Expert Systems With Applications 238 (2024) 121779

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Generalized network-based dimensionality analysis

Zsolt T. Kosztyán [a,*], Attila I. Katona [a], Marcell T. Kurbucz [b,c], Zoltán Lantos [d]

[a] Department of Quantitative Methods, University of Pannonia, Egyetem str. 10, Veszprém, H-8200, Hungary
[b] Department of Computational Sciences, Wigner Research Centre for Physics, 29-33 Konkoly Thege Miklós Street, H-1121 Budapest, Hungary
[c] Institute of Data Analytics and Information Systems, Corvinus University of Budapest, Fővám Square 8, H-1093 Budapest, Hungary
[d] Department of Virtual Health Guide Methodology, Faculty of Health Sciences, Semmelweis University, 17 Vas Street, H-1088 Budapest, Hungary

ARTICLE INFO

Keywords:
Dimensionality reduction
Nonparametric
Network science
Modularity
Similarity graphs

ABSTRACT

Network analysis opens new horizons for data analysis methods, as the results of ever-developing network science can be integrated into classical data analysis techniques. This paper presents the generalized version of network-based dimensionality reduction and analysis (NDA). The main contributions of this paper are as follows: (1) The proposed generalized dimensionality reduction and analysis (GNDA) method already handles low-dimensional high-sample-size (LDHSS) and high-dimensional and low-sample-size (HDLSS) at the same time. In addition, compared with existing methods, we show that only the proposed GNDA method adequately estimates the number of latent variables (LVs). (2) The proposed GNDA already considers any symmetric and nonsymmetric similarity functions between indicators (i.e., variables or observations) to specify LVs. (3) The proposed prefiltering and resolution parameters provide the hierarchical version of GNDA to check the robustness of LVs. The proposed GNDA method is compared with traditional dimensionality reduction methods on various simulated and real-world datasets.

1. Introduction

In recent years, handling high-dimensional data that contain tens of thousands of variables has become an increasingly frequent and important problem in many fields of modern scientific research (see, e.g., Li, Li, Lian, & Tong, 2017; Stippinger et al., 2023). However, the disadvantages of working in such a high-dimensional space are the increased prediction error, difficult interpretability, and high computational costs (Gao, Song, Liu, Shao, Liu, & Shao, 2017; Migenda, Möller, & Schenck, 2021). To avoid these problems, or in other words, to alleviate the "curse of dimensionality" (Bellman, 1957), high-dimensional data are typically transformed into a lower-dimensional representation. In practice, this challenging task is mainly performed by traditional – typically linear – methods such as Principal Component Analysis (PCA) (Abdi & Williams, 2010; Aversano, Li, Gicquel, & Parente, 2018; Jolliffe, 2002; Nakayama, Yata, & Aoshima, 2021) and Principal Factor Analysis (PFA) (Ali, Ahmed, Ferzund, Mehmood, & Rehman, 2017; Khosla, 2004) or by a neural-network-based method such as Variational Autoencoder (VAE) (Mahmud & Fu, 2019; Mahmud, Fu, Huang, & Masud, 2018; Mahmud, Huang, Fu, Ruby, & Wu, 2021), which can handle nonlinear relationships as well.

Dimension reduction is even more challenging if the number of observations is less than the number of features. This phenomenon is often referred to as the High-Dimension Low-Sample-Size (HDLSS) problem. Traditional methods such as PCA and PFA cannot be applied effectively in this situation (Mahmud & Fu, 2019). However, approaches have been developed to handle the dimension reduction problem under the HDLSS dataset, such as low-rank tensor network decompositions (Cichocki et al., 2016, 2017), deep generative models (Mahmud et al., 2021), ensemble learning methods (Dettling & Bühlmann, 2003), or alternatively, Social Network Analysis (SNA)-based methods, such as Network-based Dimensionality Reduction and Analysis (NDA) (Kosztyán, Kurbucz, & Katona, 2022), can also be applied for dimensionality identification and reduction.[1] To our knowledge, no dimensionality reduction method would perform equally well in both the Low-Dimension High-Sample-Size (LDHSS) and HDLSS datasets. In addition, in existing dimensionality reduction methods, one of the most crucial steps is estimating the number of latent variables (LVs).

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: kosztyan.zsolt@gtk.uni-pannon.hu (Z.T. Kosztyán), katona.attila@gtk.uni-pannon.hu (A.I. Katona), kurbucz.marcell@wigner.hu (M.T. Kurbucz), lantos.zoltan@semmelweis-univ.hu (Z. Lantos).

[1] The preliminary version of the NDA was introduced in Kurbucz, Katona, Lantos, and Kosztyán (2021) and was applied on the joint dataset of COVID-19 reports and the indicators of World Bank Group's TCdata360 and GovData360 platforms (Kurbucz, 2020).

https://doi.org/10.1016/j.eswa.2023.121779
Received 1 February 2023; Received in revised form 20 September 2023; Accepted 20 September 2023
Available online 25 September 2023
0957-4174/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Most existing dimensionality reduction methods employ a symmetric measure, such as the correlation between variables, for grouping variables to specify latent variables. Therefore, they are not working on any asymmetric similarities, such as semipartial correlations, which determine whether indirect effects between variables should be filtered (Pop, Ciulca, et al., 2013). To fill this gap, in this paper, a generalized version of NDA, called Generalized Network-based Dimensionality Analysis (GNDA), is presented and compared with traditional dimensionality reduction methods on various simulated and real-life datasets. First, the proposed method calculates the similarity network of the indicators (i.e., either variables or observations) and then groups them by using modularity-based community detection. By combining the eigenvector centrality (EVC) of the indicators, a LV is specified for each module. Finally, variable selection is optimally performed based on the relationship of variables to their LVs. According to the results, the proposed method can be effectively applied to both LDHSS and HDLSS data structures; meanwhile, it extends the original NDA with a hierarchical version of dimensionality reduction; it generalizes the applicable similarity functions between indicators; and identifies the proper number of LVs.

The main contributions of the paper are listed below:

- The paper proposes a novel method called GNDA that can handle different types of datasets, such as LDHSS and HDLSS, extends similarity measures, and estimates the number of LVs accurately.
- The paper introduces a prefiltering and resolution parameter that can split communities and provide a hierarchical version of GNDA, which offers subcategories of indicators and a robustness check for LVs.
- The paper compares GNDA with several existing dimensionality reduction methods in simulated and real-life databases and shows that GNDA produces the clearest set of indicators.

The remainder of this paper is organized as follows. Section 2 presents the background of the literature. Section 3 shows the employed methods. Section 4 presents the simulated and real-world datasets and methods employed in this study. Sections 5 and 6 compare and discuss the results of GNDA with traditional dimensionality reduction methods. Section 7 provides a summary and conclusions. Finally, future research directions are suggested in Section 8.

2. Background

Dimensionality reduction techniques play a crucial role in transforming high-dimensional data into a more manageable form while retaining essential information. Among these techniques, both Explanatory Factor Analysis (EFA) (Fabrigar & Wegener, 2011) and PCA stand out as widely used methods for this purpose, aiming to approximate the underlying covariance structure of the data.

EFA and PCA both seek to capture LVs, which succinctly represents the common variance shared among a set of original variables. The primary focus of EFA lies in determining whether the data conform to a predefined structure, making it a powerful tool for uncovering the underlying factors influencing the observed variables. On the other hand, PCA condenses the original variables into a smaller number of components, enabling effective dimensionality reduction and simplification of complex data.

A variation of EFA is Common Factor Analysis (CFA), also known as PFA (Kim, 2008). PFA strikes a balance between Factor Analysis (FA) and PCA, aiming to identify a minimal set of factors that account for the shared variance among variables. In PFA, the relationship between factors and observed variables is expressed through the equation $\mathbf{Z} = \mathbf{FL} + \mathbf{U}$, where $\mathbf{Z}$ is the original data matrix, $\mathbf{F}$ is the factor score matrix, $\mathbf{L}$ is the factor loading matrix, and $\mathbf{U}$ captures unique variances. Unlike PCA, which emphasizes maximizing common variance, PFA acknowledges the presence of unique variances and focuses on explaining the correlation between variables.

Estimating the appropriate number of latent variables is a pivotal step in both PCA and PFA. Various methods have been proposed for this purpose, each with its own heuristics and rules. Commonly used criteria include Kaiser's Rule (K1), Minimum Average Partial (MAP) correlation (Velicer, 1976), MAP2000 (Velicer, Eaton, & Fava, 2000), and Minimal cumulative variance explained (MCVE) (Hair, William, Barry, & Rolph, 2020), among others. It is recommended to employ multiple methods to determine a suitable number of LVs, given their tendency to suggest different solutions.

While traditional methods such as EFA and PCA have been foundational in dimensionality reduction, recent advancements have introduced more versatile techniques, such as neural networks. These newer approaches offer unique advantages, addressing limitations posed by linear assumptions. The t-distributed stochastic neighbor embedding (t-SNE) leverages a nonlinear approach to mapping high-dimensional data into a lower-dimensional space (Liu et al., 2021), emphasizing the preservation of local data relationships. As nonlinear extensions of PCA, Kernel Principal Component Analysis (KPCA) (Schölkopf, Smola, & Müller, 1997, 1998) have been proposed but may not perform as well on LDHSS datasets. Sparse Principal Component Analysis (SPCA) enforces sparsity constraints on principal components (Zhang, d'Aspremont, & El Ghaoui, 2012), focusing on the most informative variables while ignoring less relevant ones. Non-Negative Matrix Factorization (NNMF) factorizes matrices into lower-rank nonnegative components, offering an alternative perspective on dimensionality reduction (Wang & Zhang, 2012).

Feature representation and dimensionality reduction are outstandingly important in machine learning problems such as image classification. Zhang et al. (2017) proposed a Discriminative Elastic-Net Regularized Linear Regression (DENLR) model that uses a slack formulation of regression targets to provide a more feasible regression scheme by enlarging the margins between classes. Later, Zhang et al. (2017) developed a new Marginally Structured Representation Learning (MSRL) framework that improves the reliability of the regression model by utilizing the latent explanatory factors from the data and uncovering the latent correlation across the features. In machine learning approaches, the representation of the training and testing set can also be enhanced by the Block-Diagonal Low-Rank Representation (BDLRR) method, which enables the elimination of correlation between different classes and improves the accuracy of classification. Although these methods have proven to be very effective in image classification, they focus on supervised or semisupervised problems, and the extension to unsupervised problems has not yet been provided.

While these recent methods present promising alternatives, they may lack robust mechanisms for determining the optimal number of latent variables. Moreover, they might not fully accommodate the complexities of both the LDHSS and HDLSS datasets. Additionally, some techniques, such as VAE and NNMF, may require users to specify the number of dimensions or factors.

Introducing the proposed GNDA, which uniquely fulfills multiple requirements: (1) determination of suitable latent variable count, (2) applicability to diverse datasets, (3) accommodation of symmetric and nonsymmetric similarities, and (4) incorporation of an Automated Feature Selection (AFS) procedure. By addressing these aspects, GNDA presents a comprehensive and flexible solution for dimensional reduction analysis.

Given the ubiquity of PCA and PFA as standard approaches for dimensionality reduction and their compatibility with various data types, it is prudent to benchmark the proposed GNDA against these methods. This comparison is justified by the extensive body of research supporting PCA and PFA in different scenarios, including their performance with LDHSS and HDLSS datasets (Jung & Marron, 2009). Moreover, the existence of AFS procedures for PCA and PFA (Abonyi, Czvetkó, Kosztyán, & Héberger, 2022) and their successful application in artificial datasets (Van Der Maaten, Postma, & Van den Herik, 2009) further underscores their relevance for comparison.

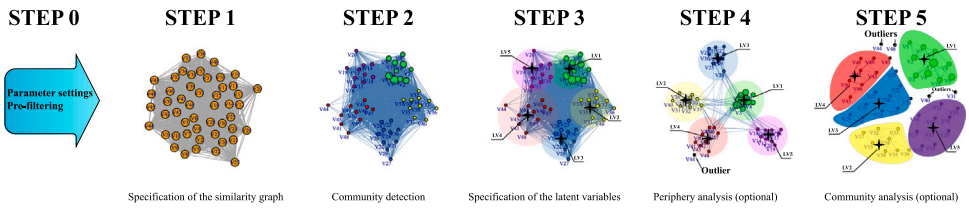


Fig. 1. The main steps of the proposed GNDA algorithm.

3. Methods

In this section, GNDA is presented in detail and then compared with two traditional dimensional reduction methods, PCA and PFA. Since these two methods are widely applied in statistical analyses, in this section, we focus only on their main features.

The proposed GNDA has five steps (see Fig. 1). The last two steps are optional, and these steps support feature selection. After specifying hyper and prefiltering parameters (step 0), the similarity graph of variables is specified (step 1) and communities of variables are determined (step 2). In step 3, LV is specified in each community based on the EVC of the standardized variable in the similarity graph. In the last steps, outliers are detected and dropped from the communities.

In this paper, we generalize the original method and therefore propose several extensions. We extend NDA to make a hierarchical decision forest of latent variables. GNDA generalizes the phase of identifying modules of variables to consider directed relationships. This extension allows specifying LVs on partial and semipartial correlation or regression networks.

Step 1. *Specification of the similarity graph.* Let $G(\mathcal{N},\mathcal{A},\mathcal{W})$ be a directed/unidirectional weighted graph of the similarity graph, where $\mathcal{N}$ is the node set, $\mathcal{A}$ is the arc set, and $\mathcal{W}$ is the weight set. Each node $i\in\mathcal{N}$ represents a variable i, $i=1,2,\dots,n$. The weight of an arc $a_{i,j}\in\mathcal{A}$ is a positive similarity function between two variables, such as the square correlation $r_{i,j}=w_{i,j}\in\mathcal{W}$. A minimal similarity value $r_{\min}$ can be specified to make the graph sparse and to ignore low correlations. Loops ($a_{i,i}\notin\mathcal{A}$) are neglected. The original NDA implemented four kinds of correlation: Pearson, Spearman, Kendall, and distance correlation, while GNDA extended them to partial and semipartial versions and handled both symmetric and asymmetric distance matrices.

Pearson correlation is suitable for interval variables and detects linear relationships, while Spearman's and Kendall's correlations detect monotonic relationships and can be used for ordinal variables. Distance correlation has the advantage that it is zero if and only if the variables are independent (Székely & Rizzo, 2013), but it is computationally expensive and limited in high-dimensional datasets.

PCA, PFA, and GNDA can use correlation matrices instead of indicators; therefore, any symmetric distance function can be used. GNDA also handles nonsymmetric similarities/distances, such as partial correlations, regressions, and other structural equation models. In this case, we use the term *similarity graph* instead of correlation graph. Nonsymmetric similarities provide a directed graph; however, all steps of the original NDA can be extended to handle it.

The proposed GNDA allows the specification of minimal similarity weights as a threshold. The increase in the minimal similarity value provides a sparser graph and more communities in the second step; hence, more latent variables are obtained. The set of latent variables can be organized into a dendrogram of latent variables.

Step 2: *Community detection.* Modularity-based community detection algorithms minimize Eq. (1).

$$M=\frac{1}{2L}\sum_{i,j}\left(r_{i,j}-\gamma\hat{r}_{i,j}\right)\delta(C_i,C_j), \tag{1}$$

where M is the modularity value; $r_{i,j}$ is the edge weight between node i and node j; $\hat{r}_{i,j}$ is the expected weight based on the null model of Newman (2006); L is the total weight in the network; γ is a constant (default 1); and δ is 1 if node i and node j belong to the same community and 0 otherwise. For directed similarity graphs, Eq. (2) must be minimized.

$$M=\frac{1}{L}\sum_{i,j}\left(r_{i,j}-\gamma\hat{r}_{i,j}\right)\delta(C_i,C_j), \tag{2}$$

The result of community detection is a partition of the graph. Isolated nodes and low-correlated variables are grouped in small communities; therefore, the minimal community size ($n^c_{\min}$) is specified. The result of modularity-based community detection is N modules, which are disjoint subgraphs of the graph. Formally, $C_1,C_2,\dots,C_N\in G, C_I\cap C_J=\emptyset, I,J\in 1,\dots,N, \cup_{I:=1}^{N}C_I\subseteq G$, and $|\mathcal{N}_I|\geq n^c_{\min}$ for all $C_I(\mathcal{N}_I,\mathcal{A}_I,\mathcal{W}_I)\in G$.

