Exploratory graph analysis (EGA) is a highly accurate technique that was recently proposed within the framework of network psychometrics to estimate the number of factors underlying multivariate data. Unlike other methods, EGA produces a visual guide—network plot—that not only indicates the number of dimensions to retain, but also which items cluster together and their level of association. However, although previous studies have found EGA to be superior to traditional methods, they are limited in the conditions considered. These issues are here addressed through an extensive simulation study that incorporates a wide range of plausible structures that may be found in practice. Additionally, a new variant of EGA based on the triangulated maximally filtered graph approach (EGAtmfg) is evaluated, and both are compared with five widely used and/or recommended factor analytic techniques. Overall, EGA and EGAtmfg are found to perform as well as the most accurate traditional method, parallel analysis, and to produce the best large-sample properties of all the methods evaluated. To increase use and transparency, we present a straightforward R tutorial on how to use and interpret EGA, and apply it to the scores from a well-known Big Five personality test. Finally, we offer a set of practical guidelines for applied researchers, and outline next steps for large scale assessments in health beyond psychology.

EGA | Network models | dimensionality | number of factors

Network models have been widely applied in psychology since the publication of the mutualism model of intelligence (Van Der Maas et al., 2006). This model explains that the positive manifold of intelligence tests is not a product of a single general factor (the g factor), but rather a consequence of a network of reciprocal causal relations between cognitive abilities. In the same line, Borsboom (2008) proposed a new view of psychopathology as a system of causal relations between symptoms. In this new paradigm, symptoms of mental disorders are parts of a larger causal network. The network perspective was also proposed as a new way to interpret comorbidity (Cramer, Waldorp, Van Der Maas, & Borsboom, 2010), reliably reproducing empirical population statistics of two mental disorders using simulations of network structures (Borsboom, Cramer, Schmittmann, Epskamp, & Waldorp, 2011). Network models have been used not only in the field of psychopathology (Fried et al., 2017), but also in clinical psychology (Bork, Borkulo, Waldorp, Cramer, & Borsboom, 2018), cognitive psychology and individual differences (Golino & Demetriou, 2017; Van Der Maas, Kan, Marsman, & Stevenson, 2017), social psychology (Dalege, Borsboom, Harreveld, Waldorp, & Maas, 2017), and many other areas (Epskamp, Rhemtulla, & Borsboom, 2017).

The network perspective of psychological constructs originated a new subfield of quantitative psychology, termed network psychometrics (Epskamp, Maris, Waldorp, & Borsboom, in press; Epskamp et al., 2017). In the network psychometrics perspective, the Gaussian graphical model (GGM: Lauritzen, 1996) is used to estimate the joint distribution of pairs of random variables (i.e. nodes in the network) by modeling the inverse of the variance-covariance matrix (Epskamp et al., 2017). Nodes (e.g. test items) are connected by edges or links, which indicate the strength of the association between the variables (S. Epskamp & Fried, 2018). Edges are typically partial correlation coefficients that indicate conditional independence between two variables after controlling the effect of all other variables.

Golino and Epskamp (2017) showed that the GGM model combined with a clustering algorithm for weighted networks (walktrap; Pons & Latapy, 2006) could accurately recover the number of factors in simulated datasets, presenting a higher accuracy than traditional factor analytic based methods, especially when the correlation between factors was high (.70). Golino and Epskamp (2017) termed this new method exploratory graph analysis (EGA). Golino and Demetriou (2017) further investigated the accuracy of EGA to estimate the number of factors using a dataset simulated by Keith, Caemmerer, and Reynolds (2016). The authors showed that EGA correctly estimated the number of factors in 100% of the cases, while the next best method produced a substantially lower accuracy level of 74%. Golino and Demetriou (2017) concluded that facing the limitations of techniques traditionally used in psychology, an optimum solution for estimating the number of factors was to use EGA to explore the basic dimensionality of a given instrument and confirmatory factor analysis (CFA) to verify the fit of the suggested structure.

In spite of the evidence showing that EGA is very accurate at estimating the number of factors, more studies are required to understand how this new technique performs under different conditions (Golino & Demetriou, 2017), and how it compares to different factor analytical methods. Among the limitations of the studies conducted by Golino and Epskamp (2017) and Golino and Demetriou (2017), four shortcomings are largely relevant for judging the suitability of EGA as a
robust dimensionality assessment method. First, the authors did not investigate the impact of varying levels of factor loadings in the accuracy of EGA to estimate the number of factors, although it’s widely known that factor loadings are one of the most important variables that affect the accuracy of the methods (Garrido, Abad, & Ponsoda, 2013). Second, neither EGA studies investigated the accuracy of EGA in the presence of many factors (more than four). This is especially relevant to studies seeking to identify the number of factors in large-scale assessments involving many measures, tests and/or questionnaires. Assessing the number of factors in the presence of multiple instruments is also central for secondary analysis of existing data sets. Third, Golino and Epskamp (2017) did not control for the level of difficulty of the items, so the item categories had an equal distribution. Finally, while Golino and Epskamp (2017) and Golino and Demetrio (2017) compared EGA with a number of traditional factor analytic methods, they did not include different types of parallel analysis or any technique based on the scree test (Cattell, 1966), which is widely applied in psychology.

Facing the limitations pointed above, the goal of the current paper is twofold. The first goal is to investigate via a simulation study the accuracy of EGA to estimate the correct number of factors under a broader and more realistic set of conditions than what has been previously considered. Additionally, we present an alternative version of EGA that replaces the GGM with the triangulated maximally filtered graph approach (TMFG; Massara, Di Matteo, & Aste, 2016), hereafter EGATmg, potentially overcoming some of the limitations of the former method. Moreover, the performance of the EGA methods will be compared with two automated versions of the scree test, the acceleration factor (AF) and the optimal coordinate (OC) methods (Raiche, Walls, Magis, Riopel, & Blais, 2013), the Kaiser-Guttman eigenvalue-greater-than-one rule (K1; Kaiser, 1960), and two versions of parallel analysis (PA), namely PA with principal component analysis extraction (PApca) and PA with principal axis factoring (PApaf). The second goal is to provide a tutorial on how to apply EGA using R, so researchers from different fields interested in estimating the dimensionality of their tests’ scores, questionnaires, or other types of instruments can easily apply it. The tutorial uses data from the Virginia Cognitive Aging Project (Salthouse, 2018) and verifies the dimensionality of a widely used personality scale: Goldberg’s 50-item Big Five personality scale (Goldberg, 1992).

This paper is organized as follows. In the first section, EGA is briefly presented, and previous research investigating its accuracy to estimate the number of factors is described in detail. In the same section, the GGM approach is described and an alternative version of EGA using the TMFG method is outlined. In the second section, a brief overview of the traditional factor-analytic techniques used in the current paper is presented. The third and fourth sections describe the current simulation study and its results. The fifth section provides a tutorial on how to use EGA via R. Finally, a discussion focusing on how EGA can guide researchers in their search for the number of factors in psychology and other applications is offered at the end of the paper.

### Exploratory Graph Analysis

Golino and Epskamp (2017) proposed EGA as a new method to estimate the number of latent variables underlying multivariate data using undirected network models (Lauritzen, 1996). EGA is part of a new field termed network psychometrics (Epskamp et al., in press, 2017), which uses probabilistic network models as a representation of the joint distribution of observed variables, modeling the variance that is unique between pairs of variables (Epskamp et al., 2017). By focusing on the unique variance, rather than the variance that is shared across all variables (as in structural equation modeling), the field of network psychometrics provides a set of methods that complement latent variable modeling (Epskamp et al., 2017).

The EGA technique estimates the number of factors by combining the GGM model (Lauritzen, 1996) with the walktrap clustering algorithm (Pons & Latapy, 2006), a common approach for estimating clusters in weighted networks. The algorithm iteratively identifies how each node is connected to neighboring nodes, using them to determine which cluster each node belongs to. First, a network must be estimated. The current approach is to estimate a GGM, as follows. Let’s assume a set of random variables $\mathbf{y}$ that are normally distributed with mean zero and variance-covariance matrix $\Sigma$, and let $\mathbf{K}$ (kappa) be the inverse of $\Sigma$, also known as the precision matrix,

$$K = \Sigma^{-1} \quad (1)$$

then, each element $k_{ij}$ (row $i$, column $j$ of $\mathbf{K}$) can be standardized to yield the partial correlation between variables $y_i$ and $y_j$, given all other variables in $\mathbf{y}$, $y_{-(i,j)}$ (Epskamp, Waldorp, Mõttus, & Borsboom, 2016):

$$\text{Cor}(y_i, y_j | y_{-(i,j)}) = - \frac{k_{ij}}{\sqrt{k_{ii}}\sqrt{k_{jj}}} \quad (2)$$

Epskamp et al. (2016) points that modeling $\mathbf{K}$ in a way that every nonzero element is treated as a freely estimated parameter generates a sparse model for $\mathbf{\Sigma}$. The sparse model of the variance-covariance matrix is the GGM (Epskamp et al., 2016).

