimates of individual parameters. In regression analysis, it is well-known that cases with extreme values of predictors but small residuals (so-called "good leverage points") have little impact on the estimated values of regression coefficients but also lead to an increased precision of estimation (in the sense of reduced standard error); conversely, a case with a large residual but small leverage will have little influence on the parameter estimates (although it decreases precision). Although leverage is not directly observable in factor analysis because the predictors are unobserved these phenomena occur analogously (see Yuan & Zhong, 2008). Thus, perhaps more than considering outlying status of cases, it is important to assess case influence on factor analysis results. Influence is most commonly measured with deletion statistics, which measure the change in a particular statistic that occurs when a particular case is deleted from the sample. Another approach to determining influence is through a forward search algorithm. Below, we describe the deletion statistics and forward search algorithm implemented in `faoutlier`.

Likelihood distance. When a model is estimated with maximum likelihood, the influence of an individual case on overall model fit can be measured with *likelihood distance* (*LD*; see Pek & MacCallum, 2011, for details). This measure is defined as

$$LD_i = 2\left[L(\hat{\theta}) - L(\hat{\theta}_{(i)})\right]$$

where $L(\hat{\theta})$ is the log-likelihood of the data given the vector of parameter estimates $\hat{\theta}$ obtained with the original, complete sample and $L(\hat{\theta}_{(i)})$ is the log-likelihood of the data given the vector of parameter estimates $\hat{\theta}_{(i)}$ obtained with case i deleted from the sample. A positive *LD* value for a given case i indicates that its removal leads to worse model fit, whereas a negative value indicates that removal of case i improves fit. Note that this definition is closely related to the goodness of fit (GOF) difference in χ^2 values (Pek & MacCallum, 2011), where $\Delta\chi^2_i = \chi^2(\hat{\theta}_{(i)}) - \chi^2(\hat{\theta})$. Again, we can see that if a model estimated without case i fits the data better than with that case included, then a negative $\Delta\chi^2_i$ value will occur.

The GOF distances for the `holzinger` data with a three-factor EFA model may be calculated and plotted using the `GOF()` function in `faoutlier` as follows:

```r
GOFresult <- GOF(holzinger, nfact)
plot(GOFresult)
```

As before, the first argument of the function names the data to be modeled, while the second argument names either the number of factors for an EFA model or the name of a model specification object for a CFA model (e.g., the `CFAmodel` object defined above). Figure 3 shows index plots of the likelihood distances from the three-factor EFA model for both `holzinger` and the `holzinger.outlier` data. It is clear that while none of the observations in `holzinger` has an excessive influence on model fit, the first case in `holzinger.outlier` has a very strong deleterious influence on fit.

Generalized Cook's distance. Whereas *LD* measures the influence of a case on model fit, case influence on a set of parameter estimates from a factor analysis model can be measured with *generalized Cook's Distance* (*gCD*; Pek & MacCallum, 2011). This measure is defined as

$$gCD_i = (\hat{\theta} - \hat{\theta}_{(i)})'(VAR(\hat{\theta}_{(i)}))^{-1}(\hat{\theta} - \hat{\theta}_{(i)}) \tag{7}$$