The γ parameter controls the resolution of community detection. Modularity optimization with the null model $\hat{r}_{i,j}$ has a resolution limit, which means it fails to identify small communities in large networks and communities with less than $(\sqrt{L/2}-1)$ internal links (Fortunato & Barthélemy, 2007). Reichardt and Bornholdt (RB) generalized the modularity function by introducing an adjustable γ parameter (Reichardt & Bornholdt, 2004, 2006) to handle this problem. The identified communities depend on the null model and the resolution parameter (γ). The null model of Newman (2006) is based on a configuration model, where an arc in a null model is calculated by the product of incoming and outgoing arcs divided by all links. The adjacency matrix of a null model is a contingency table of conditional independence. Therefore, modules identify communities where the connections (i.e., similarities) between nodes (i.e., variables) are denser than expected. Moreover, increasing the resolution coefficient (γ), such as increasing the threshold of minimal similarities, makes a sparse graph and breaks down the communities. In this way, a hierarchy of communities can be provided, which specifies a dendrogram. Thus, all the employed Louvain (Blondel, Guillaume, Lambiotte, & Lefebvre, 2008) and Leiden (Traag, Waltman, & van Eck, 2019) community detection algorithms provide a disjoint set of variables (i.e., modules).

Step 3: *Specification of the latent variables.* LV_I is specified within module I as a linear combination of the EVC value and the standardized variable v_i (see Eq. (3)).

$$LV_I=\frac{\sum_{i\in C_I}c_i z_i}{\sum_{i\in C_I}c_i}, \tag{3}$$

where LV_I is the LV for module C_I; c_i is the EVC value of variable v_i and $z_i=(v_i-\mu_{v_i})/\sigma_{v_i}$ is the standardized variable of variable v_i.

EVC is preferred because of its beneficial properties, such as anonymity,[2] symmetry,[3] positive homogeneity,[4] and robustness.[5] In addition, in directed and undirected graphs, EVC can be calculated. Let $A=(a_{k,l})$ be the adjacency matrix. The EVC, c_k, and the score of vertex k can be defined as:

$$c_k=\frac{1}{\lambda}\sum_{l\in N(k)}c_l=\frac{1}{\lambda}\sum_{l\in G}a_{k,l}c_l, \tag{4}$$

[2] The scores of nodes are unaffected by how they are labeled.
[3] Symmetric nodes receive the same score.
[4] This means homogeneity for all positive values.
[5] This indicates invariance after adding an average node.

where $N(k)$ is a set of neighbors of k, and λ is a constant.

Both the EVC value and the eigenvalue in PCA are based on eigenvalues and eigenvectors, which are used to analyze complex systems. An eigenvector of a matrix is a nonzero vector that changes by a scalar factor when multiplied by the matrix. The scalar factor is the eigenvalue. In eigenvector centrality, the eigenvector measures the importance of a node in a network. It is obtained by finding the eigenvector corresponding to the largest eigenvalue of the network's adjacency matrix. The larger the eigenvector centrality value of a node is, the more important that node is in the network.

In PCA, the eigenvalue measures the strength of the relationship between variables in a dataset. It is obtained by representing the dataset as a matrix, where the rows are data points and the columns are variables. The larger the eigenvalue is, the stronger the relationship between the variables in the dataset. The similarity between the eigenvector centrality value and the eigenvalue in PCA is that both are based on eigenvalues and eigenvectors, and both measure the importance or strength of a relationship in a complex system. However, they are used and measured in different contexts.

Step 4: Periphery analysis (optional). AFSs suggest two phases of variable selection. First, variables with EVC values below a threshold $c_{\min}$ are dropped, and LVs are recalculated without peripheral nodes. These nodes are at the edge of the network, which results in low centrality values. However, this does not imply that these variables are unimportant but only that their influence on the latent variable is low. Therefore, similar to omitting variables with low communality in PCA, we also omit those variables that do not fit the latent variable or have low contributions to it. One of the advantages of EVC is that it works for both directed and undirected graphs. Moreover, the interpretation of EVC is close to the interpretation of factor loadings in PCA or PFA, so this value is an ideal weight for calculating latent variables. A higher centrality value indicates a higher weight for the indicator. Importantly, this step does not depend on any correlation calculation. The removal of peripheral nodes (*i.e.,* , indicators) of modules of the similarity graph can be interpreted similarly as dropping indicators with low communalities in a squared correlation graph.

Step 5: Communality analysis (optional). Step 5 has two substeps and requires the similarities between the indicators to be correlations. The communality value for indicator i is the highest square correlation between i and any LV. Iteratively, the indicator with the lowest communality value below a threshold ($h_{\min}$) is dropped, and all LVs are recalculated. Then, all square correlations between each indicator and each LV are analyzed. If the difference between the two highest square correlations for indicator i is below a threshold ($C_{\min}$), i is a *common indicator*. The common indicators with the lowest communality value are dropped one by one in an iterative manner, and LVs are recalculated. This step is optional and should be skipped if the correlation between indicators cannot be calculated or is meaningless for this analysis. This step can be used in both PCA and PFA (see an application for AFS in Abonyi et al., 2022).

Step 0: Prefiltering (optional). The number of latent variables is not affected by the thresholds of the minimal EVC value ($c_{\min}$), minimal communality value ($h_{\min}$), or common communality value ($C_{\min}$). However, the thresholds of the minimal correlation (or similarity) value ($r_{\min}$) and resolution value of the null model (γ) can specify more communities and latent variables.

Increasing ($r_{\min}$) drops connections or ignores similarities between nodes below this threshold. The graph becomes sparse and splits into more components. Therefore, communities are also split, and more latent variables can be specified.

Similarly, the increase in γ detects smaller communities and splits the original communities into smaller ones (Reichardt & Bornholdt, 2004, 2006). These two parameters are called *prefiltering* parameters. The increasing value of these (hyper)parameters provides a dendrogram of GNDA and provides a hierarchical version of GNDA, revealing the *hierarchical structure of the latent variables*.

4. Experimentation

We tested the proposed method on two datasets. The first contains simulated samples to test dimensionality reduction methods, such as PCA, PFA, and GNDA, while the second is a real-world dataset called Communities and Crime (Redmond & Baveja, 2002) and is freely available at https://archive.ics.uci.edu/ml/datasets/communities+and+crime(accessed: 13 April 2023). This dataset includes variables relevant to per capita violent crime rates in different communities of the United States.

Generating simulation dataset. Our goal in generating the simulation database was to generate matrices where the number of independent components can be known in advance. Therefore, given the number of observations (n), we can obtain b independent orthogonal vectors by dividing the n-element vector into b blocks. Then, choosing a block, we write ones in that block and zeros in the others. These will be our basis vectors (see Eq. (5)).

$$\begin{aligned}\mathbf{e}_1 &= (\overbrace{1,1,\dots,1}^{\lceil n/b\rceil},0,0,\dots,0)^T\\ &\dots\\ \mathbf{e}_b &= (0,0,\dots,0,\underbrace{1,1,\dots,1}_{\lceil n/b\rceil})^T\end{aligned} \tag{5}$$

These vectors are orthogonal. All dimensionality reduction methods must identify these vectors as independent vectors. We obtain a block matrix if each basis vector is copied $\lceil m/b\rceil$ times, where m is the number of variables.

The proposed simulation dataset contains generated block matrices. At the generated block matrices, the number of rows (n) (i.e., the number of observations), the number of columns (m) (i.e., the number of variables), and the number of blocks (b) (i.e., the number of factors) can be specified. Eq. (6) shows an example for 6 by 5, 0–1 block matrices.

$$\mathbf{B}_2^{(6\times5)}=\begin{pmatrix}\overbrace{1\,1\,1}^{\mathbf{e}_1}&\overbrace{0\,0}^{\mathbf{e}_2}\\1\,1\,1&0\,0\\1\,1\,1&0\,0\\0\,0\,0&1\,1\\0\,0\,0&1\,1\\0\,0\,0&1\,1\end{pmatrix},\quad \mathbf{B}_3^{(6\times5)}=\begin{pmatrix}\overbrace{1\,1}^{\mathbf{e}_1}&\overbrace{0\,0}^{\mathbf{e}_2}&\overbrace{0}^{\mathbf{e}_3}\\1\,1&0\,0&0\\1\,1&0\,0&0\\0\,0&1\,1&0\\0\,0&1\,1&0\\0\,0&0\,0&1\end{pmatrix}. \tag{6}$$

Suppose all values are equal to 1 within all b blocks, but the remaining values are 0. In that case, the block matrix specifies b uncorrelated factors since both the vectors of columns and rows provide orthogonal vectors. Note that the rows and columns of the block matrices can be interchanged during the analysis. Within a block, the row and column vectors are the same, while between blocks, the rows and columns are orthogonal. Thus, for $n \gg m$, we obtain a binary LDHSS dataset, while its transpose becomes a binary HDLSS dataset.

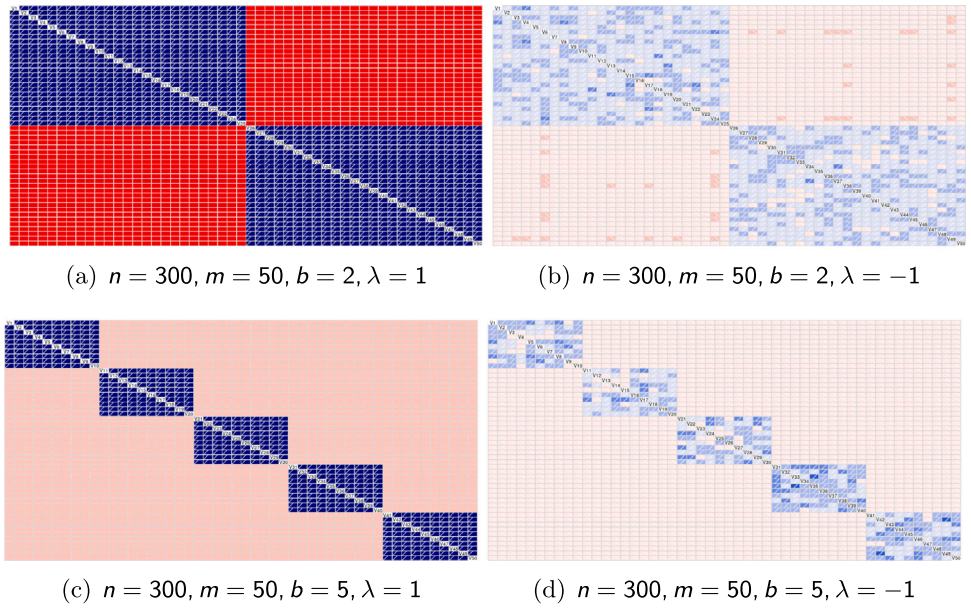
Eq. (7) is used to generate block matrices for testing dimensional reduction methods.

$$\mathbf{M}_{b,\lambda}^{(n\times m)}=\mathbf{B}_b^{(n\times m)}-\mathbf{B}_b^{(n\times m)}\circ\mathbf{U}^{(n\times m)}/\exp(\lambda), \tag{7}$$

where $\mathbf{M}$ is the result block matrix, $\mathbf{B}$ is an n by m 0–1 block matrix, where the number of blocks is b; $\mathbf{U}^{(n\times m)}$ is the n by m is a random matrix, where values follow a $U(0,1)$ uniform distribution; $\circ$ represents the elementwise multiplication of matrices; and $\lambda \in \mathbb{R}$ is the so-called exponent of *exponential smoothing*. If $\lambda \to \infty \Rightarrow \mathbf{M}_{b,\lambda}^{(n\times m)} \to \mathbf{B}_b^{(n\times m)}$. The noise follows the uniform distribution scaled by $1/\exp(\lambda)$.

Changing λ sets the noise level. PCA and PFA are very sensitive to outliers. However, there are no extreme outliers due to employing a uniform distribution. Fig. 2 shows the Pearson's correlogram[6] of the

[6] A correlogram is a graphical representation of a correlation matrix of the original data. Therefore, the number of rows and columns equals the number of variables.



(a) $n = 300, m = 50, b = 2, \lambda = 1$ (b) $n = 300, m = 50, b = 2, \lambda = -1$

(c) $n = 300, m = 50, b = 5, \lambda = 1$ (d) $n = 300, m = 50, b = 5, \lambda = -1$

Fig. 2. Correlation matrix of simulated samples, where the number of rows/observations (n) is 300, the number of columns/variables is (m) 50, the numbers of blocks (b) are 2 and 5, and exponential smoothing values (λ) are 1 and −1.

generated dataset ($\mathbf{M}_{b,\lambda}^{(n\times m)}$), where the number of rows is $n = 300$, and the number of columns is $m = 50$. The numbers of blocks are $b = 2$ and $b = 5$, and the exponents of smoothing are $\lambda = 1$ and $\lambda = -1$. The correlations are between [−1,1]. Bluish cells indicate positive correlations, while reddish cells represent negative correlations. Darker cells indicate higher absolute correlations. A correlation matrix of a block matrix is also a block matrix. Nevertheless, in this case, the blocks are square matrices because the number of rows and the number of columns are equal to each other.

Fig. 2 shows that the decrease in the exponent λ decreases the correlation between variables and increases the noise. The correlogram shows that Fig. 2(a–b) indicates two groups, while Fig. 2(c–d) indicates five groups of variables. A dimensional reduction method should find these groups of variables. The goodness of the dimension reduction methods can be compared in terms of how many LVs (which represent a group of variables) are found and whether adequate original variables belong to the adequate LV.

Employing a real-life dataset. In the case of the Communities and Crime dataset, the per capita number of violent crimes was considered a dependent variable (y). This variable was separated from the dataset. The remaining (independent) variables ($m = 124$) were grouped into LVs by the PCA, PFA, and GNDA methods. These variables characterize the community with population and law enforcement data, such as the per capita number of police officers or the percentage of officers assigned to drug units.

The per capita violent crime variable was calculated using population, and the sum of crime variables considered violent crimes in the United States: murder, rape, robbery, and assault. There was apparently some controversy in some states concerning the counting of rapes. These resulted in missing values for rape, which resulted in incorrect values for per capita violent crime. These cities are not included in the dataset. Many of these omitted communities were in the midwestern United States.

All numeric data were normalized into the decimal range of 0.00 to 1.00. Attributes retained their distribution and skew. All cities were geo-coded to visualize the spatial distribution of the score values.

Although GNDA proposed several extensions and generalizations of the original NDA, some of these new features cannot be compared with traditional dimensionality reduction methods, such as PCA and PFA; therefore, Section 5.1 focuses on the applications in both the HDLSS and LDHSS cases. In both cases, one of the most crucial steps of the analysis is to estimate the number of LVs. To be able to compare which method estimates the adequate number of LVs, LDHSS and HDLSS datasets, i.e., block matrices, were generated. In these cases, the number of LVs was known in advance. In this way, we can accomplish two tasks: (1) determine which method estimates the adequate number of LVs in the case of different noises and (2) assess whether the indicators were assigned to appropriate LVs by the methods or not. **This result shows that only the proposed GNDA method finds adequate numbers of LVs**.

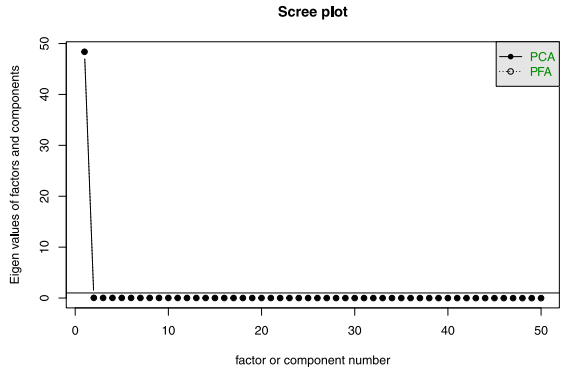
In Sections 5.2 and 5.3, the results of the proposed GNDA and those of the traditional PCA and PFA on the real-world dataset are compared. Because GNDA provides the number of latent variables and the simulated data, this estimate seemed more reliable; therefore, we consider the number of LVs recommended by GNDA as a guideline for the other methods as well. Section 5.4 introduces the opportunity for hierarchical analysis by tuning prefiltering parameters. All results are compared with the PCA and PFA methods. However, since only GNDA can handle nonsymmetric similarities, these results are shown in the appendix. The original NDA was implemented in R and has been accepted by the CRAN community (see https://cran.r-project.org/web/packages/nda/index.html, accessed: 13 April 2023). The GNDA can be downloaded from GitHub (see https://github.com/kzst/nda, accessed: 13 April 2023).