The level of sparsity of the GGM can be set using different methods. The most common approach in network psychometrics is to apply a variant of the least absolute shrinkage and selection operator (LASSO; Tibshirani, 1996) termed graphical LASSO (GLASSO; Friedman, Hastie, & Tibshirani, 2008). The GLASSO is a regularization technique that is very fast to estimate both the model’s structure and the parameters of a sparse GGM (Epskamp et al., 2016). It has a tuning parameter ($\lambda$) that can be chosen in a way to minimize the extended Bayesian information criterion (EBIC; Chen & Chen, 2008), which has been shown to accurately retrieve the true network structure in simulation studies (Epskamp, 2016; Foygel & Drton, 2010).

Golino and Epskamp (2017) studied the accuracy in estimating the number of dimensions of EGA along with six classical techniques: very simple structure (VSS; Revelle & Rocklin, 1979), minimum average partial (MAP; Velicer, 1976), Bayesian information criterion (BIC), EBIC, K1, and PA with generalized weighted least squares extraction and random data generation from a multivariate normal distribution. The authors simulated 32,000 data sets to fit known factor structures, systematically manipulating four variables: number of factors (2 and 4), number of items (5 and 10), sample size (100, 500, 1000 and 5000), and correlation between factors (0, .20, .50 and .70). The results of Golino and Epskamp (2017) showed that the accuracies of the different techniques, in
ascending order, were: 39% for VSS, 50% for MAP, 81% for K1, 81% for BIC, 82% for EBIC, 89% for PA, and 93% for EGA. Also, EGA was especially superior to the traditional techniques in the cases of larger structures (4 factors) and very high factor correlations (.70), achieving an accuracy of 71% that was much higher than the 40% of next best method (PA). Furthermore, through a series of analyses of variance (ANOVAs), Golino and Epstein (2017) also ascertained that EGA was the most robust method, as its accuracy was less affected by the manipulated variables than those of the other methods.

In a second study, Golino and Demetriou (2017) re-analyzed the data simulated by Keith et al. (2016) using EGA in the conditions that presented the lowest accuracy across the techniques examined: 500 observations, high correlation between factors (.70), moderate factor loadings (.50), and different number of indicators per factor (2, 3, 4, 6, 8 or 10). Keith et al. (2016) found the following accuracies (in ascending order) for the techniques they evaluated: 0% for MAP, 10% for the chi-square test of exact fit for exploratory factor analysis using maximum likelihood, 11% for K1, 21% for PAPca, 49% for PAPaf, 68% for the chi-square test of differences between nested CFA models, and 74% for the Akaike information criterion also between nested CFA models. Golino and Demetriou (2017) showed that when EGA was applied to the same datasets it correctly estimated the number of factors in 100% of the cases.

The higher accuracy of EGA, when compared to traditional factor analytic methods, might be explained by the network psychometrics’ approach to focus on the unique variance between pairs of variables, instead of the variance shared across all variables. When a dataset is simulated following a traditional factor model, the dimensionality structure becomes clearer when a network of regularized partial correlations is estimated. Figure 1 shows a simulated five-factor model (population correlations) with loadings of .70, two-factor correlations of .70, and eight items per factor. In this figure, the population correlation matrix is plotted as a network with a two-dimensional layout computed using the Fruchterman-Reingold algorithm. In this layout, nodes with stronger edges (e.g. high correlations) are placed closer than nodes with weak edges (e.g. low correlations). The two-dimensional layout helps to visually inspect groupings of variables, since variables with higher correlations are plotted together. The colors of the nodes represent the factors. In the left side of the figure, the population correlation matrix is plotted, while in the right side the estimated EGA structure is shown. Estimating a network using regularized partial correlations results in a clear structure with five groups of variables. Also, the strength of the regularized partial correlations is stronger within clusters than between clusters, making the true simulated five-factor structure easier to depict, even if the true correlation between factors is high (in this example the correlation is .70). This is a result of the network approach’s focus on the unique variance between pairs of variables. When the GGM is estimated, pairs of variables that are conditionally dependent given the other variables can be more easily detected, especially if the resulting network is plotted using an algorithm that takes into account the strength of the edge weights.

More recently, a new approach to estimate psychometric networks, the TMFG, entered the field (Christensen, Kenett, Aste, Silvia, & Kwapi, 2018). The TMFG method applies a structural constraint on the network, which restrains the network to retain a certain number of edges (i.e., \(3n - 6\); Massara et al. 2016), where \(n\) is the number of nodes. The network is also composed of 3- and 4-node cliques (i.e., sets of connected nodes; a triangle and tetrahedron, respectively). The TMFG method constructs a network using zero-order correlations and the resulting network can be associated with the inverse covariance matrix (yielding a GGM; Barfuss, Massara, Di Matteo, & Aste, 2016). Construction begins by forming a tetrahedron of the four nodes that have the highest sum of correlations to all other nodes. Next, the algorithm iteratively identifies the node that maximizes its sum of correlations to three of the nodes already included in the network and then adds that node to the network. In this process, the network automatically generates what’s called a planar network. A planar network is a network that could be drawn on a sphere with no edges crossing (often, however, the networks are depicted with edges crossing; Tumminello, Aste, Di Matteo, & Mantegna, 2005).

An intriguing property of planar networks is that they form a “nested hierarchy” within the overall network (Song, Di Matteo, & Aste, 2011). This simply means that sub-networks are nested within other, larger sub-networks of the overall network. The constituent elements of these sub-networks are the 3-node cliques (i.e., triangles), which form an emergent hierarchy in the overall network (Song, Di Matteo, & Aste, 2012). Research that compared a novel algorithm, which exploited this hierarchical structure, to several traditional methods of hierarchical clustering (e.g., complete linkage and k-medoids) found that the novel algorithm outperformed the traditional methods, retrieving more information with fewer clusters (Musmeci, Aste, & Di Matteo, 2015; Song et al., 2012). In psychology, the TMFG variant of EGA was applied in one study that examined the factor structure of a network composed of four Openness to Experience personality inventories (Christensen, Cotter, & Silvia, 2018). The authors found 10 different facets that were consistent with previous work and evidence for a novel third dimension that had yet to be uncovered using more traditional methods. A main limitation of this study, however, was that they did not compare EGAtmfg’s findings to any other dimension reduction methods.

**Factor Analytic Techniques**

**Eigenvalue-based methods.** The eigenvalue-greater-than-one rule, also known as Kaiser’s rule or K1, indicates that only factors with eigenvalues above one should be retained. The rationale of this rule is that a factor should explain at
At least as much variance as a variable is bestowed in the standard score space and that components with eigenvalues above one are ensured to have positive internal consistencies (Garrido et al., 2013; Kaiser, 1960). However, the proofs for this rule were developed for population statistics, and a large body of research has shown that it doesn’t perform well with finite samples (Hayon, Allen, & Scarpello, 2004). Nevertheless, recent studies have shown that this rule is still applied in practice frequently (Izquierdo, Olea, & Abad, 2014).

Parallel analysis was originally proposed by Horn (1965) as a modification of the K1 rule that took into account the sampling variability of the latent roots. The rationale behind this method is that the true dimensions should have sample eigenvalues that are larger than those obtained from random variables that are uncorrelated at the population level. The procedure is implemented by first obtaining the sampling distributions of the sequential eigenvalues under the null hypothesis of independent variables. This is achieved through the generation of a large number of matrices of random data with the same number of cases and variables as the empirical data matrix under consideration. Then, the number of factors is determined by comparing the empirical eigenvalues with the distribution of the corresponding random eigenvalues. Starting from the largest eigenvalue, factors are retained as long as their empirical eigenvalue is greater than the eigenvalue of its random counterpart. Parallel analysis has been one of the most studied and accurate dimensionality assessment methods for continuous and categorical variables to date (Crawford et al., 2010; Garrido et al., 2013, 2016; Ruscio & Roche, 2012; Schmitt, 2011; Timmerman & Lorenzo-Seva, 2011).