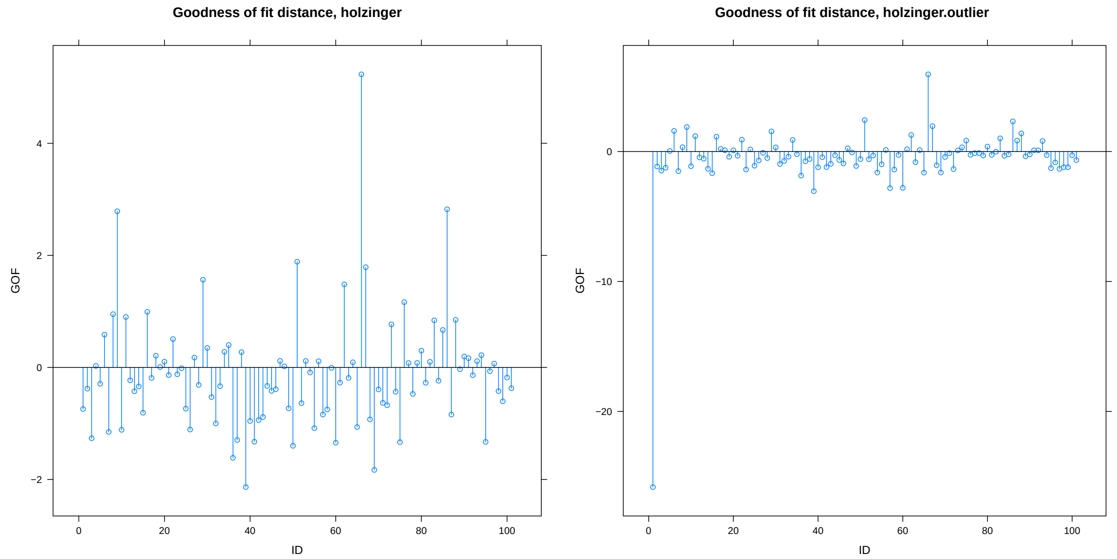

Figure 3. GOF distances for `holzinger` (left panel) and `holzinger.outlier` (right panel) data sets from a three-factor EFA model.

where $\hat{\theta}$ and $\hat{\theta}_{(i)}$ are as above and $VAR(\hat{\theta}_{(i)})$ is the estimated asymptotic variance-covariance matrix of the parameter estimates obtained with case i deleted. Like *MD*, *gCD* is in a squared metric with values close to zero indicating little case influence on parameter estimates and those far from zero indicating strong influence on parameter estimates.

The `gCD()` function in `faoutlier` calculates *gCD* case influence values for either an EFA model or a CFA model. As with the previous functions, the data set is the function's first argument while the model specification (either EFA or CFA) is the second object, as shown below for calculating *gCD* values for the `holzinger` data with a three-factor EFA model and subsequently plotting them:

```
gCDresult <- gCD(holzinger, nfact)
plot(gCDresult)
```

When these commands are repeated with the `holzinger.outlier` data, an extreme case does appear (see right panel of Figure 4). However, with this particular EFA example, the extreme *gCD* occurs for the legitimate Case 37 rather than Case 1, which was the artificially perturbed case. This result occurs because of the rotational indeterminacy property of EFA and how the model is optimized. In EFA, a factor's coefficients are unique only up to a linear transformation, and because of that models are estimated by focusing only on the optimization of the *communalities* (one minus the diagonal of $\mathbf{\Theta}^*$ in Equation 6) rather than the $\mathbf{\Lambda}$, $\mathbf{\Phi}$, and $\mathbf{\Theta}$ matrices directly. Once optimal communalities are obtained, a subsequent extraction method is used to obtain an arbitrary orientation for the initial factor loadings (Gorsuch, 1983). An unfortunate consequence of this feature is that individual cases can affect an entire solution in EFA since communality values are estimated based on the full correlation (or covariance) matrix, whereas in CFA a case's overall influence can be isolated and only affect certain elements in the model implied covariance matrix (i.e., $\mathbf{\Sigma}_\theta$).

Thus, although the set of *gCD*s reveals that there is an extremely influential case, it may not identify the correct case when an EFA model is fitted to the data. When we instead fit the three-factor CFA model specified above to `holzinger.outlier`, the *gCD*s calculated with the `gCD()` function (i.e., `gCD(holzinger.outlier, CFAmodel)`) do correctly identify Case 1 as having extreme influence (see the bottom of Figure 4).