5. Results

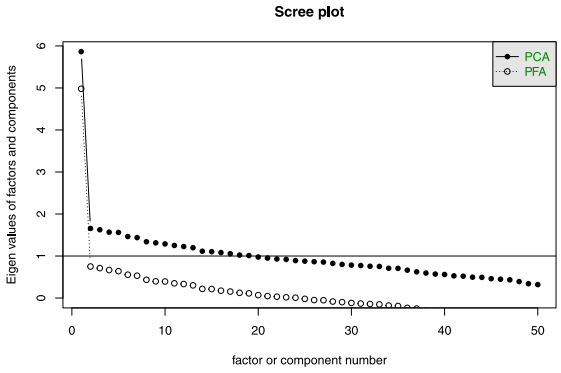
5.1. Comparing dimension reductions for simulated block matrices

In the first simulation, four block matrices, such as $\mathbf{M}_{\mathbf{b=2,\lambda=1}}^{(\mathbf{n(=300)\times m(=50)})}$, $\mathbf{M}_{\mathbf{b=2,\lambda=-1}}^{(\mathbf{n(=300)\times m(=50)})}$, $\mathbf{M}_{\mathbf{b=5,\lambda=1}}^{(\mathbf{n(=300)\times m(=50)})}$, $\mathbf{M}_{\mathbf{b=5,\lambda=-1}}^{(\mathbf{n(=300)\times m(=50)})}$, are generated (see their correlation matrices in Fig. 2). Fig. 3 shows the scree plots of PCA and PFA of the generated matrices.

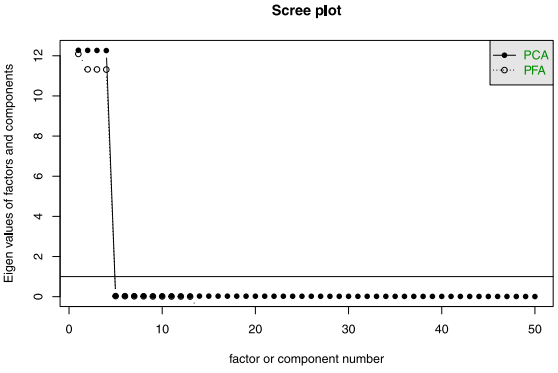
Fig. 4 shows the number of estimated LVs. The most commonly used method, Kaiser's rule, is used in the case of PCA and PFA. Fig. 4(a) shows the LDHSS case, where the number of observations ($n = 500$) is 10 times greater than the number of variables ($m = 50$), while Fig. 4(b) shows the HDLSS case, where the number of observations ($n = 50$) is ten times less than the number of variables ($m = 500$). In both cases, the number of blocks is five ($b = 5$), which must be estimated.



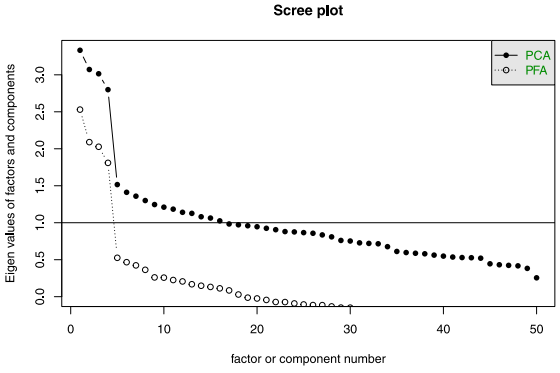
(a) $n = 300, m = 50, b = 2, \lambda = 1$



(b) $n = 300, m = 50, b = 2, \lambda = -1$

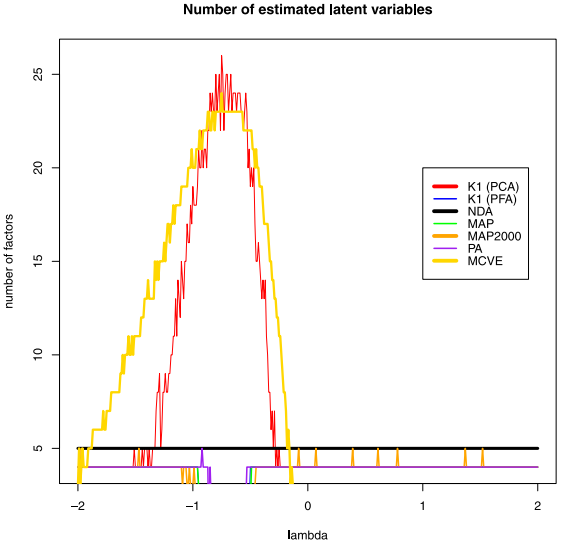


(c) $n = 300, m = 50, b = 5, \lambda = 1$

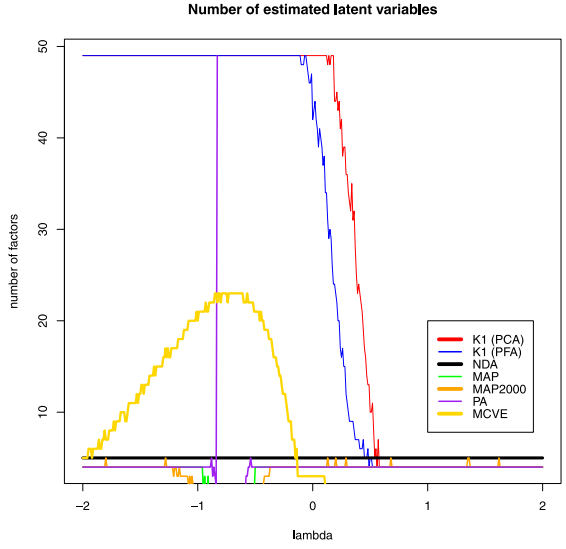


(d) $n = 300, m = 50, b = 5, \lambda = -1$

Fig. 3. Scree plots of simulated samples.



(a) $n = 500, m = 50, b = 5, \lambda \in [-2, 2]$



(b) $n = 50, m = 500, b = 5, \lambda \in [-2, 2]$

Fig. 4. Number of estimated LVs for simulated data.

Table 1
Estimated number of LVs ($n = 500, b = 5, \lambda \in \{-1, 1\}$).

Methods	λ	m				
		50	250	500	1000	5000
PCA (K1)	1	4	4(!)	4(!)	4(!)	4(!)
PFA (K1)		4(!)	4(!)	4(!)	4(!)	4(!)
GNDA		5	5	5	5	5
PCA (K1)	−1	18	92	176(!)	317(!)	499(!)
PFA (K1)		4	19	74(!)	183(!)	499(!)
GNDA		5	5	5	5	5

Notes: (!): Results with warnings.

Table 2
Estimation of the number of factors to be retained in the model.

Methods	Applied correlation methods			
	Pearson's	Spearman	Kendall	Distance
PCA (K1)	17	15	16	19
PFA (K1)	11	10	10	12
MAP	29	15	16	19
MAP(2000)	29	22	16	23
MCVE	1	1	1	1
GNDA	4	3	3	3

Note: Parallel Analysis (PA) cannot be calculated for this dataset.

Fig. 4(a) shows that in the case of PCA K1 and MCVE, between intervals $\lambda \in [-1.5, -0.3]$, the number of LVs is overestimated, while PFA K1 and other methods underestimate the number of LVs in all cases, especially in the interval between $\lambda \in [-1.0, -0.5]$. In the case of HDLSS (see Fig. 4(b)), the number of LVs is overestimated if $\lambda < 0.5$ and underestimated if $\lambda > 0.5$.

Although the number of factors (or the number of LVs) is a required parameter both for PCA and for PFA, too, which should be specified before running these methods. The scree plot can help to estimate the number of factors because the eigenvalues of the factors should be greater than 1 (K1).

Fig. 3 shows that in the case of positive, such as the $\lambda = 1$ exponent underestimate, while negative, such as $\lambda = -1$ can overestimate the number of LVs. Nevertheless, in contrast to PCA and PFA, GNDA provides the number of LVs well (see Fig. 5).

Fig. 5 shows Pearson's correlation graphs for data matrices $\mathbf{M}_{5,1}^{(300\times50)}, \mathbf{M}_{5,-1}^{(300\times50)}, \mathbf{M}_{5,1}^{(50\times300)}, \mathbf{M}_{5,-1}^{(50\times300)}$ matrices.

Fig. 5 shows that the proposed GNDA is insensitive to the low sample size (see Table 1). GNDA always finds an adequate number of factors, and all variables are associated with the right LV. Moreover, if no more than 30% of the observations were left out randomly, then after 100 runs, in the case of $\lambda = -1$, for both LDHSS and HDLSS data, the GNDA correctly identified the five latent variables in all cases, as well as the variables 99.6% in the case of LDHSS, and 98.2% in the case of HDLSS correctly assigned to an appropriate latent variable.

GNDA already work on asymmetric similarities between indicators. Fig. 6 shows that GNDA already finds an adequate number of LVs and classifies the variables well if indirect correlations are filtered by partial (see Fig. 6(a)) or semipartial correlations (see Fig. 6(b)). Since semipartial correlation provides an asymmetric correlation matrix, the correlation graph will be directed; nevertheless, the modules right back the group of variables as modules (i.e., communities).

Due to both partial and semipartial correlation filter indirect effects, Fig. 6 shows that nodes within a module are more scattered. However, in these simulations, thanks to the generated block matrices, the number of blocks (here factors) was correctly determined for all smoothing parameters (λ) and all examined row/column ratios. In the real case, filtering out indirect effects makes the network sparser and thus can increase the number of found modules.

Table 1 shows the comparison of the dimensionality reduction methods for the generated samples. The generation, the number of blocks ($b = 5$), and the number of observations ($n = 500$) are fixed, but the number of columns (variables) was a tenth ($m = 50$), half ($m = 250$), the same ($m = 500$), double ($m = 1000$), and tenfold ($m = 5000$). Two exponents are applied, such as $\lambda = 1$ and $\lambda = -1$.

The applied psych package in R (see v. 2.2.9 Revelle, 2022) can be used to implement the newest version of PCA and PFA, and we also obtained results with warnings for the high number of variables. However, both methods in all cases over- or underestimate the number of LVs. The use of both PCA and PFA results in the determinant of the smoothed correlation being zero. Chi-square of observed residuals, and the result is unreliable. GNDA always specified the number of variables unambiguously. In addition, if the number of LV for all PCA and PFA was restricted to five, the accuracy of PCA and PFA was between 75 and 85%, depending on the smoothing factor, while the accuracy of GNDA was 100% in all cases.

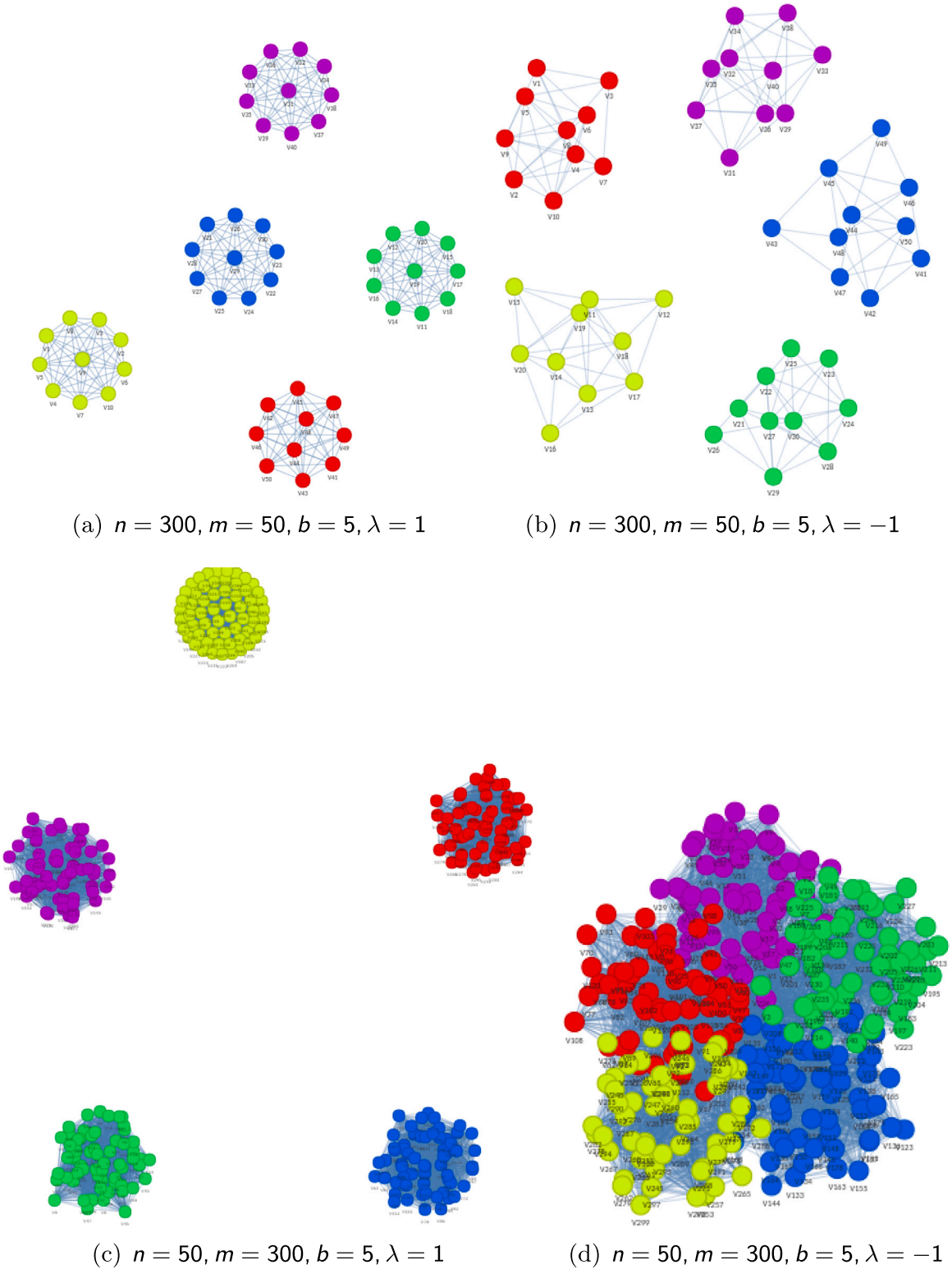
5.2. Results on the real-world dataset

Fig. 7(a) shows the scree plot of PCA and PFA, and Fig. 7(b,c) shows biplots for PCA and PFA.

Fig. 7(a) suggests that 15 to 18 components (PCA) or 10 to 12 factors (PFA) should be specified. Many factors may also be because there are more than just linear relationships between the variables. Thus, a better estimate can be obtained by calculating the possible number of variables with a monotonic (such as Spearman's and Kendall's) or general (such as distance) relationship-based correlation (see Table 2). Since GNDA indicates three LVs for Spearman's, Kendall's, and distance correlations, to compare results, three LVs are assumed for all dimension reduction methods. Because of the great number of variables, the information content of the biplots (Figs. 7(b,c)) is rather limited. Nevertheless, all Figs. 7(b,c) show that neither PCA nor PFA can separate all variables into three groups. Variables in the diagonals of biplots correlate with at least two factors/components.

Table 2 shows the number of estimated factors based on different correlation and latent variable estimation methods.