Although Horn (1965) based PA on the eigenvalues obtained from the full correlation matrix using principal component analysis (PApca), Humphreys and Ilen (1969) suggested that a more precise estimate of the number of common factors could be obtained by computing the eigenvalues from a reduced correlation matrix with estimates of communalities in its diagonal using principal axis factoring (PApaf). As a communality estimate they chose the squared multiple correlations between each variable and all the others. Even though these two variants of PA have not been compared frequently, Crawford et al. (2010) found that for continuous variables their overall accuracies were similar for structures of one, two, and four factors (60% for PApca and 65% for PApaf), with neither method being superior to the other across all the studied conditions. With categorical variables of two to five response options, however, Timmerman and Lorenzo-Seva (2011) found that PApca clearly outperformed PApaf for structures of one and three major factors (overall accuracies of 95% for PApca and 70% for PApaf).

Automated scree test methods. The scree test optimal coordinate (OC) and acceleration factor (AF) methods (Raiche et al., 2013) constitute two non-graphical solutions to Cattell’s scree test (Cattell, 1966). For \( p \) number of variables, the OC procedure aims to identify the actual factors by computing \( p-2 \) two-point regression models, and verifying if the eigenvalue in question is greater than the one estimated by these models. The last positive verification, starting from the second eigenvalue, and continuing without interruption, is used to determine the number of factors to retain. The predicted eigenvalue \( \hat{\lambda}_i \), known as the optimal coordinate, is estimated through the linear regression model using only the last eigenvalue and the \((i + 1)^{th}\) eigenvalue so that

\[
\hat{\lambda}_i = a_{i(i+1)} + b_{i(i+1)}(i)
\]

where

\[
b_{i(i+1)} = (\lambda_p - \hat{\lambda}_{i+1})/(p - i - 1)
\]

and

\[
a_{i(i+1)} = \lambda_{i+1} - b_{i+1}(i+1)
\]

On the other hand, the AF method searches for the point in the eigenvalue plot where the slope of the curve changes abruptly. In order to achieve this, the AF evaluates an approximation to the second derivative of Equation 3 at each of the \( i \)th eigenvalues (from 2 to \( p - 1 \)) using the function:

\[
f''(i) = f(i + 1) - 2f(i) - f(i - 1)
\]

Additionally, Raiche et al. (2013) complemented the OC and AF methods with the K1 rule or PApca, such that no eigenvalues are retained that are below one (K1) or below the eigenvalue obtained from independent variates (PApca). In their validation study with continuous variables, Raiche et al. (2013) found that the percentage of correct dimensionality estimates of OC (49%) was comparable to that of PA (53%), and between moderately to considerably higher than those for AF (39%), the Cattell-Nelson-Gorsuch scree test (30%), the K1 rule (21%), and the standard error scree (9%), among other methods. Similarly, Ruscio and Roche (2012) showed that the OC (74%), PA (76%), and the Akaike information criterion (73%) had comparable accuracies that were notably higher than other methods including the BIC (60%), MAP (60%), the chi-square test of model fit (59%), AF (46%), and K1 (9%).

Method

Design. The current study used a mixed factorial design to assess the accuracy of methods. The dimensionality assessment method was the within-subjects factor manipulated. The following seven techniques were employed: the original EGA with GLASSO and the recently proposed EGAtmf, two automated versions of the scree test (AF and OC), the K1 rule, and two versions of PA (PApca and PApaf). As per current recommendations, the PA method was computed by performing resampling of the original data matrices (20 resampled data matrices per replicate) and using the mean criterion to aggregate the resampled eigenvalues (Garrido et al., 2013; Lubbe, 2018). All methods used tetrachoric correlations as measures of association between the binary variables that were simulated. The between factors portion of the design included five variables that were manipulated using Monte Carlo methods: sample size (500, 1000 and 5000), factor loadings (.40, .55 and .70), variables per factor (4, 8 and 12), number of factors (3, 6 and 9), and factor correlations (0.00, 0.30, 0.50 and 0.70). In general, the levels of the variables were chosen so that they were representative of the range of values that can be encountered in applied settings (Garrido et al., 2011, 2013). As all the variables were simulated to be dichotomous, which produce greater sampling error than variables with more response options, larger samples sizes than typical were considered. In total, a \( 3 \times 3 \times 3 \times 4 \times 5 \) (sample size, factor loadings, variables per factor, number of factors, and factor correlations) design was implemented.
resulting in 324 simulated conditions.

In order to generate more realistic factor structures several steps were undertaken. First, the factor loading for each item was drawn randomly from a uniform distribution with values ranging from ± .10 of the specified level manipulated (e.g., for the level of .40 the loadings were drawn from the range of .30 to .50). Second, as it is common in practice to find complex structures in which items present non-zero loadings in multiple factors, we generated cross-loadings consistent to those commonly found in real data (Bollmann, Heene, Küchenhoff, & Bühner, 2015). The cross-loadings were generated following the procedure described in Meade (2008) and Garcia-Garzón, Abad and Garrido (in press): cross-loadings were randomly drawn from a normal distribution, $N(0, 0.05)$, for all the items except for the first two in each factor, which were set as markers (i.e., all of their cross-loadings were fixed to zero). Third, the magnitude of skewness for each item was randomly drawn with equal probability from a range of -2 to 2 in increments of 0.50, following Garrido et al. (2013). A skewness level of zero corresponds to a symmetrical distribution, while ±1 can be categorized as a meaningful departure from normality (Meyers, Gamst, & Guarino, 2006, p. 50) and ±2 as a high level of skewness (Muthén & Kaplan, 1985).

**Data generation.** For each of the 342 simulated conditions, 500 sample data matrices of binary variables were generated according to the common factor model procedure described next. First, the reproduced population correlation matrix (with communalities in the diagonal) was computed:

$$R_R = \Lambda \Phi \Lambda'$$

where $R_R$ is the reproduced population correlation matrix, $\lambda$ (Λ) is the measurement model (i.e. a $k \times r$ factor loading matrix for $k$ variables and $r$ factors) and $\phi$ (Φ) is the structure matrix of the latent variables (i.e. a $r \times r$ matrix of correlations among factors).

The population correlation matrix $R_p$ was then obtained by inserting units in the diagonal of $R_R$, thereby raising the matrix to full rank. The next step was performing a Cholesky decomposition of $R_p$, such that:

$$R_p = U'U$$

If either $R_p$ was not positive definite (i.e., at least one eigenvalue was ≤ 0) or an item’s communality was greater than 0.90, the $\Lambda$ matrix was replaced and a new $R_p$ matrix was computed following the same procedure. Subsequently, the sample data matrix of continuous variables was computed as:

$$X = ZU$$

where $Z$ is a matrix of random standard normal deviates with rows equal to the sample size and columns equal to the number of variables.

The resulting continuous variables were dichotomized by applying a set of thresholds according to the specific levels of skewness simulated (see Garrido et al., 2013, for the threshold values). For each sample data matrix generated, the convergence of EGA with GLASSO estimation was verified. If the analysis did not generate a numeric estimation (i.e. number of factors), the sample data matrix was discarded and a new one was generated, until we obtained 500 sample data matrices per condition.

**Data analysis.** We used $R$ (R Core Team, 2017) for all our analyses. The AF and OC techniques were computed using the nFactors package (Raiche, 2010), while PA with resampling was applied using the fa.parallel function contained in the psych package (Revelle, 2018). Both versions of EGA were computed using modified versions of the functions available at the EGA package (Hudson Golino, 2018). The figures were generated using the ggplot2 package (Wickham, 2016), the ggpubr package (Kassambara, 2017), R base package (R Core Team, 2017), and Excel. The paper was written following a reproducible approach, integrating text and code into two sets of files. The first set has all the codes used in the simulation. The second set contains an R Markdown file integrating the manuscript text and the codes used for the statistical and graphical analysis presented in the results’ section. The papaja package (Aust & Barth, 2018) was used to easily create a document following the APA guidelines.

In order to evaluate the performance of the dimensionality methods three complementary criteria were used: the percentage of correct estimates (PC), the mean bias error (MBE), and the mean absolute error (MAE). The corresponding formula for each criterion is as follows:

$$PC = \frac{\sum C}{N_s} \times 100$$

where $C$ is the number of sample data matrices.

$$MBE = \frac{\sum (\hat{\theta} - \theta)/N_s}{N_s}$$

$$MAE = \frac{\sum |\hat{\theta} - \theta|/N_s}{N_s}$$

where $\hat{\theta}$ is the estimated number of factors, $\theta$ is the population number of factors, and $N_s$ is the number of sample data matrices simulated.

The PC criterion has boundaries of 0% and 100%, with 0% signaling complete inaccuracy and 100% indicating perfect accuracy. In the case of the MBE, a 0 reflects a total lack of bias, while negative and positive values denote underfactoring and overfactoring, respectively. Regarding the MAE criterion, higher values signal larger departures from the population number of factors, while a value of 0 indicates perfect estimation accuracy.