Forward search. Because deletion statistics *LD* and *gCD* are calculated by deleting only a single case i from the complete data set, they are susceptible to *masking errors*, which occur when an influential case is not identified as such because it is located in multivariate space close to one or more similarly influential cases. Alternatively, a forward search algorithm can be used to identify groups of influential cases. `faoutlier` implements the forward search procedure outlined by Mavridis and Moustaki (2008). Below we describe this method briefly; see Mavridis and Moustaki for complete details and additional examples.

First, an optimal working subset of the data of initial size n_g is selected from the full sample of size n and the factor analysis model is fitted to these cases. This initial working set is optimal in the sense that the chosen cases combine to maximize the log-likelihood of the data given the model (although other criteria are possible). Next, the search iterates forward in that cases are added, one at a time, to the working set according to one or more criteria, such as contribution to the likelihood function, minimum Mahalanobis distance, or minimum model-implied residuals from the working set. At iteration I, the cases are ordered according to their closeness to working set, and the $(n_g + I)$ cases closest to the working set are selected to form a new working set. Note that a given observation in a previous working set (including the initial set) is not necessarily included in subsequent working sets; observations may enter and leave the working set at each step of the process. However, at the end of the process, all cases have been included in the working set. At this point, a “forward plot” of a given statistic by iteration number depicts how distant the observations added toward the end of the search are from the well-fitting observations included at the beginning of the search. For example, if the likelihood ratio goodness of fit statistic for the model is calculated for each iteration (i.e., for each working set), a case or group of cases contributing strongly to model misfit will appear in this plot as outliers above the final iteration or iterations.

Forward search is implemented in `faoutlier` with the `forward.search()` function. In the same manner as the other functions described above, it is necessary to supply as arguments the data to be analyzed and an EFA or CFA model specification. For example, to conduct a forward search with the `holzinger` data for three-factor EFA model and generate the subsequent forward plot, the syntax is

```r
FS <- forward.search(holzinger, nfact)
plot(FS, stat = 'RMR')
```

By default, the initial size of the working set is 40% of the full sample, and `forward.search()` will fit the model to 1000 subsamples of size $.4n$ to find an optimal initial working set. These defaults may be overridden with the `p.base` and `n.subsets` arguments, respectively. For example, to specify a 50% working set with 2000 subsets considered for the initial working set, the function call would be `forward.search(data, model, n.subsets = 2000, p.base = .5)`. Additionally, across iterations `forward.search()`