Table 2 shows that Pearson's and distance correlations provide the greatest number of factors. Since distance correlation can detect general relationships between variables, this correlation method is applied instead of the linear correlation method. Moreover, in 12% of the runs, the GNDA method also identified three latent variables in the case of Pearson correlation, while using the other correlations, the GNDA method identified three latent variables even if 30% of the observations were randomly removed. Since GNDA detected three factors when all methods were compared, the number of factors was assumed to be three. In contrast to the generated example, where we knew the correct number of LVs in advance, here we did not know how many LVs were worth examining. Different methods suggested different kinds of LVs. K1, and MAP, and MAP(2000) methods suggested more than 10 latent variables. That is why we calculated the five highest communality indicators of the first 20 variables determined by the PCA and PFA methods, as well as their possible interpretations, as shown in the appendix. However, MCVE shows that the first LV explained more than 60% of the variance, and GNDA only suggested 3 to 4 LVs. Since nonlinear correlations were employed, to handle ordinal values and nonlinear connections between variables, three LVs were assumed, and all methods were compared for three LVs. The different dimension reduction methods can only be compared if the same LV number was employed for each method. Even with the same number of LVs, their interpretation may differ depending on which indicators characterize the LVs or vice versa. Nevertheless, in the case of different LVs, the variable groups and the interpretation of the latent variables cannot correspond to each other. Therefore, we assume three LVs in each method, which was suggested by GNDA. In the main text, we only compare the results of the correlation-based methods, such as PCA and PFA, with the results given by GNDA. However, in the appendix, recent methods, such as the



(a) $n = 300, m = 50, b = 5, \lambda = 1$ (b) $n = 300, m = 50, b = 5, \lambda = -1$

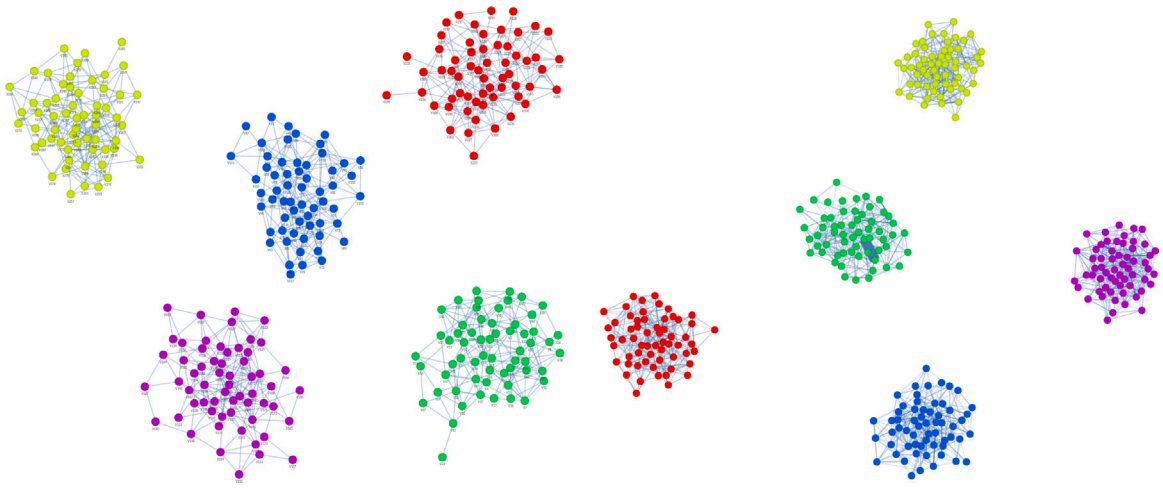
(c) $n = 50, m = 300, b = 5, \lambda = 1$ (d) $n = 50, m = 300, b = 5, \lambda = -1$

Fig. 5. Network plots for simulated samples. Figs. (a–b) provide the clustered correlation graph (network plot) for LDHSS, while Figs. (c–d) provide the HDLSS dataset.

results of NNMF, SPCA, KPCA, t-SNE, can also be found (see Table A.8). At the same time, their reduction mechanism is either significantly different from correlation-based methods (e.g., NNMF, t-SNE), and/or there is no known method that would determine the number of LV. Therefore, the three latent variables estimated by GNDA can only be considered here with a possible assumption.

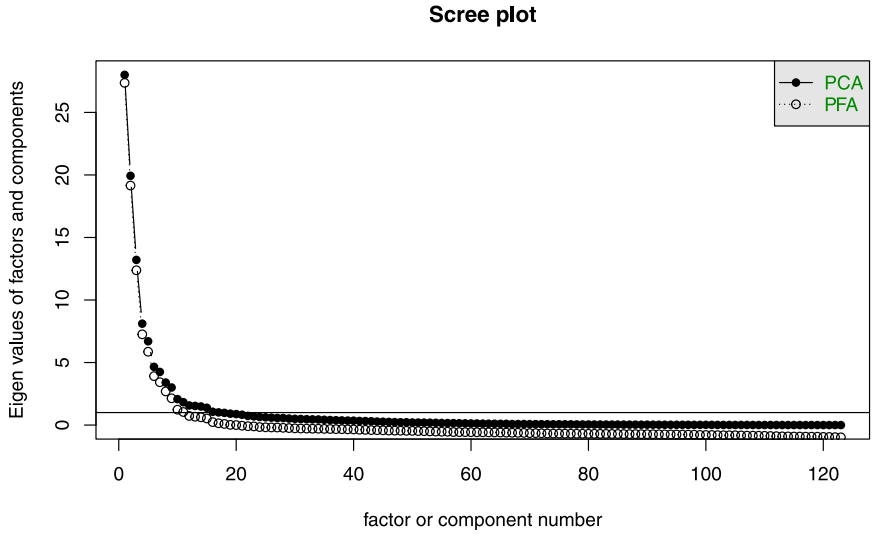
Fig. 8 shows the word cloud of the terms of the indicators. Colorful terms represent the top 10% words, and the size of the font is proportional to their frequencies.

The frequencies of terms can help to interpret LVs. Fig. 8 shows that the most frequent terms in PCA were income, household, people, and police, and in the case of PFA, they were police, income, and people.

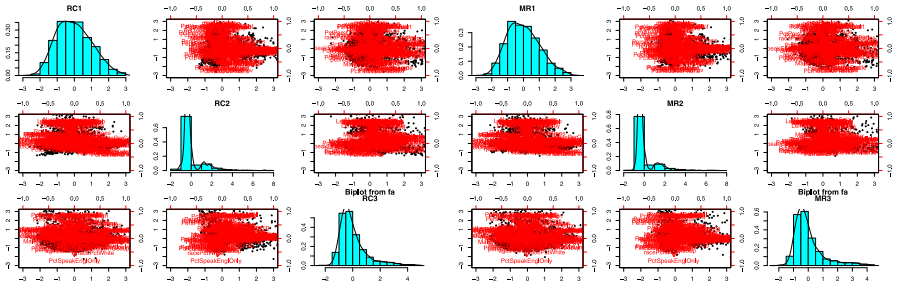


(a) Partial correlation graph (b) Semipartial correlation graph

Fig. 6. Communities on Pearson's partial correlation graph on simulated samples ($n = 300, m = 50, b = 5, \lambda = 1$).

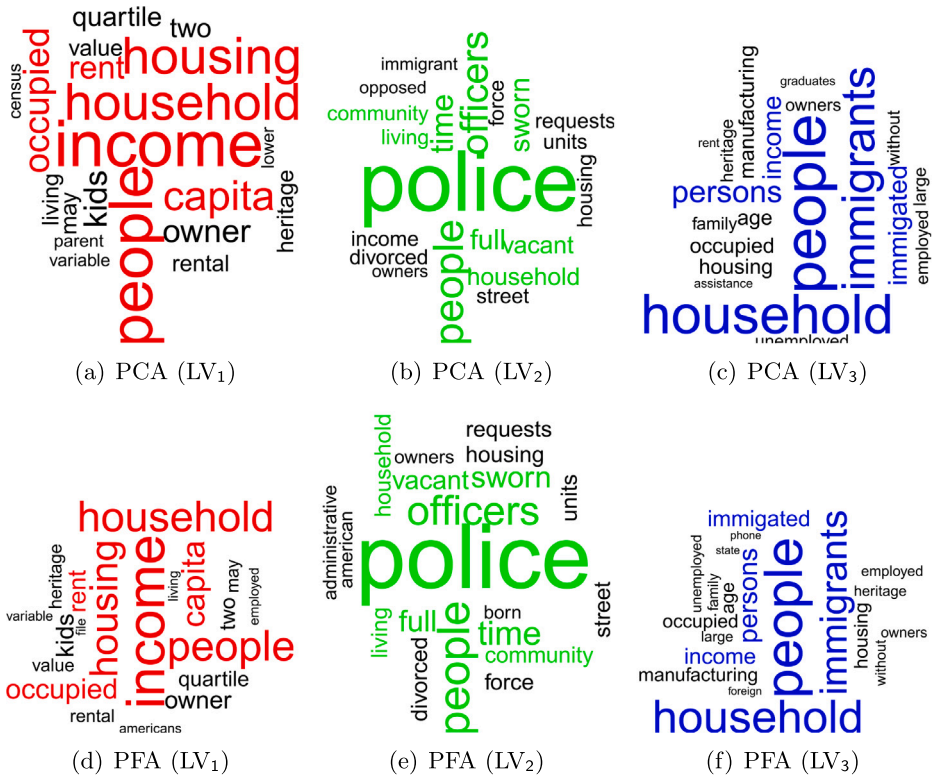


(a) Scree plot for PCA and PFA



(b) Biplot of PCA (c) Biplot of PFA

Fig. 7. Comparison of PCA and PFA on the Crime 1990 dataset.



(a) PCA (LV_1) (b) PCA (LV_2) (c) PCA (LV_3)

(d) PFA (LV_1) (e) PFA (LV_2) (f) PFA (LV_3)

Fig. 8. Word cloud of the terms of the indicators in the case of three factors.

However, these terms were involved in other groups of variables (other factors) with high frequency.

Fig. 9 shows the results of GNDA without feature selection. The applied default $cut = 0.3$ parameter is only used when plotting the variable correlation network to speed up the layout calculated by the Force Atlas algorithm. A higher value provides fewer connected graphs but faster calculation, while a lower threshold provides fewer components. At the same time, the calculation of the final layout requires more computational time.

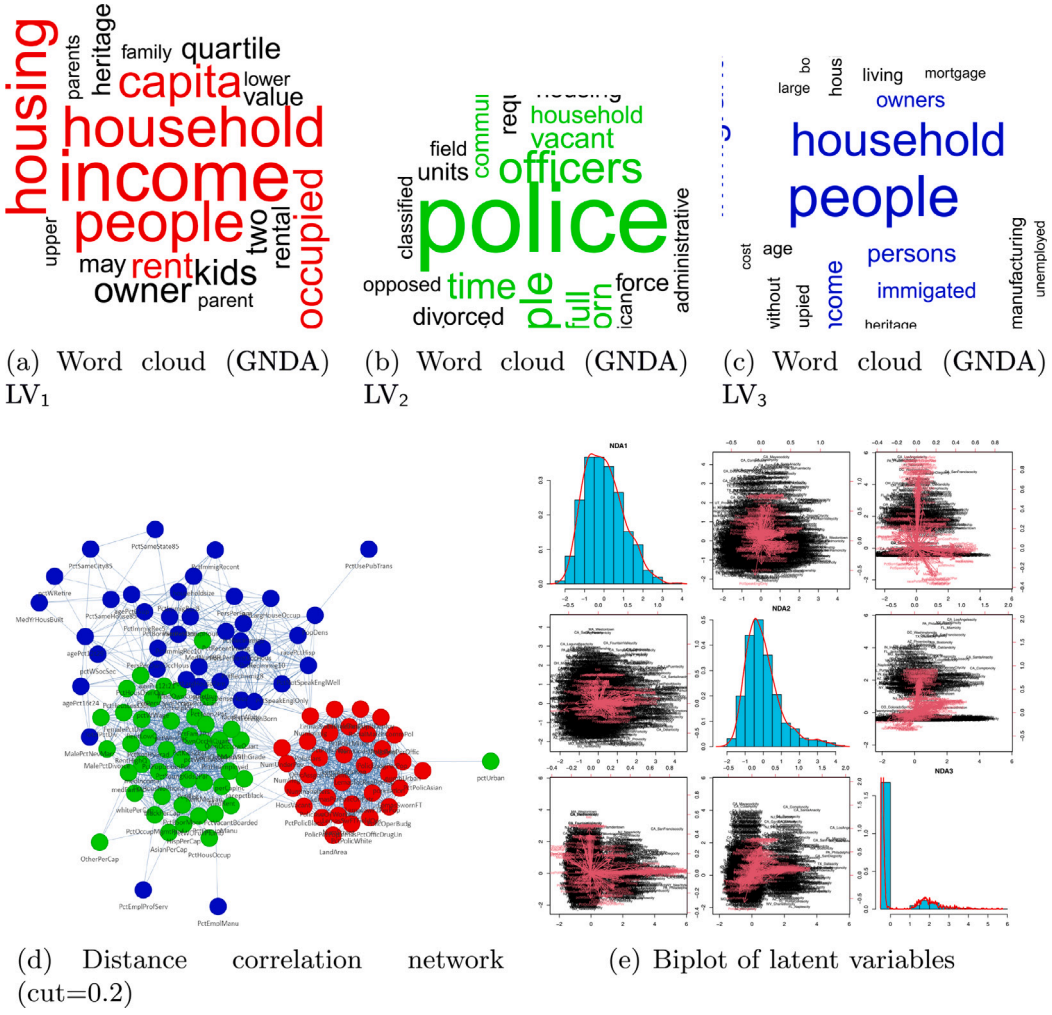
Without variable (i.e., feature) selection, the most frequent terms were income, police, and people. The correlation graph shows that LV_1 (income) and LV_2 (police presence) appeared cleaner, while the last one was mixed.

Table 3 displays the top five indicators for all explored dimension reduction methods with the greatest loading values. Based on the top five indicators (which had the greatest communality values), the three LVs were as follows: (1) income, (2) police presence, and (3) immigrants. The word cloud also detected the related term of immigration, but the people word was more frequent in the description. At the same time, since the top five indicators were related to immigration, LV_3 is referred to as immigration. The first row in every block represents Pearson's correlation between LV and the output variable (crimes per population). None of these methods claim to increase the correlation between the latent and output variables. Nevertheless, the comparison of the sign and the value of the correlations helps to validate the content of the latent variables. Significant differences and even the sign change in correlations between the given latent and the outcome variables indicate different contents.

Compared to the word clouds of indicators (see Figs. 8–11), which were based on the frequency of terms of the variable descriptions, the analysis of the top five indicators (see Table 3) provided cleaner LVs. The LVs could be interpreted as the (family/household) income, police presence, and immigrants. PCA and PFA found exactly the same top five indicators. Only Pearson's correlations between the given LV and the output variables were slightly different. The sign of the pairs of correlations between a given LV and the outcome variable was the same in all dimension reduction methods. There was no significant difference between correlation values between methods in the cases of LV_2 and LV_3, while GNDA provided a lower correlation value between LV_1 and the output variable than PCA and PFA. GNDA suggested several new indicators in addition to the groups of top five indicators, such as "population for the community" (LV_1); "percent of police officers that are Caucasian/African American", and sworn full-time police officers in field operations (LV_2); and "percent of people who have immigrated within the last 5 years", and "percent of family households that are large (6 or more)". In this way, we obtained cleaner LV_2 (police presence) and LV_3 (immigrants) than PCA or PFA provided. At the same time, except for the first indicator (population for community) of LV_1 in GNDA, all top indicators were related to income.

5.3. Employing feature selection

Feature selection is optional if the question is simply which indicators belong to which LVs. At the same time, interpretation is facilitated if those variables that are less correlated to other variables, or in other words, are located on the periphery in a correlation graph measured by EVC, are lower than a threshold ($c_{\min}$, only in GNDA) and will be dropped. Variables can also be dropped if the square correlation (communality values) between LV and the indicator (original variable) is lower than a threshold ($h_{\min}$). If this (square) correlation is equally great between a given indicator and more than one LV, then the indicator is a common indicator; therefore, it should be dropped if the difference between the communality is lower than a threshold ($C_{\min}$).