Finally, analyses of variance (ANOVA) were conducted to investigate how the factor levels and their combinations impacted the accuracy of the dimensionality methods. The PC and MAE were set (separately) as the dependent variables and the manipulated variables constituted the independent factors. The MBE criterion was not used in the ANOVAs because it is not a measure of estimation accuracy, as errors of under- and overfactoring can compensate each other. The partial eta squared ($\eta_p^2$) measure of effect size was used to assess the magnitude of the main effects and interactions, per technique. According to Cohen (1992), $\eta_p^2$ values of 0.01, 0.06, and 0.14 can be considered as small, medium, and large effect sizes, respectively.
**Results**

**Overall performance.** The overall performance of the dimensionality methods, as well as their performance across the levels of the independent variables, is presented in Table 1. According to the accuracy of the methods shown in the table, the methods can be classified into two groups: those with low accuracy (AF, OC, and K1) and those with high accuracy (EGA, EGAtmfg, PApca, and PApaf). In terms of the PC criterion, the methods were, from best to worst: EGA 72.33%, PApaf 70%, PApca 68.39%, EGAtmfg 65.42%, OC 32.89%, K1 26.98%, and AF 22.72%. In terms of the MBE, the PApaf method showed the least overall bias, with a very small tendency to overfactor (0.10), followed by EGAtmfg (-0.60), OC (-0.83), and PApca (-0.88), which had a moderate tendency to underfactor. The rest of the methods had considerable larger MBEs, with EGA (2.58) and K1 (7.09) overfactoring, and AF (-3.86) underfactoring. Regarding the MAE, the two best methods were PApaf(0.72) and EGAtmfg (0.73), followed at a notch below by PApca (1.07); the remaining methods, OC (2.48), EGA (3.06), AF (3.86), and K1 (7.12), produced MAEs that were markedly worse.

Looking at the performance of the methods across the factor levels in Table 1 reveals several notable trends. First, while EGA was the best method in terms of PC, presenting the largest value in 43.75% of the conditions summarized in Table 1, PApaf presented the largest PC value in only 25% of them. Notably, of the four methods with high accuracy none produced the best estimates across all criteria for all factor levels, with each showing strengths and weaknesses. For example, whereas EGA was the best method for all three criteria with samples of 5000, it produced the second worst MAE (7.56) with samples of 500. Similarly, EGAtmfg was uniformly the best method for structures of three factors, but markedly inferior in terms of PC (but not MAE) with large structures of nine factors (47%), in general.

The three worst methods (AF, OC, and K1) did not produce accuracy levels (PC or MAE) that were close to the four best methods for any of the factor levels studied with one exception: AF produced the best PC of all the methods (88%) and a low MAE (0.51) for orthogonal structures. However, the accuracy of this method was nearly zero for oblique structures (PC of 2%, 0% and 0% for factor correlations of 0.30, 0.50, and 0.70, respectively). A closer look at the estimations of AF revealed that for oblique structures it suggested one-factor structures in 99.0% of the cases. Oblique structures will produce more than one break in the scree plot, and it appears that the AF method focused on the first break in the plot rather than the last. Third, as expected, the majority of the methods produced better estimates with larger samples, higher factor loadings, more variables per factor, smaller structures, and lower factor correlations. The exceptions were the OC and K1 methods, which produced inconsistent or reversed trends for variables per factor and factor correlations, and EGA for the MAE criterion related to the number of variables per factor.

In order to better understand the performance of the dimensionality methods, we present in Figure 2 box plots for the PC, MBE, and MAE values across the 324 conditions, as well as the performance value for each condition (represented by circles). In terms of the PC, the ordering of the methods from most to least variable was: K1 (SD = 0.41), PApca (SD = 0.41) AF (SD = 0.40), EGA (SD = 0.38), EGAtmfg (SD = 0.37), PApaf (SD = 0.36), and OC (SD = 0.30). The methods with the highest medians were PApca (99%) and EGA (97.80%), followed a step below by PApaf (91.20%) and then EGAtmfg (83.70%), with the rest having very low medians (OC = 19.30%, K1 = 0%, and AF = 0%). In the case of OC, Figure 2 shows that its low variability in PC is a result of the method performing consistently poorly; it is the only method that didn’t have 100% accuracy for any condition (the next method with the least conditions with a PC of 100% was K1, which had 40). In contrast, AF had the unusual pattern discussed previously of performing either very well or very poorly, with very few cases in between.

A look at the MBE in Figure 2 reveals that the two methods that had the most variability in terms of bias were EGA (SD = 10.05) and K1 (SD = 8.52), followed well below by AF (SD = 2.96), OC (SD = 2.32), and PApca (SD = 2.07), and then by the two least variable methods, PApaf (SD = 1.26), and EGAtmfg (SD = 1.13). The K1 and EGA methods were the only ones that showed a marked tendency to overfactor; however, in the case of K1 a substantial amount of conditions had notably large positive bias (29.3% of the conditions had MBEs > 10), whereas for EGA this occurred much less frequently (6.8% of the conditions had MBEs > 10), but with a few more extreme cases (MBEs as high as 70). The pattern for the rest of the methods was similar in that the conditions with the largest MBEs in absolute value were generally in the direction of underfactoring. In general, EGAtmfg and PApaf were the two methods with the smallest MBEs across conditions. Finally, regarding the MAE, the results mirrored those obtained for the MBE in that the variability for the methods in the MAE was very similar to their variability in the MBE. Again, EGA and K1 were the most variable and had the most extreme MAEs, and EGAtmfg and PApaf were the least variable with the smallest MAEs across conditions.

Figure 3 depicts the variability of the best four techniques per condition, summarized across the number of factors. When the factor loading is low (0.4) the techniques performed well (mean accuracy ≥ 90%) almost always for high sample sizes (5,000), irrespective of the number of variables, for correlations of zero and .3. When the correlation is .5 and the sample size is 5,000, EGA, PApca and EGAtmfg presented accuracies greater than 90% with 8 or 12 variables per factor. With correlations of .7, EGA only presented very high accuracies (mean accuracy ≥ 90%) for three factor structures with 8 or 12 items, and PApca when the number of factors was three and six, with 12 items each.

For moderate loadings (0.55), the four best methods performed well for correlations of zero and 0.3 with a sample size of 1,000 and 5,000. With lower sample sizes (i.e. 500), the techniques performed well mainly when the number of items was 8 or 12. When the correlation between factors is moderate (0.50), they tend to perform well with a sample size of 1,000 and eight or 12 items, or with a sample size of 5,000 in almost all the cases. The performance of the methods changes when the correlation between factors is high (0.70). With sample sizes of 1,000, EGA only presented higher PC values with 12 variables per factor, while PApca and EGAtmfg didn’t present high accuracies even with 12 variables per factor. For large sample sizes (5,000) the tendency for EGA, PApaf and PApca was to present high PC values for 8 or 12 items per factor.
Table 1. Performance of the dimensionality methods across the levels of the independent variables and in total

<table>
<thead>
<tr>
<th>Method</th>
<th>Sample Size</th>
<th>Factor Loading</th>
<th>Variables per Factor</th>
<th>Number of Factors</th>
<th>Factor Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>500 1000 5000</td>
<td>0.40 0.55 0.70</td>
<td>4 8 12</td>
<td>3 6 9</td>
<td>0.00 0.30 0.50 0.70</td>
</tr>
<tr>
<td></td>
<td>Percentage Correct (PC)</td>
<td>Mean Absolute Error (MAE)</td>
<td>Mean Bias Error (MBE)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AF</td>
<td>19.7 23.1 25.3</td>
<td>18.3 24.3 25.6</td>
<td>20.6 23.2 24.3</td>
<td>25.8 22.1 20.3</td>
<td>88.5 2.4 0.0 0.0</td>
</tr>
<tr>
<td>OC</td>
<td>23.9 31.8 42.9</td>
<td>17.2 31.2 50.3</td>
<td>21.3 39.6 37.8</td>
<td>52.6 27.4 18.6</td>
<td>27.9 37.0 37.0 29.7</td>
</tr>
<tr>
<td>K1</td>
<td>8.5 17.8 54.6</td>
<td>4.7 24.1 52.3</td>
<td>45.6 20.8 14.6</td>
<td>41.7 22.8 16.4</td>
<td>28.1 28.0 28.1 23.8</td>
</tr>
<tr>
<td>PApca</td>
<td>52.0 65.4 87.7</td>
<td>46.7 75.0 83.5</td>
<td>52.8 72.5 79.9</td>
<td>81.1 66.4 57.7</td>
<td>86.1 84.9 70.3 32.3</td>
</tr>
<tr>
<td>PApaf</td>
<td>54.6 66.3 89.1</td>
<td>36.2 78.3 95.5</td>
<td>65.3 70.1 74.7</td>
<td>77.8 69.7 62.5</td>
<td>81.1 77.0 69.3 52.7</td>
</tr>
<tr>
<td>EGA</td>
<td>54.2 71.1 91.7</td>
<td>41.3 81.3 94.4</td>
<td>62.3 75.9 78.8</td>
<td>84.5 71.4 61.1</td>
<td>84.3 78.8 71.0 55.2</td>
</tr>
<tr>
<td>EGAtmfg</td>
<td>53.3 64.2 78.7</td>
<td>46.2 68.8 81.3</td>
<td>45.0 72.5 78.7</td>
<td>84.8 64.1 47.4</td>
<td>86.3 75.6 60.5 39.4</td>
</tr>
</tbody>
</table>