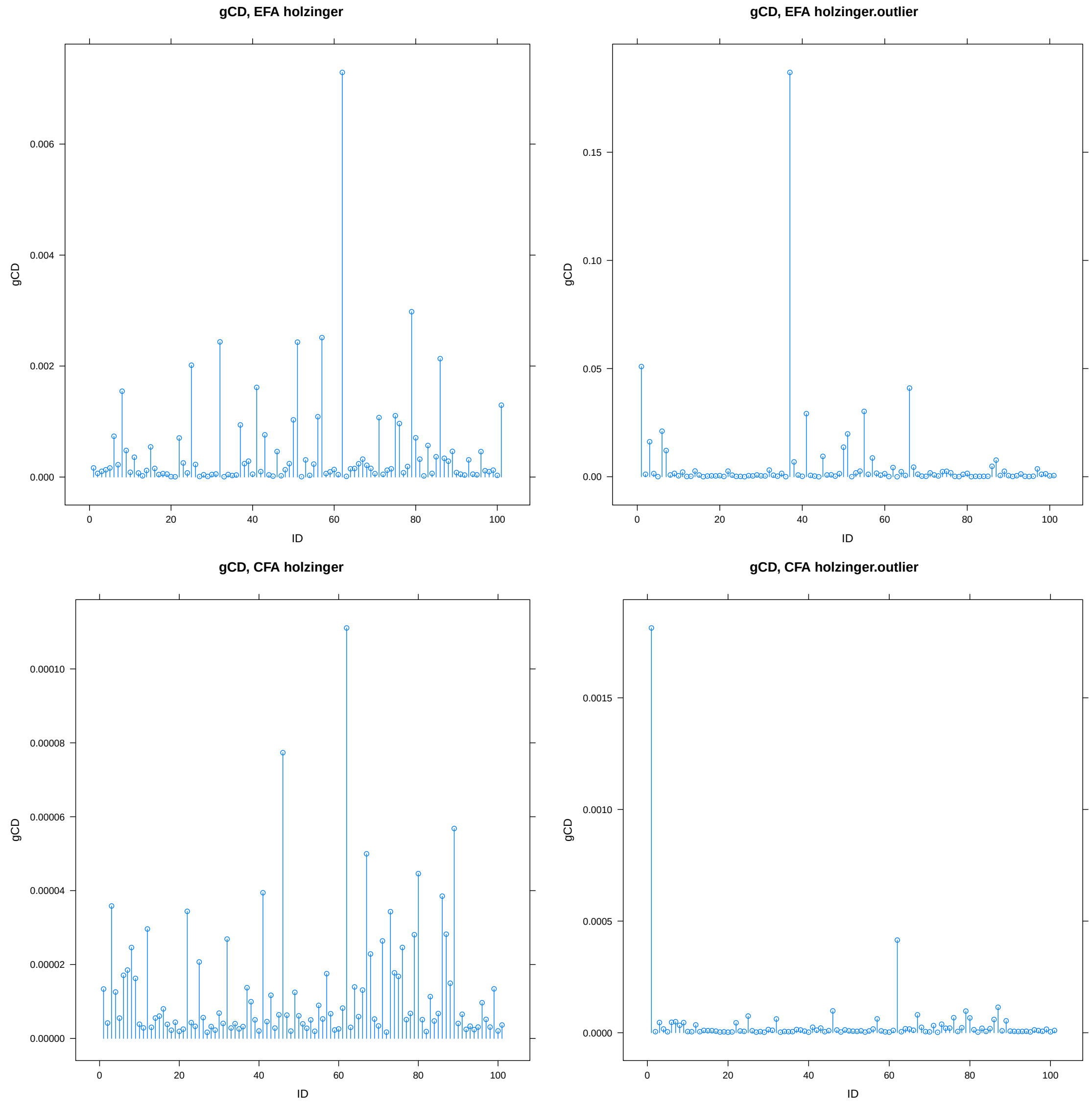

Figure 4. *gCD* for EFA (top) and CFA (bottom), for `holzinger` data with (right) and without (left) an outlier.

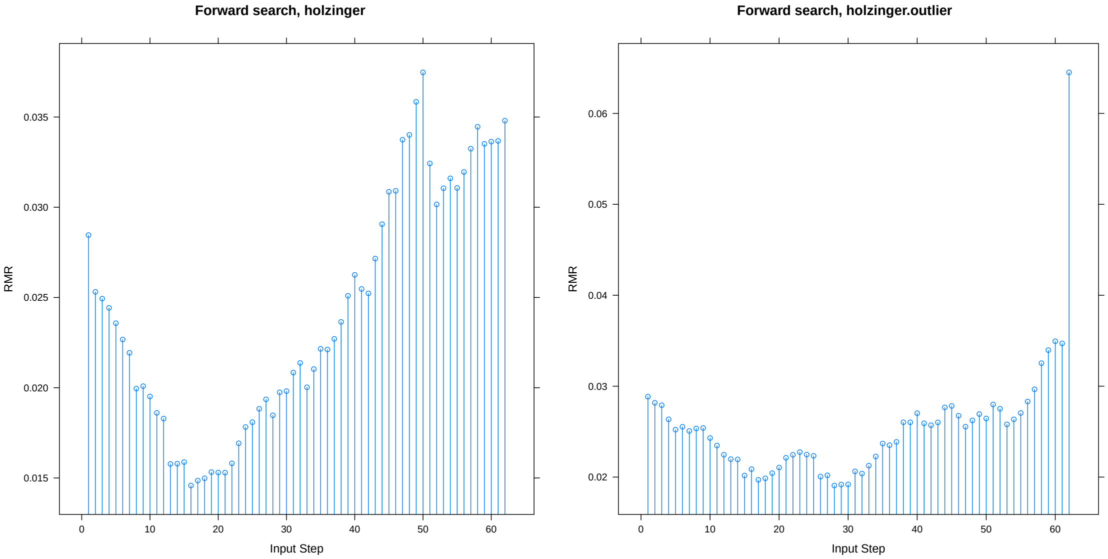

Figure 5. Forward search method for `holzinger` (left panel) and `holzinger.outlier` (right panel) data sets from a three-factor EFA model.