(a) Word cloud (GNDA) LV_1

(b) Word cloud (GNDA) LV_2

(c) Word cloud (GNDA) LV_3

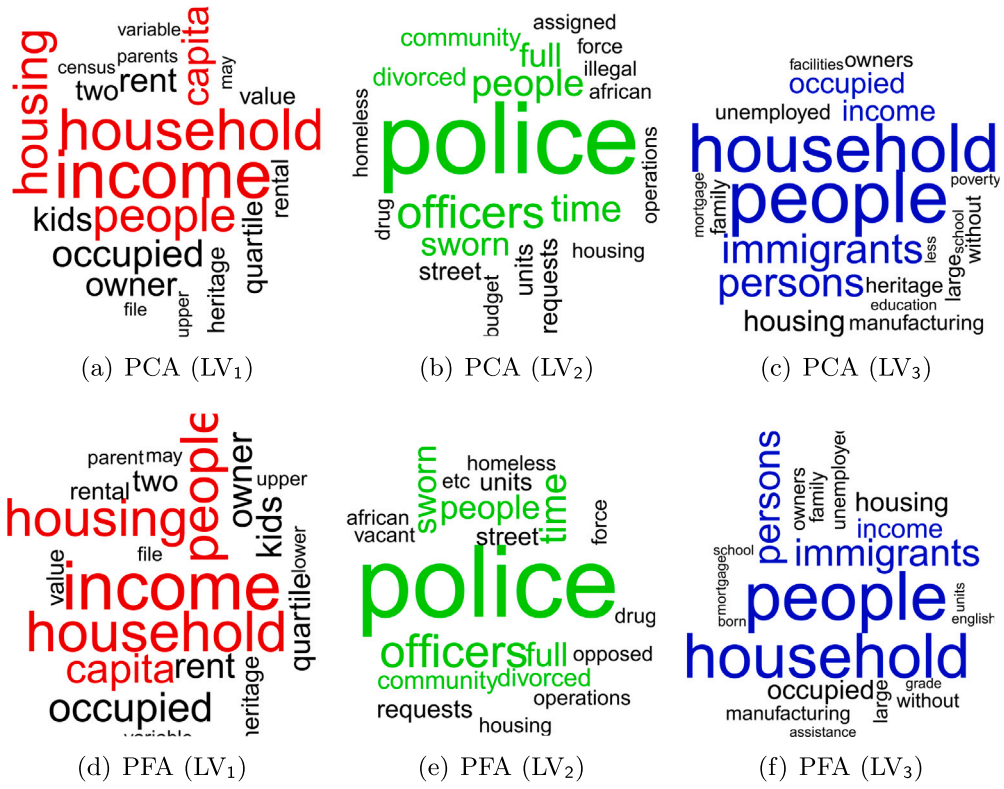
(d) Distance correlation network (cut=0.2)

(e) Biplot of latent variables

Fig. 9. Word cloud of terms of the variable descriptions (a-c), distance correlation network of indicators (d), and biplot of latent variables (e) (GNDA without feature selection).

Table 3
Top five indicators with greatest loadings.

Method			Income	Police presence	Immigrants
PCA, PFA	Correlations (PCA)		−0.4867	0.4456	0.2618
	Correlations (PFA)		−0.4835	0.4445	0.2625
	Top 5 variables:	1.	per capita income	police average overtime worked	percent of housing units with less than 3 bedrooms
		2.	percentage of households with wage	percentage of people living in areas classified as urban	percent of population who speak only English
		3.	per capita income for caucasian	mean people per household	percent of family households that are large (6 or more)
		4.	rental housing - upper quartile rent	police operating budget	percent of the population who have immigrated within the last 10 years
		5.	median gross rent	gang unit deployed	percent of the population who have immigrated within the last 8 years
GNDA	Correlations (GNDA)		−0.1538	0.4146	0.2694
	Top 5 variables:	1.	population for community	police average overtime worked	percent of the population who have immigrated within the last 8 years
		2.	percentage of households with wage or salary income	percent of police that are caucasian	percent of the population who have immigrated within the last 10 years
		3.	per capita income	gang unit deployed	percent of people who speak only English
		4.	median gross rent	sworn full-time police officers in field operations	percent of the population who have immigrated within the last 5 years
		5.	median gross rent as a percentage of household income	percent of police officers who are African American	percent of family households that are large (6 or more)



(a) PCA (LV_1) (b) PCA (LV_2) (c) PCA (LV_3)

(d) PFA (LV_1) (e) PFA (LV_2) (f) PFA (LV_3)

Fig. 10. Word cloud of the terms of the indicators after feature selection ($h_{\min} = C_{\min} = 0.1$) in the case of three factors.

Fig. 10 shows that after feature selection, the frequency of terms is slightly changed. Later, income, police presence, and people/immigrants can be identified by both PCA and PFA.

Fig. 11(a-c) shows the word cloud of terms of three LVs, where the most frequent terms were similar to those of PCA and PFA: immigrants, police, and income. The hyper parameters suggested by Kosztyán et al. (2022) were $c_{\min} = 0.065, h_{\min} = C_{\min} = 0.1$, indicating that the minimal EVC must be higher than $c_{\min} = 0.065$, and the minimal communality ($h_{\min}$) and the common communality ($C_{\min}$) values should be greater than 0.1. In Fig. 11(d), black nodes represent the dropped indicators.

Fig. 11(a) shows that the group of indicators was cleaner, and Fig. 11(d) shows that the selected indicators were closer. The increase in minimal common communality eliminated the variable in biplot diagonals (compare Figs. 9(d) and 11(d)), which further promotes the interpretation.

Table 4 displays the top five indicators for all explored dimension reduction methods with the greatest loading values. The head row indicates the interpretation of the 3 LVs, namely, (1) income, (2) police presence, and (3) immigrants. The first row in every block displays Pearson's correlation between LV and the output variable (crimes per population).

After feature selection, PCA and PFA provided similar results but not the same top five indicators. The main difference was their ranks. The signs of correlations for given LVs were the same, but PFA provided the highest absolute correlation value for LV_1 and LV_2. GNDA provided the highest correlation value for LV_3. GNDA provided more balanced correlations between the LVs and the outcome variables.

To be able to compare the results of the given interpretation of LVs (see the head of Table 4 is used for every dimension reduction method. Nevertheless, only in the case of GNDA did feature selection provide a clean set of indicators.

The negative correlation value between income and the output variable (crimes per population) indicated that cities with lower household incomes had higher levels of crime. While police presence increased most of the detected crimes in the population, the increase in the number of immigrants increased the number of crimes per population.

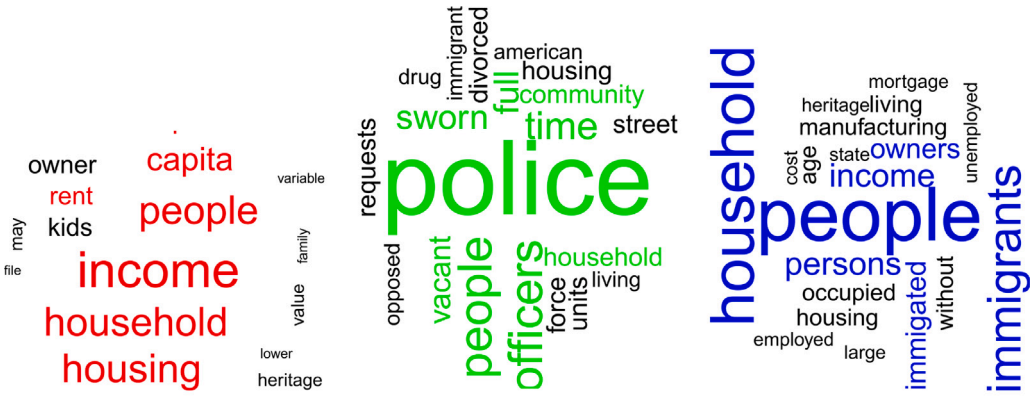
GNDA also highlights that these three LV are sufficient to characterize the set of indicators. Fig. 12 shows the spatial distribution of the scores of LVs.

The map in Fig. 12 indicates that the highest score of immigrants was near the border of the United States, and the police presence was greatest in large cities, such as Los Angeles, New York, and Washington. Table 5 shows the top five cities with the highest score values.

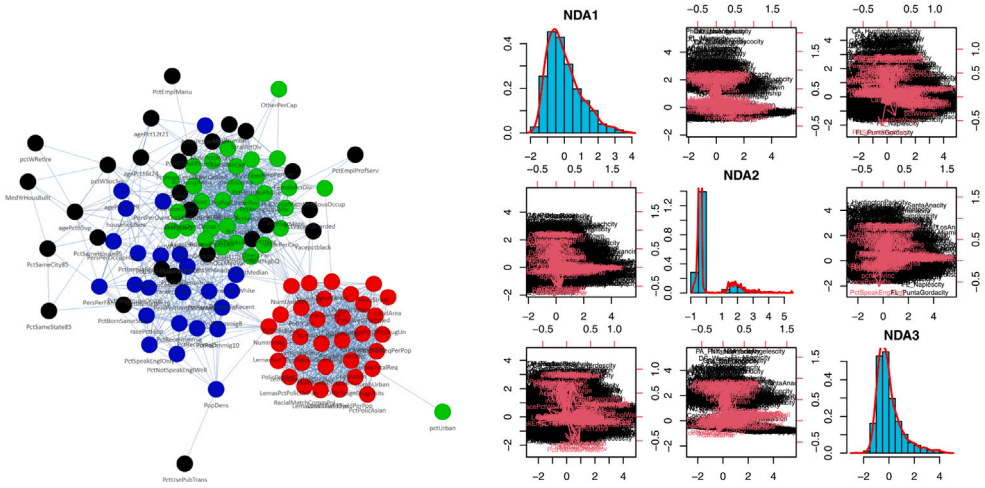
Despite differences in the top five cities, both the highest income and the highest challenges related to immigration occurred in California (CA) County. Police presence was high in large cities, such as Los Angeles, New York, Philadelphia, Miami, and Washington; however, in these cities, the detected crimes per population were also high.

5.4. Effects of filtering

Although GNDA is a nonparametric method, neither hyperparameter ($c_{\min}, h_{\min}, C_{\min}$) used for feature selection influences the proposed number of LVs. Nevertheless, there are two options to filter or mitigate the values of connection between indicators. Instead of using square correlation, a square of partial and semipartial correlations can be used as a similarity measure between indicators. These types of correlations filter the indirect effects between indicators. They measure the degree of association between two indicators, with the effect of a set of controlling indicators removed. Therefore, partial and semipartial correlations are not greater than correlations between indicators. The filtering indirect effects make the similarity graph sparser and it

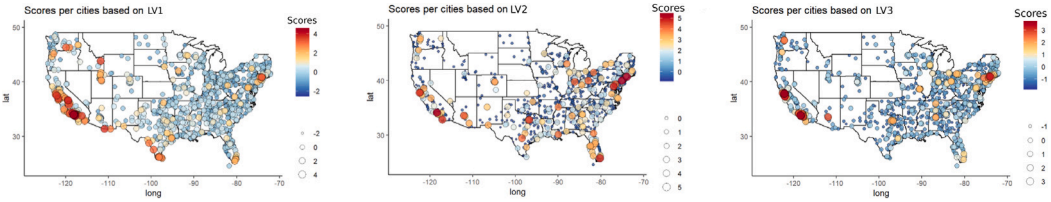


(a) Word cloud (GNDA) LV_1 (b) Word cloud (GNDA) LV_2 (c) Word cloud (GNDA) LV_3



(d) Distance correlation network (cut=0.2) (e) Biplot of latent variables

Fig. 11. Word cloud of terms of the variable descriptions (a-c), distance correlation network of indicators (d), and biplot of latent variables (e) (GNDA with feature selection, $C_{\min} = h_{\min} = 0.1, c_{\min} = 0.065$).



(a) LV1 - Income (b) LV2 - Police prescence (c) LV3 - Immigrants

Fig. 12. Spatial heatmap of score values of LVs (NDA).

specifies more modules (see Fig. A.15(a) in Appendix). In addition, semipartial correlation is usually nonsymmetric; therefore, it specifies a directed similarity graph between indicators (see Fig. A.16(a) in Appendix). In this case, the communality analysis (see Step 5) of AFS cannot be used because the communalities can be interpreted only on correlations. However, peripheral analysis (see Step 4) can be employed. In this way, individual and slightly connected nodes can be omitted in the case of calculating LVs (see Figs. A.15(b) and A.16(b) in Appendix). This analysis is worthwhile if indirect effects must be filtered. In this case, the examination of the abandoned variables becomes more valuable since the variables thus left-alone variables and small communities do not directly correlate with any other variables;

Table 4
Top five indicators with greatest loadings.

Method			Income	Police presence	Immigrants
PCA	Correlations		−0.5047	0.4396	0.2330
	Top 5 variables:	1.	median family income	number of different kinds of drugs seized	percent of people who do not speak English well
		2.	median household income	number of people living in areas classified as urban	percent of the population who have immigrated within the last 10 years (numeric
		3.	per capita income	population for community	percent of the population who have immigrated within the last 10 years
		4.	percentage of households with investment/rent income	number of police cars	percent of people foreign-born
		5.	rental housing - upper quartile rent	percent of sworn full-time police officers on patrol	percent of the population who have immigrated within the last 8 years
PFA	Correlations		−0.5106	0.5004	0.3217
	Top 5 variables:	1.	median family income	number of different kinds of drugs seized	percent of people who do not speak English well
		2.	median household income	number of people living in areas classified as urban	percent of persons in dense housing
		3.	per capita income	number of police cars	percent of all occupied households that are large
		4.	rental housing - upper quartile rent	population for community	percent of family households that are large (6 or more)
		5.	rental housing - median rent	a measure of the racial match between the community and the police force	percent of the population who have immigrated within the last 10 years
GNDA	Correlations		−0.3629	0.3211	0.3871
	Top 5 variables:	1.	median family income	number of different kinds of drugs seized	percent of persons in dense housing (more than 1 person per room)
		2.	median household income	a measure of the racial match between the community and the police force	percent of the population who have immigrated within the last 5 years
		3.	rental housing - median rent	percent of sworn full-time police officers on patrol	percent of the population who have immigrated within the last 10 years
		4.	rental housing - upper quartile rent	number of sworn full-time police officers in field operations	percent of the population who have immigrated within the last 8 years
		5.	median gross rent	percent of police that are caucasian	percent of the population who have immigrated within the last 3 years

Table 5
Top five cities with highest scores.

Top five scores	LV1 - Income	LV2 - Police presence	LV3 - Immigrations
1	Piedmontcity CA	Los Angeles City CA	Huntington Park City CA
2	Orindacity CA	New York City CA	Paramount City CA
3	Manhattan Beach City CA	Philadelphia City PA	Santa Ana City CA
4	Rancho Palos Verdes City CA	Maimi City FL	South El Monte City CA
5	La Canada Flintridge City CA	Washington DC	Bell City CA

they can show unique phenomena. However, a deeper analysis of this is beyond the scope of this paper.

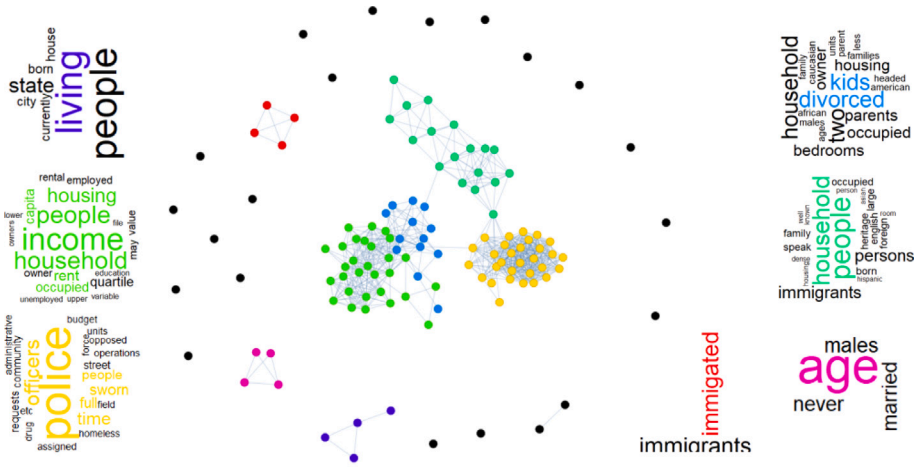
The other option is to increase the minimal (square) correlation ($R^2_{\min}$) value between variables. This leads to a similar situation: the number of modules can be increased. In this way, the original modules can be separated. However, this method has two advantages. First, communality analysis (Step 5) can be used further. Second, this prefiltering provides hierarchical clustering, which can be used to test the robustness of the community of indicators.