Note. AF = scree test acceleration factor; OC = scree test optimal coordinate; K1 = eigenvalues-greater-than-one rule; PApca = parallel analysis with principal component analysis; PApaf = parallel analysis with principal axis factoring; EGA = exploratory graph analysis with the graphical LASSO; EGAtmfg = exploratory graph analysis with the triangulated maximally filtered graph approach. The best column values are bolded and underlined (highest PC, closest MBE to zero, and lowest MAE). Values close to the best column values are highlighted in grey (within 0.10 of the highest PC, absolute values within 0.50 of the lowest absolute MBE, and within 0.50 of the lowest MAE).
Figure 2. Box plots illustrating the performance of the methods across the simulated conditions. AF = scree test acceleration factor; OC = scree test optimal coordinate; K1 = eigenvalues-greater-than-one rule; PApca = parallel analysis with principal component analysis; PApaf = parallel analysis with principal axis factoring; EGA = exploratory graph analysis with the graphical LASSO; EGAtmfg = exploratory graph analysis with the triangulated maximally filtered graph approach. The horizontal lines indicate the median values, the numbers inside the plots are the mean values, and the circles are the values corresponding to each condition.
The best scenario was achieved with a factor loading of .7. The accuracies are high across conditions, especially for correlations of zero, .3 and .5. It’s notorious that for correlations of .7, EGA and PApaf were very consistent, presenting higher PC values in most cases with samples of 5,000. For sample sizes of 1,000, both techniques were highly accurate when the number of variables per factor were 8 or 12.

Impact of the independent variables on the performance. In this section we will further evaluate the impact of the independent variables on the performance of the best four methods (EGA, EGAtmfg, PApca, and PApaf) through ANOVAs, using the two accuracy criteria (PC and MAE) separately as the dependent variables. The effect sizes derived from these analyses are shown in Table 2.

The results in Table 2 show that the variables that most impacted each of the methods were: the factor loadings for EGA ($\eta_p^2[PC] = \eta_p^2[MAE] = 0.47$) and PApaf ($\eta_p^2[PC] = 0.43$; $\eta_p^2[MAE] = 0.37$), the factor correlations for EGAtmfg ($\eta_p^2[PC] = 0.26$) and PApca ($\eta_p^2[PC] = 0.48$; $\eta_p^2[MAE] = 0.77$), also the number of factors for EGAtmfg ($\eta_p^2[MAE] = 0.40$). In all cases these results coincided with the conditions where the methods showed the largest spread between the manipulated factor levels (e.g., EGA had PCs ranging from 41.3% to 94.4% for factors loadings from .40 to .70). In addition to these main effects, three of the four methods (all except PApaf) showed at least one interaction with a large effect size. Because the effect sizes were larger for the MAE criterion, they will be our focus.

Figure 3. Mean accuracy by conditions for the best four methods. PApca = parallel analysis with principal component analysis; PApaf = parallel analysis with principal axis factoring; EGA = exploratory graph analysis with the graphical LASSO; EGAtmfg = exploratory graph analysis with the triangulated maximally filtered graph approach.
The EGA method had multiple interactions with large effect sizes, but all of them were contained within the four-way sample size x factor loading x variables per factor x number of factors interaction ($\eta^2_p = 0.15$), which is presented in Figure 4. The four-way interaction can be described as follows: for smaller samples (particularly $N = 500$) the MAE was considerably higher for structures that had lower factor loadings, but especially if they were combined with more factors and a larger number of variables per factor, where the MAE reached values as high as 69 or 37. In contrast, for very large samples (i.e., $N = 5000$), the performance was very similar (highly accurate) across different factor loadings, number of factors, and variables per factor.

In the case of EGAtmg, it produced a large two-way interaction for variables per factor x number of factors ($\eta^2_p = 0.24$). This two-way interaction, presented in Figure 5, emerged as a result of the differential effect that the number of factors had for structures with varying numbers of variables per factor. Specifically, whereas for smaller structures of three factors the MAEs were very similar across the levels of variables per factor (between .12 and .21), they became very different with more factors, in particular for nine factors and four variables per factor (MAE = 2.68 vs. 1.02 for eight variables per factor, and .58 for 12 variables per factor). Also shown in Figure 4 is the largest interaction for the PApaf method, the two-way sample size x factor loading interaction, which although did not reach the large effect size threshold, it was relatively close ($\eta^2_p = 0.11$). This interaction shows that while the accuracy of PApaf across factor loadings was relatively similar with very large samples of 5000 observations (between .00 and .53) it was much worse for factor loadings of .40 combined with smaller samples, especially those with 500 cases (MAE = 2.46 vs. .91 and .26 for factor loadings of .53 and .70 respectively).

The last interaction presented is the four-way sample size x variables per factor x number of factors x factor correlation for PApca ($\eta^2_p = 0.14$), which contained all the salient three-way and two-way interactions for this method (Figure 6). This four-way interaction may be described in the following manner: with more factors the differences in MAE across the factor correlations became greater (higher factor correlations produced much higher

**Table 2. ANOVA partial eta squared ($\eta^2_p$) effect sizes for the percentage correct (PC) and mean absolute error (MAE) criterion variables**