adds cases according to their contribution to the likelihood function and the minimum robust *MD* from the working set. This default can be overridden with the `criteria` argument; for example, to iterate only according to minimized residuals, the function call would be `forward.search(data, model, criteria = 'res')`.

Figure 5 presents forward plots with the root mean squared residual (RMR) statistic for the `holzinger` and `holzinger.outlier` data with the three-factor EFA model. With the `holzinger` data, one can see that the RMR statistic gradually increases as cases are added across iterations, but with the addition of the last few iterations the model fit does not change dramatically. But with the `holzinger.outlier` data, one can see that the final iteration is associated with a substantial, outlying increase in the RMR statistic; this increase is caused by the addition of the non-random outlying data point to the working set.

Discussion

The `R` package `faoutlier` provides a suite of functions for detecting multivariate outliers in a data set to be factor analyzed (i.e., robust *MD*), and more importantly, for identifying factor model outliers (i.e., Bollen & Arminger, 1991, residuals) and influential observations using either case-deletion statistics (i.e., *LD* for model fit and *gCD* for parameter estimates) or a forward-search method. As with any parametric procedure, the presence of outliers and influential cases can have serious consequences for factor analytic results. As described above, because factor analysis is the regression of observed variables on factors, principles and procedures from multiple regression case diagnostics can be applied. Unfortunately, extant factor analysis software does not include such procedures. To remedy this shortcoming, use of `faoutlier` in concert with other `R` packages for factor analysis, such as `psych` (Revelle, 2012) for EFA and `sem` for CFA, facilitates a thorough analysis that includes the use of diagnostics for the discovery of outlying or influential cases.

Upon finding outlying or otherwise influential cases, it is important to attempt to identify their source. If such cases are due to researcher or participant error, then they should be corrected or removed. Otherwise, contrary to common practice, methodologists generally recommend that outliers and influential cases not be deleted from the data (e.g., Bollen & Arminger, 1991; Pek & MacCallum, 2011; Yuan & Zhong, 2008). Instead, robust procedures that minimize the excessive influence of extreme cases are recommended; one such approach is to factor analyze a MCD estimated covariance matrix (Pison, Rousseeuw, Filzmoser, & Croux, 2003), which can be calculated with the `R` package `MASS`. Additionally, it is important to recognize that when a model is poorly specified (e.g., the wrong number of factors has been extracted), it is likely that many cases in a sample would be flagged as influential, but when there are only a few bad cases, the model may be consistent with the major regularities in the data except for these cases (Pek & MacCallum, 2011).

In closing, we wish to point out that many of the functions in `faoutlier` are not limited to factor analysis, and will work with any structural equation model object specified using syntax of the `sem, lavaan,` or `OpenMx` package. This generality reflects the theme that disproportionally influential observations can arise in any modeling procedure, and thus it is prudent to utilize diagnostic procedures as a matter of routine.

References

Bentler, P. M. (2007). Can scientifically useful hypotheses be tested with correlations? *American Psychologist*, *62*, 772–782.

Boker, S., Neale, M., Maes, H., Wilde, M., Spiegel, M., Brick, T., ... Fox, J. (2011). OpenMx: An open source extended structural equation modeling framework. *Psychometrika*, *76*(2), 306–317.

Bollen, K., & Arminger, G. (1991). Observational residuals in factor analysis and structural equation models. *Sociological Methodology*, *21*, 235–262.