Fig. 13 shows the results of GNDA without (see Fig. 13(a)) and with (see Fig. 13(b)) feature selections, where the minimal number of variables in the module must not be lower than ten (min_comm=10).

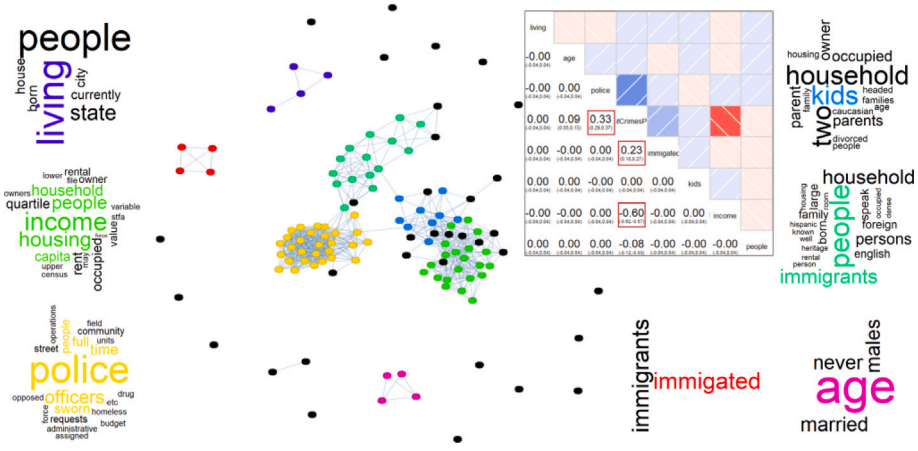
If the minimal variables in a latent variables should be more than ten, both GNDAs provide seven groups of LVs; however, GNDA without feature selections dropped variables, which became isolated nodes after prefiltering. If the minimum number of variables in a module is reduced, as the prefiltering parameter increases, we obtain not only isolated points (cluster marked 0) but also variable groups with a small number of indicators (see Fig. 14(b)). The most frequent terms in groups were living, income, police, kids, people, immigrated, and age. However, Fig. 13(b) shows that only LVs, where the most frequent terms were police, immigrated, and income, were significantly correlated with crimes per population.

Fig. 14(a) shows the hierarchy of variable groups with different prefiltering parameters ($R^2_{\min}$), where the minimal number of variables in a module is ten (min_comm=10). Fig. 14 shows the alluvial diagram, where the minimal number of variables in a module is two (min_comm=2).

In all levels of the hierarchical GNDA forest, three LVs, namely, immigrants, income, and police presence, were significantly correlated with the output (crimes per population) variable (see Fig. 13). Fig. 14 shows that variable group 1 (income) and variable group 3 (police) are the most stable, while variable group 2 (immigrants) early ($r_{\min} > 0.1$) is split into a bigger and a smaller modules. These charts provide an opportunity to check the cross-migration of variables between modules. The results showed that this cross-migration is lower than 3%. Nevertheless, in this case, the structure of the correlation graph is also changing because low-correlated arcs are eliminated. When the minimum square correlation between variables is increased, a separation of LVs is observed, as shown in Fig. 14(a).



(a) Without feature selection



(b) With feature selection

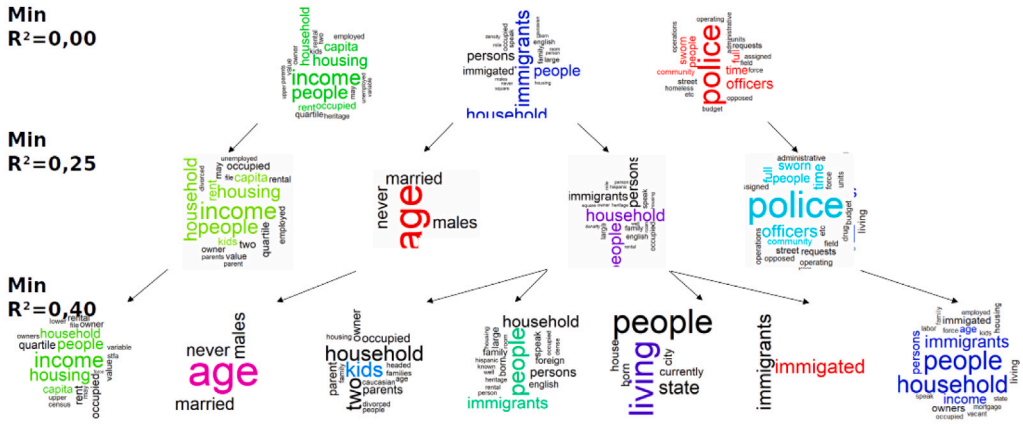
Fig. 13. Correlation graph and LVs in the case of prefiltering (minimal $R^2_{\text{min}} = 0.4$, min_comm = 10).

6. Discussion

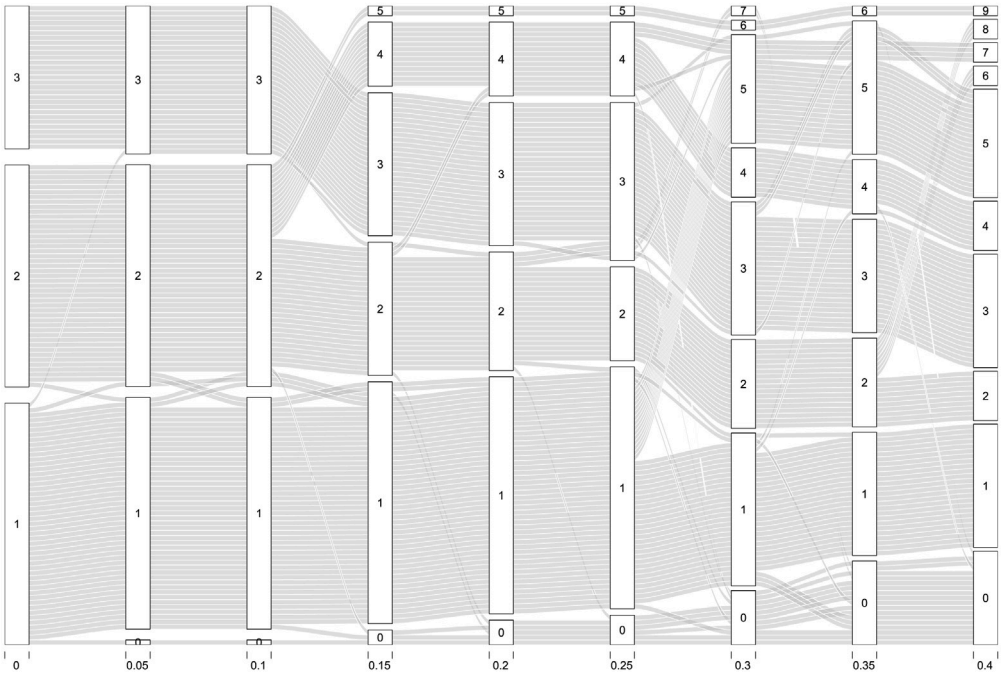
For unsupervised machine learning problems, we usually do not know exactly how many groups of variables and, in this way, how many LVs should be specified. Therefore, even recent methods, such as SPCA (Zhang et al., 2012), KPCA (Schölkopf et al., 1997, 1998), NNMF (Wang & Zhang, 2012), t-SNE (Liu et al., 2021), and VAE (Mahmud et al., 2021), usually leave it up to the user to choose the number of latent variables. However, the specification of the adequate number of LVs has a major impact on the interpretability of the results, especially when the data contain many variables HDLSS or many observations LDHSS. Procedures for estimating the number of latent variables are mainly found in the PCA and PFA methods. However, they identify a different number of factors (see Figs. 3 and 4 and Table 1). The paper proposes a method to generate either HDLSS or LDHSS block matrices to compare estimates of latent variables from dimensionality reduction methods. The results showed that GNDA was the only method that always found the exact number of latent variables. All variables are assigned to the adequate LVs (see Fig. 5).

In the real database, we also find that different methods overestimate the number of LVs (see Fig. 7(a) and Table 2), so we can only compare the results of different dimension reduction methods if we agree on a common number of latent variables (see Fig. 8(b–c) and 8). Nevertheless, recent methods are still debtor as to how many LVs can be run with. The estimates of correlation-based (PCA, PFA) methods and the result of GNDA can only be used as a guide. Assuming that the number of latent variables is correctly given by the GNDA method, the results show that the PCA and PFA methods define latent variables with similar content (compare Figs. 8 and 9(a–c), and Table 3). Importantly, the analysis was not designed to identify latent variables that correlate as closely as possible with the explained variable, which in this case was crime per population. For this reason, we did not select variables that increase the correlation between latent variables and the outcome variable the least. This would be a different *supervised learning* task, which is not part of this study. Here, the omission of variables was justified if they did not fit the latent variables. However, it should be noted that AFS increased the correlation in all cases. GNDA provided the clearest set of variables (compare Figs. 10 and 11); however, correlation-based methods generally provided higher correlations between latent and outcome variables (compare Tables 3 and 4). In other words, if a supervised learning version of the GNDA method is constructed, then other mechanisms for variable selection must be found than testing the fit to the latent variable.

The GNDA method predetermines the number of latent variables. The number of groups is not affected by the hyperparameters used in



(a) Word cloud of GNDA, (min_comm=10)



(b) Alluvial diagram of variables in GNDA, (min_comm=2, cluster 0 contains dropped variables; horizontal axis denotes the threshold of prefiltering)

Fig. 14. The hierarchical GNDA tree for prefiltering.

AFS. However, two parameters (γ and $r_{\min}$) can increase the number of latent variables. Increasing γ increases the resolution of the modules, and increasing the prefiltering parameter ($r_{\min}$) makes the correlation graph sparser; therefore, the original modules, and in this way, the LVs, are split down (see Figs. 13 and 14). Even though the examination of missing data was not the subject of our study, if the data are missing, then correlation relationships between variables are dropped from the correlation graph in the same way as in the case of increasing the parameters of the prefiltering. Those modules where the correlation between the variables is higher do not fall into modules even if the edges with a low correlation square are removed from the correlation graph. In the same way, increasing the resolution will not result in a new group of variables.

If the similarity between variables is the correlation function and the number of LVs is fixed according to the value of the GNDA method, then the proposed method can be compared with other dimensional reduction methods. However, if the similarity function is asymmetric, e.g., partial or semipartial correlation, the results cannot be compared with state-of-the-art methods. For block matrices, GNDA correctly identified the number of LVs based on partial (see Fig. 6(a)) and semipartial correlations (see Fig. 6(b)). At the same time, for real-world data, partial correlations can give a sparser similarity graph, which is why the GNDA method will also determine more latent variables (see Fig. A.15). Remember that, in this case, LVs can also be interpreted differently.

7. Summary and conclusion

Dimensionality reduction methods are essential parts of data analysis. The paper provides a new approach to transforming data into a network and allows scholars to combine descriptive network science and exploratory analytical methods. The proportion of observations and variables often varies. Therefore, robust methods should be used to compare results. Usually, different methods are used to solve HDLSS and LDHSS problems. The number of LVs is the most difficult

but crucial step in interpreting results. In this paper, we proposed a nonparametric network-based dimensionality analysis method, which determines the number of LVs. The proposed method works in HDLSS but also in a high number of observations. Hyperparameters, such as minimal EVC ($c_{\min}$), minimal communality ($h_{\min}$), and common communality values ($C_{\min}$), are only used for feature selection, but prefiltering — which increases the minimal (square) correlation between variables, $R^2_{\min}$) – can separate modules of variables and provides a hierarchy of variable groupings. For both simulated and real-world data, GNDA best identifies the number of LVs. In addition, GNDA can provide a cleaner group of variables. Finally, the proposed set of visual and analysis tools, such as correlation graphs, word clouds of terms, and the list of greatest common communalities, also supports the interpretation in the case of a high number of variables and observations. The replacement of applied methods, such as the correlation method, modularity method, and measure of centrality, provides possible further improvements. GNDA is implemented by MATLAB and R, and the developer version of GNDA is freely available at https://github.com/kzst/nda (accessed: 13 April 2023).

8. Limitations and future works

Our earlier study (Kosztyán et al., 2022) showed the employment of NDA in the case of HDLSS datasets. This paper focused on the case of a larger number of observations than variables. The implemented GNDA employs only Person's, Spearman's, Kendall's, and distance correlation methods; however, further distances, such as contingency and Jaccard's distance, can be implemented to handle this method for topic mining, where one of the crucial problems is to specify the number of topics. Furthermore, combining GNDA with advanced artificial intelligence techniques offers a promising approach to enhance the performance of the original algorithm. Another possible improvement would be to extend this method for biclustering and multiclustering, which would open this method up for further applications. Since the explored dimensionality reduction methods can only handle a symmetric correlation matrix or a symmetric distance matrix, it was not possible to compare the proposed method with existing methods. At the same time, the appendix contains the proposed dimensionality reduction methods obtained for partial and semipartial correlations. The discussion of these results should be elaborated on in a subsequent study not only due to page limitations but also because the factor loading and communality values must then be interpreted differently.

Acronyms

AFS	Automated Feature Selection
BDLRR	Block-Diagonal Low-Rank Representation
CFA	Common Factor Analysis
DENLR	Discriminative Elastic-Net Regularized Linear Regression
EFA	Explanatory Factor Analysis
EVC	eigenvector centrality
FA	Factor Analysis
GNDA	Generalized Network-based Dimensionality Analysis
HDLSS	High-Dimension Low-Sample-Size
K1	Kaiser's Rule
KPCA	Kernel Principal Component Analysis
LDHSS	Low-Dimension High-Sample-Size
LV	latent variable
MAP	Minimum Average Partial
MCVE	Minimal cumulative variance explained
MSRL	Marginally Structured Representation Learning
NDA	Network-based Dimensionality Reduction and Analysis
NNMF	Non-Negative Matrix Factorization
PA	Parallel Analysis
PCA	Principal Component Analysis
PFA	Principal Factor Analysis
SNA	Social Network Analysis
SPCA	Sparse Principal Component Analysis
t-SNE	t-distributed stochastic neighbor embedding
VAE	Variational Autoencoder

Code availabilty

The R package of GNDA can be downloaded from the CRAN official site: https://cran.r-project.org/web/packages/nda/index.html. The test version can be downloaded from https://github.com/kzst/nda or from the Code Ocean cite (Kosztyán, 2023).

CRediT authorship contribution statement

Zsolt T. Kosztyán: Conceptualization, Methodology, Software, Writing – original draft. **Attila I. Katona:** Data curation, Software, Writing – review & editing. **Marcell T. Kurbucz:** Data curation, Software, Writing – review & editing. **Zoltán Lantos:** Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data is included to the proposed package.

Acknowledgments

The authors would like to thank Prof. János Abonyi (University of Pannonia) and Prof. Károly Héberger (Eötvös Loránd Research Network, Institute of Excellence, Hungarian Academy of Sciences) for their valuable comments and advice.

Funding

The research is supported by the Research Centre at the Faculty of Business and Economics (PE-GTK-GSKK A095000000-7) of the University of Pannonia (Veszprém, Hungary). Project no. K 143482 was implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development, and Innovation Fund, financed under the K_22 "OTKA" funding scheme. Project no. PD 142593 was implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development, and Innovation Fund, financed under the PD_22 "OTKA" funding scheme.

Appendix

See Figs. A.15 and A.16 and Tables A.6–A.9.

Table A.6
20 LVs of PCA and the IDs of top five indicators.