<table>
<thead>
<tr>
<th>Variables</th>
<th>PC</th>
<th>MAE</th>
<th>PC</th>
<th>MAE</th>
<th>PC</th>
<th>MAE</th>
<th>PC</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>.29</td>
<td>.40</td>
<td>.20</td>
<td>.19</td>
<td>.29</td>
<td>.37</td>
<td>.11</td>
<td>.14</td>
</tr>
<tr>
<td>FLOAD</td>
<td>.32</td>
<td>.35</td>
<td>.43</td>
<td>.37</td>
<td>.47</td>
<td>.47</td>
<td>.19</td>
<td>.22</td>
</tr>
<tr>
<td>VARFAC</td>
<td>.20</td>
<td>.41</td>
<td>.02</td>
<td>.01</td>
<td>.08</td>
<td>.12</td>
<td>.20</td>
<td>.33</td>
</tr>
<tr>
<td>FAC</td>
<td>.15</td>
<td>.56</td>
<td>.04</td>
<td>.12</td>
<td>.14</td>
<td>.19</td>
<td>.21</td>
<td>.40</td>
</tr>
<tr>
<td>FCORR</td>
<td>.48</td>
<td>.77</td>
<td>.12</td>
<td>.17</td>
<td>.17</td>
<td>.00</td>
<td>.26</td>
<td>.28</td>
</tr>
<tr>
<td>N * FLOAD</td>
<td>.09</td>
<td>.11</td>
<td>.08</td>
<td>.11</td>
<td>.15</td>
<td>.51</td>
<td>.02</td>
<td>.02</td>
</tr>
<tr>
<td>N * VARFAC</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.19</td>
<td>.00</td>
<td>.03</td>
</tr>
<tr>
<td>N * FAC</td>
<td>.03</td>
<td>.21</td>
<td>.01</td>
<td>.05</td>
<td>.02</td>
<td>.18</td>
<td>.00</td>
<td>.05</td>
</tr>
<tr>
<td>N * FCORR</td>
<td>.03</td>
<td>.21</td>
<td>.01</td>
<td>.06</td>
<td>.01</td>
<td>.01</td>
<td>.00</td>
<td>.01</td>
</tr>
<tr>
<td>FLOAD * VARFAC</td>
<td>.01</td>
<td>.01</td>
<td>.01</td>
<td>.01</td>
<td>.02</td>
<td>.26</td>
<td>.00</td>
<td>.02</td>
</tr>
<tr>
<td>FLOAD * FAC</td>
<td>.02</td>
<td>.13</td>
<td>.01</td>
<td>.07</td>
<td>.04</td>
<td>.29</td>
<td>.01</td>
<td>.08</td>
</tr>
<tr>
<td>FLOAD * FCORR</td>
<td>.02</td>
<td>.07</td>
<td>.02</td>
<td>.04</td>
<td>.03</td>
<td>.00</td>
<td>.01</td>
<td>.02</td>
</tr>
<tr>
<td>VARFAC * FAC</td>
<td>.00</td>
<td>.16</td>
<td>.00</td>
<td>.01</td>
<td>.02</td>
<td>.06</td>
<td>.07</td>
<td>.24</td>
</tr>
<tr>
<td>VARFAC * FCORR</td>
<td>.14</td>
<td>.40</td>
<td>.01</td>
<td>.02</td>
<td>.03</td>
<td>.00</td>
<td>.01</td>
<td>.04</td>
</tr>
<tr>
<td>FAC * FCORR</td>
<td>.03</td>
<td>.57</td>
<td>.01</td>
<td>.09</td>
<td>.02</td>
<td>.00</td>
<td>.02</td>
<td>.12</td>
</tr>
<tr>
<td>N * FLOAD * VARFAC</td>
<td>.01</td>
<td>.02</td>
<td>.01</td>
<td>.01</td>
<td>.06</td>
<td>.33</td>
<td>.01</td>
<td>.01</td>
</tr>
<tr>
<td>N * FLOAD * FAC</td>
<td>.01</td>
<td>.03</td>
<td>.01</td>
<td>.01</td>
<td>.03</td>
<td>.29</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>N * FLOAD * FCORR</td>
<td>.07</td>
<td>.06</td>
<td>.08</td>
<td>.02</td>
<td>.08</td>
<td>.02</td>
<td>.03</td>
<td>.03</td>
</tr>
<tr>
<td>N * VARFAC * FAC</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.07</td>
<td>.00</td>
<td>.02</td>
</tr>
<tr>
<td>N * VARFAC * FCORR</td>
<td>.09</td>
<td>.22</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
</tr>
<tr>
<td>N * FAC * FCORR</td>
<td>.01</td>
<td>.12</td>
<td>.01</td>
<td>.07</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
</tr>
<tr>
<td>FLOAD * VARFAC * FAC</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.13</td>
<td>.01</td>
<td>.02</td>
</tr>
<tr>
<td>FLOAD * VARFAC * FCORR</td>
<td>.03</td>
<td>.12</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
<td>.03</td>
<td>.02</td>
</tr>
<tr>
<td>FLOAD * FAC * FCORR</td>
<td>.02</td>
<td>.02</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
</tr>
<tr>
<td>VARFAC * FAC * FCORR</td>
<td>.05</td>
<td>.13</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
<td>.00</td>
<td>.02</td>
<td>.02</td>
</tr>
<tr>
<td>N * FLOAD * VARFAC * FAC</td>
<td>.01</td>
<td>.01</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.15</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>N * FLOAD * VARFAC * FCORR</td>
<td>.06</td>
<td>.12</td>
<td>.01</td>
<td>.01</td>
<td>.03</td>
<td>.01</td>
<td>.01</td>
<td>.02</td>
</tr>
<tr>
<td>N * FLOAD * FAC * FCORR</td>
<td>.04</td>
<td>.03</td>
<td>.02</td>
<td>.01</td>
<td>.03</td>
<td>.01</td>
<td>.01</td>
<td>.02</td>
</tr>
<tr>
<td>N * VARFAC * FAC * FCORR</td>
<td>.02</td>
<td>.14</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>FLOAD * VARFAC * FAC * FCORR</td>
<td>.01</td>
<td>.04</td>
<td>.00</td>
<td>.00</td>
<td>.01</td>
<td>.00</td>
<td>.01</td>
<td>.01</td>
</tr>
<tr>
<td>N * FLOAD * VARFAC * FAC * FCORR</td>
<td>.07</td>
<td>.07</td>
<td>.01</td>
<td>.02</td>
<td>.03</td>
<td>.01</td>
<td>.02</td>
<td>.01</td>
</tr>
</tbody>
</table>

Note. PApca = parallel analysis with principal component analysis; PApaf = parallel analysis with principal axis factoring; EGA = exploratory graph analysis with the graphical LASSO; EGAtmg = exploratory graph analysis with the triangulated maximally filtered graph approach. N = sample size; FLOAD = factor loading; VARFAC = variables per factor; FAC = number of factors; FCORR = factor correlation. Large effect sizes ($\eta^2_p \geq 0.14$) are bolded, underlined, and highlighted in grey.
However, as both the sample size and the number of variables per factor increased these differences across factor correlations became smaller, to the point that they were negligible with samples of 5000 observations and 12 variables per factor (all MAEs were between .00 and .18).

**Figure 4.** Four-way sample size x factor loading x variables per factor x number of factors interaction for the exploratory graph analysis (EGA) method. N = sample size; VARFAC = variables per factor; FLOAD = factor loading.

**Figure 5.** Two-way variables per factor x number of factors interaction for EGAtmfg and two-way sample size x factor loading interaction for PApaf. EGAtmfg = exploratory graph analysis with the triangulated maximally filtered graph approach; PApaf = parallel analysis with principal axis factoring; VARFAC = variables per factor; FLOAD = factor loading.
Figure 6. Four-way sample size x variables per factor x number of factors x factor correlation interaction for parallel analysis with principal component analysis extraction (PApca) method. N = sample size; VARFAC = variables per factor; FCORR = factor correlation.

A final comparison between the four best performing methods is shown in Figure 7 (PC criterion) and Figure 8 (MAE criterion). The heat maps contained in these figures indicate the performance of the methods across each of the 324 conditions simulated, while making it possible to identify the factor levels corresponding to each one. Several trends are noticeable in these figures: first, none of the methods were effective with loadings of .40, unless the sample size was very large (N = 5000) and the factor correlations were not too high (≤ .50). Second, only EGA and PApaf could consistently recover structures with factor correlations of .70, but they needed samples bigger than 1000 observations to do so. In contrast, both EGAtmfg and PApca showed a bias when the number of variables per factor was small (4), so that they couldn’t accurately recover the dimensionality for these structures even with factor loadings of .70 and sample sizes of 5000 cases. Third, EGAtmfg was the only method that did not produce a MAE of five or greater for any condition, thus emerging as the method that could be counted on to consistently provide relatively close estimates to the population dimensionality. Fourth, when comparing the two EGA methods it was apparent that EGAtmfg would be the preferred for structures with low loadings (.40) and samples of 1000 or less observations, and that EGA would be the method of choice for structures with medium (.55) or high (.70) loadings.

**How to use EGA in R.** In order to demonstrate how to implement EGA in R, a brief example will be presented below. In this example, Goldberg’s 50-item Big-Five personality scale (Goldberg, 1992) was analyzed. This scale contains 50 items designed to measure five broad factors of personality. The dataset used in the current paper was part of the Virginia Cognitive Aging Project (VCAP; Salthouse, 2018), and was composed of 2247 people that answered all the scale items in the first measurement occasion, between 2001 and 2017. The sample participants were 64.8% women, had ages ranging from 18 to 97 years (M = 50.72, SD = 18.73), and had an average of 15.65 years of education.

The EGA package can be installed in R using the devtools package (Wickham & Chang, 2017), since the source code is available on Github. Thus, the first step is to load the devtools package, and then install the EGA package using the install_github function:

```r
library("devtools")
devtools::install_github("hfgolino/EGA")
```
Figure 7. Heat map illustrating the percentage of correct estimates (PC) values across the simulated conditions. N = sample size; FLOAD = factor loading; VF = variables per factor; F = number of factors; FC = factor correlation.
**Figure 8.** Heat map illustrating the mean absolute error (MAE) values across the simulated conditions. N = sample size; FLOAD = factor loading; VF = variables per factor; F = number of factors; FC = factor correlation.
The EGA package was developed as a simple and easy way to implement the exploratory graph analysis technique. The package has several functions, but only two will be used in this tutorial: EGA and CFA. The first function, EGA, estimates the number of dimensions of a given dataset/instrument using GLASSO with the lambda parameter set via EBIC or using the TMFG method. The number of underlying dimensions (or factors) is detected using the walktrap algorithm. There are four arguments in the EGA function: data, model, plot.EGA and n. The first one specifies the dataframe with the variables to be used in the analysis, or, a correlation matrix. If the data used is a correlation matrix, then the fourth argument n (sample size) needs to be specified. The second argument is a string indicating the method to use: glasso for the Gaussian graphical model estimation using graphical LASSO with the EBIC criterion to select the optimal regularization parameter (default method) or tmfg for estimating a Triangulated Maximally Filtered Graph. The third argument, plot.EGA, is a logical argument indicating whether a plot of the network with the estimated dimensions should be returned. The next chunks of code show how to implement the EGA analysis using the EGA package.