Browne, M. W., Cudeck, R., Tateneni, K., & Mels, G. (2008). CEFA: Comprehensive exploratory factor analysis, version 3.03 [Computer software manual]. Retrieved from `http://faculty.psy.ohio-state.edu/browne/`

Chalmers, P. (2011). faoutlier: Influential case detection methods for factor analysis and sem [Computer software manual]. Retrieved from `https://github.com/philchalmers/faoutlier` (R package version 0.2.3)

Cudeck, R. (1989). Analysis of correlation matrices using covariance structure models. *Psychological Bulletin*, *105*, 317–327.

Dalgaard, P. (2008). *Introductory statistics with R* (2nd ed.). New York: Springer.

Flora, D. B., LaBrish, C., & Chalmers, R. P. (2012). Old and new ideas for data screening and assumption testing for exploratory and confirmatory factor analysis. *Frontiers in Psychology*, *3*, 1–21. Retrieved from `http://www.frontiersin.org/Journal/Abstract.aspx?s=956&name=quantitative_psychology_and_measurement&ART_DOI=10.3389/fpsyg.2012.00055` doi: 10.3389/fpsyg.2012.00055

Fox, J. (1991). *Regression diagnostics: An introduction.* Thousand Oaks, CA: Sage Publications.

Fox, J. (2008). *Applied regression analysis and generalized linear models.* Thousand Oaks, CA: SAGE Publications.

Fox, J., Nie, Z., & Byrnes, J. (2012). sem: Structural equation models [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=sem` (R package version 3.0-0)

Gorsuch, R. L. (1983). *Factor analysis* (2nd ed.). Hillsdale, NJ: Erlbaum.

Harman, H. H. (1960). *Modern factor analysis.* Chicago, IL: University of Chicago Press.

Jörsekog, K. G. (1969). A general approach to confirmatory maximum likelihood factor analysis. *Psychometrika*, *34*(2), 183–202.

Lawley, D. N., & Maxwell, A. E. (1963). *Factor analysis as a statistical method.* London: Butterworth.

MacCallum, R. C. (2009). Factor analysis. In R. Millsap & A. Maydeu-Olivares (Eds.), *The sage handbook of quantitative methods in psychology* (pp. 123–147). Thousand Oaks, CA: SAGE Publications.

Mavridis, D., & Moustaki, I. (2008). Detecting outliers in factor analysis using the forward search algorithm. *Multivariate Behavioral Research*, *43*, 453–475. doi: 10.1080/00273170802285909

Pek, J., & MacCallum, R. C. (2011). Sensitivity analysis in structural equation models: Cases and their influence. *Multivariate Behavioral Research*, *46*(2), 202–228.

Pison, G., Rousseeuw, P. J., Filzmoser, P., & Croux, C. (2003). Robust factor analysis. *Journal of Multivariate Analysis*, *84*, 145–172.

R Core Team. (2014). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `http://www.R-project.org/`

Revelle, W. (2007). Using R for personality research [Computer software manual]. Retrieved from `http://personality-project.org/r/r.short.html`

Revelle, W. (2012). psych: Procedures for psychological, psychometric, and personality research [Computer software manual]. Evanston, Illinois. Retrieved from `http://personality-project.org/r/psych.manual.pdf` (R package version 1.2.1)

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, *48*(2), 1–36. Retrieved from `http://www.jstatsoft.org/v48/i02`

Thurstone, L. L. (1947). *Multiple factor analysis.* Chicago: University of Chicago Press.

van Prooijen, J.-W., & van der Kloot, W. A. (2001). Confirmatory analysis of exploratively obtained factor structures. *Educational and Psychological Measurement*, *61*, 777–792.

Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with S.* Springer.

Yuan, K.-H., & Zhong, X. (2008). Outliers, leverage observations, and influential cases in factor analysis: Using robust procedures to minimize their effect. *Sociological Methodology*, *38*, 329–368.