PCA	LV_1	LV_2	LV_3	LV_4	LV_5
1	LemasSwFTFieldOps	RentHighQ	PctRecImmig10	PctIlleg	PersPerOccupHous
2	LemasSwFTFieldPerPop	RentMedian	PctRecImmig8	pctWPubAsst	PersPerFam
3	LemasSwFTPerPop	MedRent	PctRecImmig5	racePctWhite	household size
4	PolicPerPop	OwnOccLowQuart	PctRecentImmig	PctUnemployed	PersPerOwnOccHous
5	NumKindsDrugsSeiz	OwnOccMedVal	PctForeignBorn	PctNotHSGrad	PctLargHouseOccup
	LV_6	LV_7	LV_8	LV_9	LV_{10}
1	population	PctImmigRec5	pctWSocSec	PctHousLess3BR	agePct16t24
2	NumUnderPov	PctImmigRec8	agePct65up	MedNumBR	agePct12t29
3	NumInShelters	PctImmigRec10	pctWWage	PctHousOwnOcc	agePct12t21
4	NumIlleg	PctImmigRecent	pctWRetire	PctPersOwnOccup	MalePctNevMarr
5	HousVacant	PctSameHouse85	PctEmploy	PctFam2Par	PctHousOwnOcc
	LV_{11}	LV_{12}	LV_{13}	LV_{14}	LV_{15}
1	TotalPctDiv	PctSameState85	PctEmplProfServ	MedOwnCostPctIncNoMtg	PctWorkMomYoungKids
2	FemalePctDiv	PctSameCity85	PctEmplManu	MedYrHousBuilt	PctWorkMom
3	MalePctDivorce	PctBornSameState	PctBSorMore	PctUsePubTrans	PctEmploy
4	PctFam2Par	PctSameHouse85	PctOccupMgmtProf	PopDens	PctUnemployed
5	PctSameHouse85	PctEmplManu	PctOccupManu	PctVacMore6Mos	pctWPubAsst
	LV_{16}	LV_{17}	LV_{18}	LV_{19}	LV_{20}
1	HispPerCap	pctWFarmSelf	MedRentPctHousInc	PctHousOccup	fold
2	OtherPerCap	pctUrban	MedOwnCostPctInc	PctVacMore6Mos	indianPerCap
3	racePctHisp	PctUsePubTrans	PctEmplManu	pctUrban	PctWOFullPlumb
4	PctPersDenseHous	PopDens	PctEmploy	HousVacant	MedOwnCostPctInc
5	indianPerCap	numbUrban	PctPopUnderPov	PctHousNoPhone	PopDens

Table A.7
20 LVs of PFA and the IDs of top five indicators.

PFA	LV_1	LV_2	LV_3	LV_4	LV_5
1	PctKids2Par	perCapInc	PctRecImmig10	racePctWhite	PctImmigRec8
2	PctFam2Par	RentHighQ	PctRecImmig8	PctLargHouseFam	PctSameHouse85
3	NumUnderPov	RentMedian	PctRecImmig5	PctPersDenseHous	PctImmigRec5
4	PctPopUnderPov	OwnOccLowQuart	PctRecentImmig	PctLargHouseOccup	PctImmigRecent
5	PctYoungKids2Par	OwnOccMedVal	PctNotSpeakEnglWell	agePct12t29	PctImmigRec10
	LV_6	LV_7	LV_8	LV_9	LV_{10}
1	agePct65up	PctBSorMore	LemasSwFTFieldPerPop	agePct12t21	TotalPctDiv
2	pctWSocSec	PctOccupManu	LemasSwFTFieldOps	agePct16t24	MalePctDivorce
3	pctWWage	PctOccupMgmtProf	LemasSwFTPerPop	agePct12t29	FemalePctDiv
4	MedYrHousBuilt	PctNotHSGrad	PolicPerPop	whitePerCap	PctEmplManu
5	householdsize	PctSameCity85	PctPolicWhite	householdsize	PctBSorMore
	LV_{11}	LV_{12}	LV_{13}	LV_{14}	LV_{15}
1	MedYrHousBuilt	racepctblack	PctWorkMom	PctEmplProfServ	pctUrban
2	racepctblack	PctImmigRec8	PctWorkMomYoungKids	PctUsePubTrans	PopDens
3	PctUsePubTrans	PctImmigRec10	PctSameHouse85	PctBSorMore	MedYrHousBuilt
4	PopDens	PctImmigRec5	PctEmploy	PctImmigRec5	PctUsePubTrans
5	MalePctNevMarr	racePctWhite	MedRentPctHousInc	PctWorkMom	pctWFarmSelf
	LV_{16}	LV_{17}	LV_{18}	LV_{19}	LV_{20}
1	PctHousOccup	PctHousOccup	MedOwnCostPctInc	MedRentPctHousInc	MedRentPctHousInc
2	PctEmplManu	pctUrban	PctEmplProfServ	MedYrHousBuilt	pctUrban
3	HousVacant	MedRentPctHousInc	pctUrban	PctEmplProfServ	PctLargHouseFam
4	PersPerOccupHous	PctEmplProfServ	pctWFarmSelf	MedOwnCostPctInc	numbUrban
5	PersPerOwnOccHous	TotalPctDiv	PctPopUnderPov	PctPersDenseHous	racePctWhite

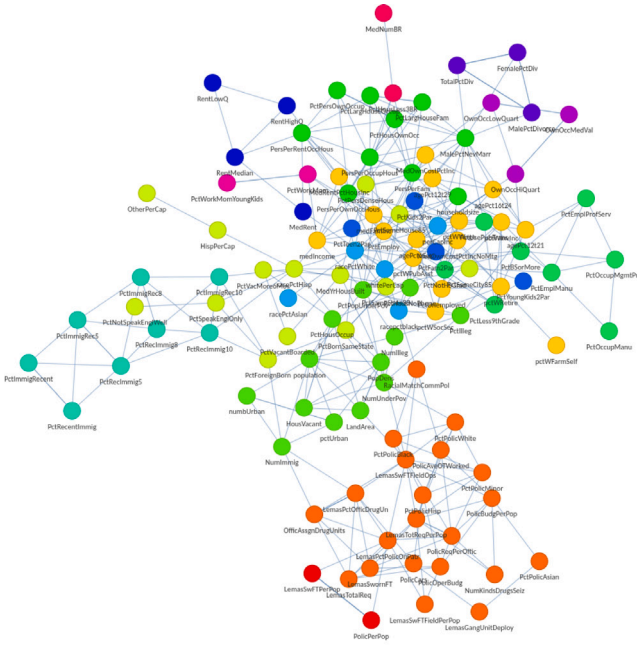
Table A.8
LVs of state-of-the-art dimension reduction methods, correlations between the output variable and LVs, and top five indicators, if the number of LVs are assumed as the result of GNDA (i.e., $n = 3$).

			LV_1	LV_2	LV_3
NNMF	Correlations		0.4967	−0.4983	0.3774
	Top 5 variables	1.	fold	fold	fold
		2.	LemasSwFTFieldOps	pctUrban	PctSpeakEnglOnly
		3.	pctUrban	PctYoungKids2Par	PctBornSameState
		4.	PctPolicWhite	racePctWhite	PctSameState85
		5.	LemasPctPolicOnPatr	PctKids2Par	PctSameCity85
SPCA	Correlations		−0.0352	−0.6504	−0.2817
	Top 5 variables	1.	fold	PctYoungKids2Par	PctSpeakEnglOnly
		2.	pctUrban	medIncome	PctBornSameState
		3.	RentHighQ	PctKids2Par	racePctWhite

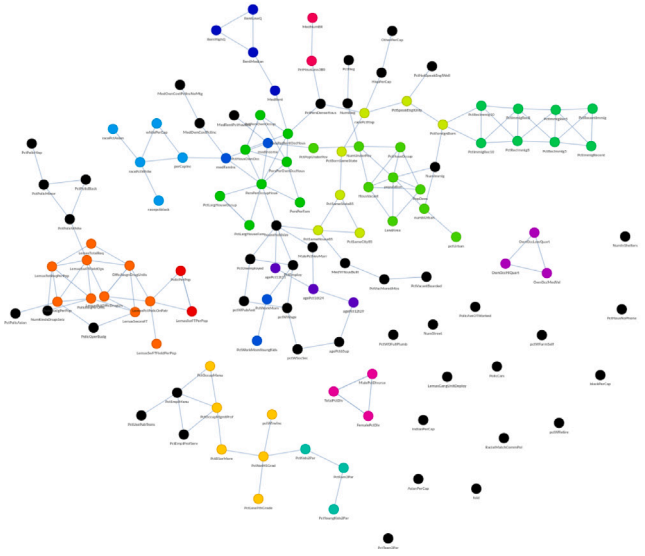
(*continued on next page*)

Table A.8 (*continued*).

			LV_1	LV_2	LV_3
		4.	medIncome	medFamInc	PctVacMore6Mos
		5.	PctYoungKids2Par	RentHighQ	pctWSocSec
KPCA	Correlations		−0.4646	−0.2211	0.0392
	Top 5 variables	1.	PctPolicAsian	agePct12t29	fold
		2.	LemasGangUnitDeploy	PersPerFam	pctUrban
		3.	PctPolicHisp	PersPerOwnOccHous	PctHousOccup
		4.	LemasPctOfficDrugUn	MedRentPctHousInc	PctSameState85
		5.	PctPolicBlack	householdsize	PctSpeakEnglOnly
t-SNE	Correlations		0.2376	−0.0511	−0.5696
	Top 5 variables	1.	fold	LemasGangUnitDeploy	OtherPerCap
		2.	PctHousOccup	PctPolicAsian	HispPerCap
		3.	PctEmploy	PctPolicWhite	OwnOccHiQuart
		4.	PctTeen2Par	LemasSwFTFieldOps	RentMedian
		5.	MedYrHousBuilt	PolicReqPerOffic	AsianPerCap

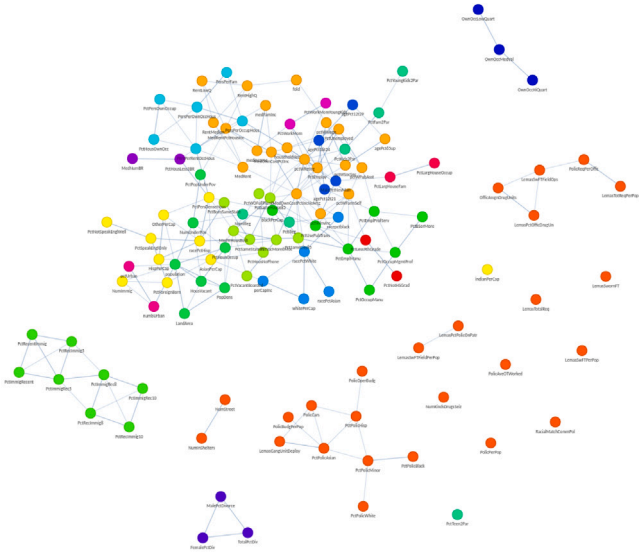


(a) GNDA without feature selection

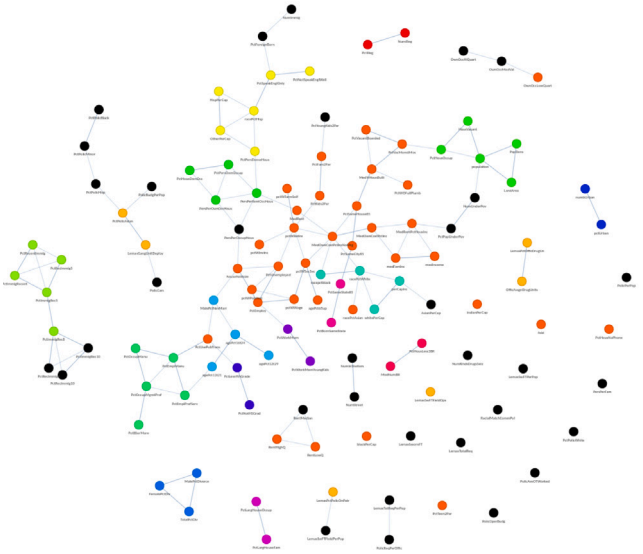


(b) GNDA with feature selection ($c_{\min} = 0.065$)

Fig. A.15. GNDA employed on Spearman's partial correlation graph. (Note: black nodes are dropped from the modules (8 factors).).



(a) GNDA without feature selection



(b) GNDA with feature selection ($c_{\min} = 0.065$)

Fig. A.16. GNDA employed on Spearman's semipartial correlation graph. (Note: black nodes are dropped from the modules (16 factors).).

Table A.9
The list of indicators in the Crimes 1994 dataset.

ID	Description
state	US state (by 2-letter postal abbreviation)(nominal)
county	county ID
community	community
communityname	communityname
fold	fold number for nonrandom 10-fold cross-validation, potentially useful for debugging, paired tests - not predictive (numeric - integer)
population	population of the community (numeric - expected to be an integer)
household size	mean people per household (numeric - decimal)
racepctblack	percentage of the population that is African American (numeric - decimal)
racePctWhite	percentage of the population that is Caucasian (numeric - decimal)
racePctAsian	percentage of the population that is of Asian heritage (numeric - decimal)
racePctHisp	percentage of the population that is of Hispanic heritage (numeric - decimal)
agePct12t21	percentage of the population that is 12-21 in age (numeric - decimal)
agePct12t29	percentage of the population that is 12-29 in age (numeric - decimal)
agePct16t24	percentage of the population that is 16-24 in age (numeric - decimal)
agePct65up	percentage of the population that is 65 and over in age (numeric - decimal)
numbUrban	number of people living in areas classified as urban (numeric - expected to be an integer)
pctUrban	percentage of people living in areas classified as urban (numeric - decimal)
medIncome	median household income (numeric - may be an integer)
pctWWage	percentage of households with wage or salary income in 1989 (numeric - decimal)
pctWFarmSelf	percentage of households with farm or self-employment income in 1989 (numeric - decimal)
pctWInvInc	percentage of households with investment/rent income in 1989 (numeric - decimal)
pctWSocSec	percentage of households with social security income in 1989 (numeric - decimal)
pctWPubAsst	percentage of households with public assistance income in 1989 (numeric - decimal)
pctWRetire	percentage of households with retirement income in 1989 (numeric - decimal)
medFamInc	median family income (differs from household income for nonfamily households) (numeric - may be an integer)
perCapInc	per capita income (numeric - decimal)
whitePerCap	per capita income for Caucasians (numeric - decimal)
blackPerCap	per capita income for African Americans (numeric - decimal)
indianPerCap	per capita income for native Americans (numeric - decimal)
AsianPerCap	per capita income for people with Asian heritage (numeric - decimal)
OtherPerCap	per capita income for people with 'other' heritage (numeric - decimal)
HispPerCap	per capita income for people with Hispanic heritage (numeric - decimal)
NumUnderPov	number of people under the poverty level (numeric - expected to be an integer)
PctPopUnderPov	percentage of people under the poverty level (numeric - decimal)
PctLess9thGrade	percentage of people 25 and over with less than a 9th grade education (numeric - decimal)
PctNotHSGrad	percentage of people 25 and over that are not high school graduates (numeric - decimal)
PctBSorMore	percentage of people 25 and over with a bachelor's degree or higher education (numeric - decimal)
PctUnemployed	percentage of people 16 and over, in the labor force, and unemployed (numeric - decimal)
PctEmploy	percentage of people 16 and over who are employed (numeric - decimal)
PctEmplManu	percentage of people 16 and over who are employed in manufacturing (numeric - decimal)
PctEmplProfServ	percentage of people 16 and over who are employed in professional services (numeric - decimal)
PctOccupManu	percentage of people 16 and over who are employed in manufacturing (numeric - decimal) #### No longer sure of a difference from PctEmplManu - may include unemployed manufacturing workers ####
PctOccupMgmtProf	percentage of people 16 and over who are employed in management or professional occupations (numeric - decimal)
MalePctDivorce	percentage of males who are divorced (numeric - decimal)
MalePctNevMarr	percentage of males who have never married (numeric - decimal)
FemalePctDiv	percentage of females who are divorced (numeric - decimal)
TotalPctDiv	percentage of the population who are divorced (numeric - decimal)
PersPerFam	mean number of people per family (numeric - decimal)
PctFam2 Par	percentage of families (with kids) that are headed by two parents (numeric - decimal)
PctKids2 Par	percentage of kids in family housing with two parents (numeric - decimal)
PctYoungKids2 Par	percent of children aged 4 and under in two-parent households (numeric - decimal)
PctTeen2 Par	percent of children aged 12-17 in two-parent households (numeric - decimal)
PctWorkMomYoungKids	percentage of moms of kids 6 and under in labor force (numeric - decimal)
PctWorkMom	percentage of mothers of children under 18 in the labor force (numeric - decimal)
NumIlleg	Number of illegal immigrants
PctIlleg	Percentage of illegal immigrants
NumImmig	total number of people known to be foreign born (numeric - expected to be an integer)
PctImmigRecent	percentage of _immigrants_ who immigrated within last 3 years (numeric - decimal)
PctImmigRec5	percentage of _immigrants_ who immigrated within the last 5 years (numeric - decimal)
PctImmigRec8	percentage of _immigrants_ who immigrated within the last 8 years (numeric - decimal)
PctImmigRec10	percentage of _immigrants_ who immigrated within the last 10 years (numeric - decimal)

(*continued on next page*)

Table A.9 (*continued*).