To use the EGA package, the first step is to load it using the library function. Then, the dataset should be imported into R. In this case, Goldberg’s Big-Five dataset composed of dichotomous variables is saved as a .csv file in the local directory, so the function to import the dataset into R is the read.csv function. An object named personality can be created to store the data and, as a last step, the EGA function is used. It is important to note that before importing the dataset the reversed items had been recoded so that all the items were in the same direction.

```
library("EGA")
personality <- read.csv("./Datasets/personality.csv")
ega.personality <- EGA(data = personality, model = "glasso", plot.EGA = TRUE)
```

The results in Figure 9 show five dimensions for Goldberg’s 50-item personality scale. The 10 items of Imagination/Intellect (items Im1 to Im9 and I10) formed the first dimension or factor. The second dimension, by its turn, had all the items of Conscientiousness (items Cn1 to Cn9 and C10), while the third dimension had all the items of Agreeableness (items Ag1 to Ag9 and A10) and one item of extraversion (item Ex3). The fourth dimension estimated via EGA contained all the Emotional Stability items (items Es1 to Es10), while the fifth dimension had nine Extraversion items (items Ex1 to E10, excluding Ex3). Although not shown here, EGA using tmfg also produced a network of five dimensions.

The CFA function can be used to verify via confirmatory factor analysis the fit of the simple structure corresponding to the item groupings identified by EGA:

```
CFA(ega.obj, estimator, plot.CFA = TRUE, data)
```

This function has four arguments. The first one, `ega.obj` specifies the name of the EGA object used in the previous step (in our case, the `ega.personality` object). The second argument (estimator) specifies the estimator to be used in the confirmatory factor analysis (WLSMV, ML, WLS, and other estimators available in the package lavaan). The third argument, plot.CFA, indicates if a CFA structure with its standardized loadings should be plotted (Figure 10). Finally, the data argument indicates the dataframe to be used in the analysis:

```
cfa.personality <- CFA(ega.obj = ega.personality, estimator = "WLSMV", plot.CFA = TRUE, data = personality)
```

Figure 9. Dimensionality structure of Goldberg’s 50-item Big-Five personality scale using exploratory graph analysis (EGA). Ag = agreeableness; Cn = conscientiousness; Ex = extraversion; ES = emotional stability; Im = Imagination.

Figure 10. Confirmatory Factor Model of 50-item Big-Five personality scale with five factors. Ag = agreeableness; Cn = conscientiousness; Ex = extraversion; ES = emotional stability; Im = Intellect; Ft = factor.
The fit of the model can be found using one of the following codes:

```r
cfa.personality$fit.measures
library(lavaan)
fitMeasures(cfa.personality$fit)
```

The first code above will print the chi-square, degrees of freedom, p-value, CFI, RMSEA, GFI and NFI fit indexes. The second code uses the `fitMeasures` function from `lavaan` to compute additional fit indexes. The five-factor model identified via EGA presented an okay fit (chi-square $[df = 1165] = 18,564.45, p < .001, CFI = 0.85, RMSEA = 0.07, GFI = 0.92, NFI = 0.85$).

To compare the EGA results with the results of parallel analysis with resampling, the `psych` package (Revelle, 2018) can be used. This package implements parallel analysis in the `fa.parallel` function:

```r
library("psych")
pa.personality <- fa.parallel(personality, fa = "both", cor = "poly", sim = FALSE, plot = TRUE)
```

As can be seen in Figure 11, the PApaf method suggested 10 factors, while PApca indicated that eight factors should be retained. Regarding the other methods, K1 suggested 11 factors (the 11th eigenvalue was 1.064 while the 12th was 0.999), which, as expected, was the highest dimensionality estimate of all the methods. In the case of the AF method, the most abrupt change in the plot (acceleration of 2.47) was found for the second eigenvalue, thus indicating that a one-factor structure should be retained. This result mirrors those of the simulation in that with correlated factors, as personality constructs are, the AF method will tend to suggest unidimensional structures. It is interesting to note that the second largest break in the eigenvalue plot was found at the sixth eigenvalue (acceleration of 0.71), which would have suggested a five-factor structure to be retained (the third largest acceleration was just 0.10). Thus, it appears at AF detected two large breaks, but focused on the incorrect one. Finally, OC estimated a nine-factor structure, finding significant differences between the actual and predicted 8th eigenvalues (1.30 vs. 1.18), but not between the actual and predicted 9th eigenvalues (1.15 vs. 1.14).

![Parallel Analysis Scree Plots](image)

*Figure 11. Parallel Analysis of 50-item Big-Five personality scale. PC = principal component analysis; FA = principal axis factoring.*

**Discussion**

The present study examined the dimensionality identification accuracy of two exploratory graph analysis methods, as well as several traditional factor-analytic-based techniques, using an extensive Monte Carlo simulation. Aside from manipulating five salient variables across ranges of plausible values that may be found in applied settings, all the structures that were generated had varying main factor loadings, cross-loadings, and skewness across items in order to enhance the ecological validity of the simulation. Overall, the EGA methods performed as well as the most accurate traditional technique, parallel analysis, and provided the best large-sample properties of all the methods evaluated. In addition, a straightforward R tutorial on how to use an interpret EGA was provided, and the method was applied to an empirical database composed of scores from a well-known Big Five personality test. This study extends previous research for EGA with GLASSO estimation by providing evidence of its accuracy across a broader set of conditions than previously considered, and is the first to examine the performance of EGA with TMFG, which emerges as an important complementary technique.
**Method performance.** The results from the simulation study revealed that the methods could be classified into two groups, those with high accuracy (EGA, EGAtmfg, PApca, and PApaf) and those with low accuracy (AF, OC, and K1). Of the high performing methods, none was the best across every variable and criteria, and all showed strengths and weaknesses.

EGA with GLASSO was the most accurate method with medium (.55) and high (.70) factor loadings, followed closely by PApaf. Also, of the four best methods, EGA and PApaf were the two most robust to the factor correlations, sustaining the smallest decreases in accuracy with higher factor correlations. The excellent performance of EGA in these conditions is in line with previous research (Golino & Demetriou, 2017; Golino & Epksamp, 2017). With low loadings (.40) combined with smaller samples (500), however, the performance of EGA was very poor. Although EGA in these conditions had rates of correct estimates that were in line with those of the other good methods, it tended to err by extremely large amounts. Indeed, it appears that in these conditions the method can “break down” and suggest implausible solutions, in some instances providing estimates of dimensionality equal to the number of variables in the database. Interestingly enough, with low loadings and very large samples (N = 5000), EGA was the best performing method (PC = 78.9%, MAE = 0.48) along with PApca (PC = 79.7%, MAE = 0.66); none of the other methods achieved an accuracy of 70%. Thus, it appears that the problem occurs only when both low loadings are combined with smaller samples, something that could be related to the regularization procedure.

EGA with TMFG provided correct estimates at a rate just below that of the other high performing methods, but its most notable characteristic was that its estimates, along those of PApaf, were the closest to the population values. Indeed, EGAtmfg provided the best MAE of all the methods for seven of 16 variable levels manipulated. In comparison to the other good performing methods, EGAtmfg was at its best for structures with fewer factors (3), weaker factor correlations (≤ .30), and many variables per factor (≥ 8). In contrast, the biggest limitations of EGAtmfg came for high dimensional structures that were composed of few variables per factor, and with highly correlated factors. It is likely that these conditions create problems for EGAtmfg due to the way it constructs the network, through the formation of tetrahedrons (groups of four nodes), which severely limits on the cross-dimension connections.

In terms of the two PA methods, they generally performed well, thus extending the vast literature supporting the accuracy of this procedure (e.g., Garrido et al., 2013, 2016; Timmerman & Lorenzo-Seva, 2012). In general, PApaf was the most consistent and least biased of all the methods examined, while PApca excelled with many variables per factor and weaker factor correlations (≤ .30). Also, these two methods complemented each other, with one being stronger where the other was weaker, and vice versa (e.g., for factor loadings, variables per factor, and factor correlations). In the case of PApca, the method showed a clear bias in the condition of few variables per factor (4) combined with very high factor correlations (.70). In these cases the method will generally produce a one-factor estimate regardless of the actual dimensionality of the data. The reason for this is simple: the population eigenvalues after that corresponding to the first factor will be lower than one, and thus, asymptotically PApca is not able to retain them. In terms of PApaf, it produced its comparatively poorest performance with low factor loadings (.40).

Regarding AF, OC, and K1 techniques, their percentage of correct estimates was more than 30% below to that of the EGA and PA methods. The AF method was one of the most accurate methods for orthogonal structures, but its accuracy plummeted to near zero for correlated factors. Indeed, in 99% of the datasets for oblique structures AF suggested a one-factor structure. The reason for this is because this method identifies the biggest break in the eigenvalue plot, and with correlated structures the biggest break usually occurs at the second eigenvalue. However, the correct way to determine dimensionality using the scree test is to identify the last “big” break (as correlated structures will produce more than one break) in the eigenvalue plot, but this is not how the method has been implemented. In the case of K1, the method tended to overestimate the population dimensionality by very large amounts, as has been widely documented in the literature (Costello & Osborne, 2005).

**EGA in practice.** Which EGA method should be used with empirical data? In this section we will provide some practical recommendations to guide researchers in the implementation of EGA and EGAtmfg.

On one hand, it is useful to always compute both EGA and EGAtmfg and see if their estimates agree. In our simulation, 58.0% of the cases where EGA erred it did so by overfactoring, while in 85.6% of the cases that EGAtmfg erred it was due to underfactoring. Thus, when the methods agree it is likely because they have found the optimal solution. For example, in this study EGA and EGAtmfg provided the same estimate for 63% of the datasets, and for these, their accuracy was nearly perfect (PC = 94.0%, MAE = 0.11). Therefore, if both EGA and EGAtmfg produce the same dimensionality estimate researchers can have increased confidence that the solution suggested is optimal, or if not, very close to it.

On the other hand, when the two methods disagreed in the present study the accuracy of both EGA (PC = 35.4%, MAE = 8.08) and EGAtmfg (PC = 16.8%, MAE = 1.79) was very poor. In these instances when EGA and EGAtmfg provide different estimates in practice, researchers can look at the heat maps provided in Figures 6 and 7 to see the method that is likely to perform better in the conditions that they think most apply to their data. In general, it is important to know that EGA is likely to perform better for medium (.45 to .65) and high (.60 to .80) factor loadings, while EGAtmfg is preferred for structures with low loadings (.30 to .50). Although researchers cannot generally know the size of the population loadings for their data, they can use previous research, as well as the exploration of different factor solutions, to gauge the potential range of loadings of their data. Additionally, in these cases where EGA and EGAtmfg disagree, it is important to more strongly consider potential alternative solutions (with less or more dimensions, respectively) to those suggested by the methods.

**Implications of EGA beyond psychology.** Thus far, EGA is extensively tested using measures of personality and other psychological phenomena. However, the EGA technique can be extended to any area dealing with multivariate data, and which dimensionality assessment plays an important role. One interesting area of application, with a broad impact around the globe is the area of healthy ageing, proposed by the World Health Organization (WHO, 2015). Healthy ageing is defined as the process of developing and maintaining the functional ability that enables wellbeing in older age, with functional ability determined by the intrinsic capacity of the individual, the environments they inhabit and the interaction between them (Beard et al., 2016; WHO, 2015). Functional ability comprises the health-related attributes that enable people to be and to do what they have reason to value. It is determined by the intrinsic capacity of the individual, i.e. the combination of all the individual’s physical and mental capacities. Environments comprise all the factors in the extrinsic world.
impactful variables in shown that the number of response options is usually one of the least should be noted, however, that previous dimensionality studies have dichotomous variables, due to the already large simulation design. It dimensionality estimate.

regularized partial correlations and will always produce a correlated and the sample size is small. In these conditions the estimate in some occasions, mainly when the variables are lowly and after it has been established that there is more than one is potential unidimensional we recommend that EGA is used only if (i.e., dimensions). Because of this, they are not useful a data, purposefully splitting the network into different communities (i.e., dimensions). Because of this, they are not useful and do not to work establish unidimensionality. For this reason, with data that is potential unidimensional we recommend that EGA is used only if and after it has been established that there is more than one dimension underlying the data (e.g., with parallel analysis).

Second, the EGA method may not produce a dimensionality estimate in some occasions, mainly when the variables are lowly correlated and the sample size is small. In these conditions the GLASSO regularized partial correlations will tend to zero and the model will not converge to a solution. When this happens, researchers can still use the EGAtmg method, which is not based on regularized partial correlations and will always produce a dimensionality estimate.

Third, the performance of the methods was evaluated only for dichotomous variables, due to the already large simulation design. It should be noted, however, that previous dimensionality studies have shown that the number of response options is usually one of the least impactful variables in the accuracy of the dimensionality estimates (Garrido et al., 2011, 2013). Nevertheless, future research is needed to understand how much the performance of EGA would improve given more response options in the data.

Fourth, there is the potential that the number of estimated dimensions matches the number of simulated dimensions, but that the items might not exactly correspond between the two. Future studies may want to consider additional measures of accuracy, such as normalized mutual information (NMI; Danon et al., 2005). The NMI provides a continuous estimate (between zero and one) of how well the estimated dimensions correspond to the known (or simulated dimensions), even if the number of dimensions is different. Therefore, NMI only produces a metric of one if the number of estimated dimensions and the items within those dimensions exactly match the number of simulated dimensions and their item content.

Finally, to extend the use of EGA to other areas, such as the growing field of healthy ageing, and compare the dimensionality structure of several instruments, require the development of fit indexes specifically designed to quantify the uncertainty in the estimation of the number of factors. This is necessary not only to compare theoretical structures with the results obtained via EGA, but also to expand the field of structural analysis, that relies in classical fit indexes like the confirmatory fit index (CFI) and the root mean square error of approximation (RMSEA), that are not reliable for the investigation of dimensionality (Clark & Bowles, 2018; Beierl, 2018; Garrido et al., 2016).

**Limitations and future directions**. The current study has several limitations of note and suggests potential avenues for future exploration. First, one-factor structures were not simulated. Community detection algorithms (such as the walktrap algorithm used by EGA) were designed to work with highly multidimensional data, purposefully splitting the network into different communities (i.e., dimensions). Because of this, they are not useful and do not to work establish unidimensionality. For this reason, with data that is potential unidimensional we recommend that EGA is used only if and after it has been established that there is more than one dimension underlying the data (e.g., with parallel analysis).

Second, the EGA method may not produce a dimensionality estimate in some occasions, mainly when the variables are lowly correlated and the sample size is small. In these conditions the GLASSO regularized partial correlations will tend to zero and the model will not converge to a solution. When this happens, researchers can still use the EGAtmg method, which is not based on regularized partial correlations and will always produce a dimensionality estimate.

Third, the performance of the methods was evaluated only for dichotomous variables, due to the already large simulation design. It should be noted, however, that previous dimensionality studies have shown that the number of response options is usually one of the least impactful variables in the accuracy of the dimensionality estimates (Garrido et al., 2011, 2013). Nevertheless, future research is needed to understand how much the performance of EGA would improve given more response options in the data.

Fourth, there is the potential that the number of estimated dimensions matches the number of simulated dimensions, but that the items might not exactly correspond between the two. Future studies may want to consider additional measures of accuracy, such as normalized mutual information (NMI; Danon et al., 2005). The NMI provides a continuous estimate (between zero and one) of how well the estimated dimensions correspond to the known (or simulated dimensions), even if the number of dimensions is different. Therefore, NMI only produces a metric of one if the number of estimated dimensions and the items within those dimensions exactly match the number of simulated dimensions and their item content.

Finally, to extend the use of EGA to other areas, such as the growing field of healthy ageing, and compare the dimensionality structure of several instruments, require the development of fit indexes specifically designed to quantify the uncertainty in the estimation of the number of factors. This is necessary not only to compare theoretical structures with the results obtained via EGA, but also to expand the field of structural analysis, that relies in classical fit indexes like the confirmatory fit index (CFI) and the root mean square error of approximation (RMSEA), that are not reliable for the investigation of dimensionality (Clark & Bowles, 2018; Beierl, 2018; Garrido et al., 2016).

**Final conclusion**. The findings from the present study suggest that EGA and EGAtmg are strong dimensionality assessment methods that can be recommended for applied use. Unlike other dimensionality techniques, EGA produces a visual guide–network plot–that not only indicates the number of dimensions to retain, but also which items cluster together and their level of association, all without the need from any further specification from users. The EGA methods are computationally fast, and can easily be implemented through a freely available R package that is outlined in the tutorial section of this manuscript. In sum, considering that dimensionality decisions have far reaching consequences for construct validation and theory development, researchers are poised to benefit from including EGA in their arsenal of methods to assess latent dimensionality.

**References**


exploratory factor analysis: Four recommendations for getting the most from your analysis. *Practical assessment, research & evaluation, 10*(7), 1-9.


Raiche, G. (2010). *An r package for parallel analysis and non
graphical solutions to the cattell scree test. Retrieved from http://CRAN.R-project.org/package=nFactors