ID	Description
PctRecentImmig	percent of _population_ who have immigrated within the last 3 years (numeric - decimal)
PctRecImmig5	percent of _population_ who have immigrated within the last 5 years (numeric - decimal)
PctRecImmig8	percent of _population_ who have immigrated within the last 8 years (numeric - decimal)
PctRecImmig10	percent of _population_ who have immigrated within the last 10 years (numeric - decimal)
PctSpeakEnglOnly	percent of people who speak only English (numeric - decimal)
PctNotSpeakEnglWell	percent of people who do not speak English well (numeric - decimal)
PctLargHouseFam	percent of family households that are large (6 or more) (numeric - decimal)
PctLargHouseOccup	percent of all occupied households that are large (6 or more people) (numeric - decimal)
PersPerOccupHous	mean persons per household (numeric - decimal)
PersPerOwnOccHous	mean persons per owner-occupied household (numeric - decimal)
PersPerRentOccHous	mean persons per rental household (numeric - decimal)
PctPersOwnOccup	percent of people in owner-occupied households (numeric - decimal)
PctPersDenseHous	percent of persons in dense housing (more than 1 person per room) (numeric - decimal)
PctHousLess3BR	percent of housing units with fewer than 3 bedrooms (numeric - decimal)
MedNumBR	median number of bedrooms (numeric - decimal)
HousVacant	number of vacant households (numeric - expected to be an integer)
PctHousOccup	percent of housing occupied (numeric - decimal)
PctHousOwnOcc	percent of household owner occupied (numeric - decimal)
PctVacantBoarded	percent of vacant housing that is boarded up (numeric - decimal)
PctVacMore6Mos	percent of vacant housing that has been vacant more than 6 months (numeric - decimal)
MedYrHousBuilt	median year housing units built (numeric - may be an integer)
PctHousNoPhone	percent of occupied housing units without phones (in 1990, this was rare!) (numeric - decimal)
PctWOFullPlumb	percent of housing without complete plumbing facilities (numeric - decimal)
OwnOccLowQuart	owner-occupied housing - lower quartile value (numeric - decimal)
OwnOccMedVal	owner-occupied housing - median value (numeric - decimal)
OwnOccHiQuart	owner-occupied housing - upper quartile value (numeric - decimal)
RentLowQ	rental housing - lower quartile rent (numeric - decimal)
RentMedian	rental housing - median rent (Census variable H32B from file STF1A) (numeric - decimal)
RentHighQ	rental housing - upper quartile rent (numeric - decimal)
MedRent	median gross rent (census variable H43A from file STF3A - includes utilities) (numeric - decimal)
MedRentPctHousInc	median gross rent as a percentage of household income (numeric - decimal)
MedOwnCostPctInc	median owners cost as a percentage of household income - for owners with a mortgage (numeric - decimal)
MedOwnCostPctIncNoMtg	median owner cost as a percentage of household income - for owners without a mortgage (numeric - decimal)
NumInShelters	number of people in homeless shelters (numeric - expected to be an integer)
NumStreet	number of homeless people counted in the street (numeric - expected to be an integer)
PctForeignBorn	percent of people foreign born (numeric - decimal)
PctBornSameState	percent of people born in the same state as currently living (numeric - decimal)
PctSameHouse85	percent of people living in the same house as in 1985 (5 years before) (numeric - decimal)
PctSameCity85	percent of people living in the same city as in 1985 (5 years before) (numeric - decimal)
PctSameState85	percent of people living in the same state as in 1985 (5 years before) (numeric - decimal)
LemasSwornFT	number of sworn full-time police officers (numeric - expected to be an integer)
LemasSwFTPerPop	sworn full-time police officers per 100 K population (numeric - decimal)
LemasSwFTFieldOps	number of sworn full-time police officers in field operations (on the street as opposed to administrative, etc.) (numeric - expected to be an integer)
LemasSwFTFieldPerPop	sworn full-time police officers in field operations (on the street as opposed to administrative, etc.) per 100 K population (numeric - decimal)
LemasTotalReq	total requests for police (numeric - expected to be an integer)
LemasTotReqPerPop	total requests for police per 100 K population (numeric - decimal)
PolicReqPerOffic	total requests for police per police officer (numeric - decimal)
PolicPerPop	police officers per 100 K population (numeric - decimal)
RacialMatchCommPol	a measure of the racial match between the community and the police force. High values indicate proportions in the community and police force are similar (numeric - decimal)
PctPolicWhite	percent of police that are Caucasian (numeric - decimal)
PctPolicBlack	percent of police that are African American (numeric - decimal)
PctPolicHisp	percent of police that are Hispanic (numeric - decimal)
PctPolicAsian	percent of police that are Asian (numeric - decimal)
PctPolicMinor	percent of police that are a minority of any kind (numeric - decimal)
OfficAssgnDrugUnits	number of officers assigned to special drug units (numeric - expected to be an integer)
NumKindsDrugsSeiz	number of different kinds of drugs seized (numeric - expected to be an integer)
PolicAveOTWorked	police average overtime worked (numeric - decimal)
LandArea	land area in square miles (numeric - decimal)
PopDens	population density in persons per square mile (numeric - decimal)
PctUsePubTrans	percent of people using public transit for commuting (numeric - decimal)
PolicCars	number of police cars (numeric - expected to be an integer)
PolicOperBudg	police operating budget (numeric - may be an integer)
LemasPctPolicOnPatr	percent of sworn full-time police officers on patrol (numeric - decimal)
LemasGangUnitDeploy	gang unit deployed (numeric - integer - but truly nominal - 0 means NO, 10 means YES, 5 means Part Time)
LemasPctOfficDrugUn	percent of officers assigned to drug units (numeric - decimal)
PolicBudgPerPop	police operating budget per population (numeric - decimal)
ViolentCrimesPerPop	total number of violent crimes per 100 K population (numeric - decimal) GOAL attribute (to be predicted)

References

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *WIREs Computational Statistics*, *2*(4), 433–459. http://dx.doi.org/10.1002/wics.101.

Abonyi, J., Czvetkó, T., Kosztyán, Z. T., & Héberger, K. (2022). Factor analysis, sparse PCA, and sum of ranking differences-based improvements of the Prometheе-GAIA multicriteria decision support technique. *PLoS One*, *17*(2), Article e0264277. http://dx.doi.org/10.1371/journal.pone.0264277.

Ali, M. U., Ahmed, S., Ferzund, J., Mehmood, A., & Rehman, A. (2017). Using PCA and Factor Analysis for dimensionality reduction of Bioinformatics Data. arXiv preprint arXiv:1707.07189.

Aversano, G., Li, Z., Gicquel, O., & Parente, A. (2018). Model reduction by PCA and Kriging. In *International conference of computational methods in sciences and engineering* (pp. 1–4). URL https://dipot.ulb.ac.be/dspace/bitstream/2013/276893/3/main.pdf.

Bellman, R. (1957). *Rand corporation research study, Dynamic Programming*. Princeton University Press.

Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2008*(10), P10008. http://dx.doi.org/10.1088/1742-5468/2008/10/p10008.

Cichocki, A., Lee, N., Oseledets, I., Phan, A. H., Zhao, Q., & Mandic, D. P. (2016). Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions. *Foundations and Trends® in Machine Learning*, *9*(4–5), 249–429. http://dx.doi.org/10.1561/2200000059.

Cichocki, A., Phan, A. H., Zhao, Q., Lee, N., Oseledets, I., Sugiyama, M., et al. (2017). Tensor networks for dimensionality reduction and large-scale optimization: Part 2 applications and future perspectives. *Foundations and TrendsÂ® in Machine Learning*, *9*(6), 431–673. http://dx.doi.org/10.1561/2200000067.

Dettling, M., & Bühlmann, P. (2003). Boosting for tumor classification with gene expression data. *Bioinformatics*, *19*(9), 1061–1069. http://dx.doi.org/10.1093/bioinformatics/btf867.

Fabrigar, L., & Wegener, D. (2011). *Understanding statistics, Exploratory factor analysis*. Oxford University Press.

Fortunato, S., & Barthélemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, *104*(1), 36–41.

Gao, L., Song, J., Liu, X., Shao, J., Liu, J., & Shao, J. (2017). Learning in high-dimensional multimedia data: The state of the art. *Multimedia Systems*, *23*(3), 303–313. http://dx.doi.org/10.1007/s00530-015-0494-1.

Hair, J. F. J., William, C. B., Barry, J. B., & Rolph, E. A. (2020). *Multivariate data analysis* (7th ed.). Upper Saddle River, NJ: Prentice Hall.

Jolliffe, I. (2002). *Springer series in statistics, Principal component analysis*. Springer, URL https://books.google.hu/books?id=_olByCrhjwIC.

Jung, S., & Marron, J. S. (2009). PCA consistency in high dimension, low sample size context. *The Annals of Statistics*, *37*(6B), 4104–4130. http://dx.doi.org/10.1214/09-AOS709.

Khosla, N. (2004). *Dimensionality reduction using factor analysis*. Australia: Griffith University, http://dx.doi.org/10.25904/1912/3890.

Kim, H. J. (2008). Common factor analysis versus principal component analysis: Choice for symptom cluster research. *Asian Nursing Research*, *2*(1), 17–24. http://dx.doi.org/10.1016/S1976-1317(08)60025-0, URL https://www.sciencedirect.com/science/article/pii/S1976131708600250.

Kosztyán, Z. T. (2023). Generalized network-based dimensionality analysis and reduction (GNDA). http://dx.doi.org/10.24433/CO.1191558.v1, https://www.codeocean.com/.

Kosztyán, Z. T., Kurbucz, M. T., & Katona, A. I. (2022). Network-based dimensionality reduction of high-dimensional, low-sample-size datasets. *Knowledge-Based Systems*, Article 109180. http://dx.doi.org/10.1016/j.knosys.2022.109180.

Kurbucz, M. T. (2020). A joint dataset of official COVID-19 reports and the governance, trade and competitiveness indicators of world bank group platforms. *Data in Brief*, *31*, Article 105881. http://dx.doi.org/10.1016/j.dib.2020.105881, URL https://www.sciencedirect.com/science/article/pii/S2352340920307757.

Kurbucz, M. T., Katona, A. I., Lantos, Z., & Kosztyán, Z. T. (2021). The role of societal aspects in the formation of official COVID-19 reports: A data-driven analysis. *International Journal of Environmental Research and Public Health*, *18*(4), 1505. http://dx.doi.org/10.3390/ijerph18041505.

Li, Y., Li, G., Lian, H., & Tong, T. (2017). Profile forward regression screening for ultrahigh dimensional semiparametric varying coefficient partially linear models. *Journal of Multivariate Analysis*, *155*, 133–150. http://dx.doi.org/10.1016/j.jmva.2016.12.006.

Liu, H., Yang, J., Ye, M., James, S. C., Tang, Z., Dong, J., et al. (2021). Using t-distributed stochastic neighbor embedding (t-SNE) for cluster analysis and spatial zone delineation of groundwater geochemistry data. *Journal of Hydrology*, *597*, Article 126146. http://dx.doi.org/10.1016/j.jhydrol.2021.126146.

Mahmud, M. S., & Fu, X. (2019). Unsupervised classification of high-dimension and low-sample data with variational autoencoder-based dimensionality reduction. In *2019 IEEE 4th international conference on advanced robotics and mechatronics* (pp. 498–503). IEEE, http://dx.doi.org/10.1109/ICARM.2019.8834333.

Mahmud, M. S., Fu, X., Huang, J. Z., & Masud, M. A. (2018). High-dimensional limited-sample biomedical data classification using variational autoencoder. In *Australasian conference on data mining* (pp. 30–42). Springer, http://dx.doi.org/10.1007/978-981-13-6661-1_3.

Mahmud, M. S., Huang, J. Z., Fu, X., Ruby, R., & Wu, K. (2021). Unsupervised adaptation for high-dimensional with limited-sample data classification using variational autoencoder. *Computing and Informatics*, *40*(1), 1–28. http://dx.doi.org/10.31577/cai_2021_1_1.

Migenda, N., Möller, R., & Schenck, W. (2021). Adaptive dimensionality reduction for neural network-based online principal component analysis. *PLoS One*, *16*(3), Article e0248896. http://dx.doi.org/10.1371/journal.pone.0248896.

Nakayama, Y., Yata, K., & Aoshima, M. (2021). Clustering by principal component analysis with Gaussian kernel in high-dimension, low-sample-size settings. *Journal of Multivariate Analysis*, Article 104779. http://dx.doi.org/10.1016/j.jmva.2021.104779.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, *103*(23), 8577–8582. http://dx.doi.org/10.1073/pnas.0601602103.

Pop, A., Ciulca, S., et al. (2013). Correlative analysis of the relationships among different yield traits in dry bean. *Research Journal of Agricultural Science*, *45*(3), 149–154.

Redmond, M., & Baveja, A. (2002). A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, *141*(3), 660–678. http://dx.doi.org/10.1016/S0377-2217(01)00264-8.

Reichardt, J., & Bornholdt, S. (2004). Detecting fuzzy community structures in complex networks with a Potts model. *Physical Review Letters*, *93*(21), Article 218701. http://dx.doi.org/10.1103/PhysRevLett.93.218701.

Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E*, *74*(1), Article 016110. http://dx.doi.org/10.1103/PhysRevE.74.016110.

Revelle, W. (2022). *psych: Procedures for psychological, psychometric, and personality research*. Evanston, Illinois: Northwestern University, R package version 2.2.3. URL https://CRAN.R-project.org/package=psych.

Schölkopf, B., Smola, A., & Müller, K. R. (1997). Kernel principal component analysis. In *International conference on artificial neural networks* (pp. 583–588). Springer, http://dx.doi.org/10.1007/BFb0020217.

Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, *10*(5), 1299–1319. http://dx.doi.org/10.1162/089976698300017467.

Stippinger, M., Hanák, D., Kurbucz, M. T., Hanczár, G., Törteli, O. M., & Somogyvári, Z. (2023). BiometricBlender: Ultrahigh dimensional, multiclass synthetic data generator to imitate biometric feature space. *SoftwareX*, *22*, Article 101366. http://dx.doi.org/10.1016/j.softx.2022.101297.

Székely, G. J., & Rizzo, M. L. (2013). The distance correlation t test of independence in high dimension. *Journal of Multivariate Analysis*, *117*, 193–213. http://dx.doi.org/10.1016/j.jmva.2013.02.012, URL https://www.sciencedirect.com/science/article/pii/S0047259X13000262.

Traag, V. A., Waltman, L., & van Eck, N. J. (2019). From Louvain to Leiden: Guaranteeing well-connected communities. *Scientific Reports*, *9*(1), 5233. http://dx.doi.org/10.1038/s41598-019-41695-z.

Van Der Maaten, L., Postma, E., & Van den Herik, J. (2009). Dimensionality reduction: A comparative review. *Journal of Machine Learning Research*, *10*, 66–71.

Velicer, W. F. (1976). Determining the number of components from the matrix of partial correlations. *Psychometrika*, *41*(3), 321–327. http://dx.doi.org/10.1007/BF02293557.

Velicer, W. F., Eaton, C. A., & Fava, J. L. (2000). Construct explication through factor or component analysis: A review and evaluation of alternative procedures for determining the number of factors or components. *Problems and Solutions in Human Assessment*, 41–71. http://dx.doi.org/10.1007/978-1-4615-4397-8_3.

Wang, Y. X., & Zhang, Y. J. (2012). Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, *25*(6), 1336–1353. http://dx.doi.org/10.1109/TKDE.2012.51.

Zhang, Y., d'Aspremont, A., & El Ghaoui, L. (2012). Sparse PCA: Convex relaxations, algorithms and applications. In *Handbook on semidefinite, conic and polynomial optimization* (pp. 915–940). Springer.

Zhang, Z., Lai, Z., Xu, Y., Shao, L., Wu, J., & Xie, G.-S. (2017). Discriminative elastic-net regularized linear regression. *IEEE Transactions on Image Processing*, *26*(3), 1466–1481.