An end-to-end workflow for high-throughput discovery of clinically relevant insights from large biomedical datasets - Additional File 1

Multivariate Analysis and Visualization using R Package muvis

Abstract

Increased application of multivariate data in many scientific areas has considerably raised the complexity of analysis and interpretation. Although quite a few approaches have been suggested to address this issue, there is still a gap between the most efficient proposed methods and available software. **muvis** is an R package (core team (2017)) which is a toolkit for analyzing multivariate datasets. Several tools are implemented for common analyses of multivariate datasets, including preprocessing, dimensionality reduction, statistical analysis, Probabilistic Graphical Modeling, hypothesis testing, and visualization. Furthermore, we have implemented two novel methods–Variable-wise Kullback-Leibler Divergence (`VKL`) and Violating Variable-wise Kullback-Leibler Divergence (`VVKL`)–in **muvis**, which are proposed to find the features with most different probability distributions between two groups of samples. The main aim of the package is to provide a wide range of users with different levels of expertise in R with a set of tools for comprehensive analysis of multivariate datasets. We exploited the NHANES dataset to declare the functionality of **muvis** in practice.

Keywords: Probabilistic Graphical Models, Variable-wise Kullback-Leibler Divergence, Multivariate Analysis, Multivariate Visualization, Statistical Modeling.

Contents

1. Introduction

The recent advent of huge data in a wide range of scientific fields such as sociology, environmental research, economics, and biomedical research has raised demands for methods and tools to interpret and analyze high-dimensional data, where each dataset contains a large number of measurements or variables. Quite a few approaches have been put forward to analyze multivariate datasets (Timm (2004), Coghlan (2014), Esbensen, Guyot, Westad, and Houmoller (2002)). There are a number of widely-used approaches for analysis of multivariate data, including hypothesis testing to assess the significance of the association between two variables, fitting linear or non-linear models to associate one feature to another feature or a set of other features in the dataset, and using correlation analysis to capture how variables interact with or influence each other (Cohen, West, and Aiken (2014)).

The usual practice for analyzing multivariate data includes several steps: (i) Pre-processing and quality assurance, including identification and filtering of outliers and low-quality samples and missing data imputation. (ii) Multivariate analysis with possibly many different approaches, including dimensionality reduction, hypothesis testing, predictive models, correlation analysis, graphical modeling, etc. (iii) Visualization and interpretation of the results, including uni-, bi- and multivariate plots, interactive and dynamic graphical representations, network visualization of interactions, etc. Several R packages have been developed so far for

carrying out any of the above tasks (Tsagris (2016), Everitt and Hothorn (2011), Lê, Josse, and Husson (2008)). Since there is no single package providing all of these functionalities, conducting the whole analysis requires using many different packages and re-adaptation of data among them, which is a cumbersome challenge for many people of different scientific backgrounds who need to analyze their multivariate data.

Here we introduce **muvis** as a comprehensive toolkit for multivariate analysis and visualization providing an end-to-end analysis pipeline. Furthermore, we highlight the necessity of the paradigm shift from regular correlation analysis to Probabilistic Graphical Modeling in multivariate settings. Additionally, we introduce two novel distribution-based methods based on Kullback-Leibler Divergence analogous to hypothesis testing (we use the term *KL-based methods* to refer to these two methods). These methods will be introduced in details in the following sections.

This paper is organized as follows: Section 2 gives a brief theoretical background of Graphical Models (GMs) and the novel KL-based methods. In section 3, the implementation of the package functions is described in details. Section 4 contains the results of the package, applied on a real multivariate dataset. Our conclusions are drawn in the final section.

2. Theoretical Background

2.1. Graphical Models

One of the key challenges in the analysis of a large dataset, containing many variables (i.e., measurements) and observations (i.e., samples), is to capture the associations among variables and represent them in a simple manner. Consider a dataset of n variables measured in m observations. One can investigate $\binom{n}{2}$ pairwise associations among variables, which is computationally intensive and difficult to interpret. Therefore, the objective is to prune the complete graph to one containing a subset of key associations rather than all possible links. To this end, one of the most acclaimed approaches is to use Probabilistic Graphical Models (PGMs), in which partial (conditional) dependencies among variables are represented as a sparse graph.

Graphical Models (GMs) are renowned for modeling relations among variables in a compact manner. Based on principles in probability and graph theory, they supply effective tools to deal with complexity as well as uncertainty underlying the structure of associations among variables. More precisely, given a multivariate random variable, PGMs are aimed to describe the probability distribution of the variable which is equivalent to the structure of the partial dependencies among variables (Lauritzen (1996), Koller, Friedman, Getoor, and Taskar (2007)). The most distinctive feature of GMs is partial independence structure in the joint distribution of the variables which is often sparse even in complex phenomena. This leads to a sparse representation of the dependency structure. The theory behind GMs is described in more details in the following section. In parallel with theoretical developments, several software packages are developed for analysis of data using GMs. Particularly, R community has made a significant contribution in this regard (Højsgaard, Edwards, and Lauritzen (2012)).

- **Markov Networks (Markov Random Fields):** A Markov Random Field is a

joint probability distribution of a number of Gaussian random variables $X_1, X_2, \ldots, X_n$ represented as an undirected graph G. Each node of G represents a variable, and each edge indicates a non-zero partial correlation between a pair of variables (Rue (2005)). Here, we focus on Gaussian Graphical Models (GGMs). Gaussianity is proposed to be a reasonable assumption according to its mathematical simplicity and its dominance in nature (Véron and Rohrbasser (2003); Uhler (2017)).

– *Conditional Dependence:* In many statistical analyses, the problem is to find the relationships (dependence structure) among a subset of variables given occurrence of an event. This concept is defined as conditional dependence in statistics. Given three random variables, X_i, X_j, and X_k, X_i and X_j are independent conditioned on X_k if and only if

$$P(X_i; X_j|X_k) = P(X_i|X_k)P(X_j|X_k). \tag{1}$$

– *Precision Matrix:* Let $X = (X_1, X_2, \ldots, X_n)$ be an n-dimensional normally-distributed random vector. Assuming $X \sim N_n(\mu, \Sigma)$, the density function of X can be shown as

$$f_{\mu,\Sigma}(x) = (2\pi)^{-n/2}(det\Sigma)^{-1/2}e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}, \tag{2}$$

were μ is mean and Σ is the covariance matrix of X. We call Σ^{-1} the concentration/precision matrix. An interesting property of this formula is that the precision matrix consists of partial correlations such that Σ^{-1}_{ij} is equal to the partial correlation of X_i and X_j (Wasserman (2013)). It can be shown that zero partial correlation is equivalent to partial independence. Throughout this paper we will use zero (non-zero) partial correlation and partial independence (dependence) alternatively. Therefore, a normal distribution can be represented as a graph $G = (V, E)$ in which V is the set of vertices (nodes), each node V_i representing a variable X_i, and E is the set of edges such that for an edge e_{ij} exists if and only if $\Sigma^{-1}_{ij} \neq 0$. We call such graph a GGM. The objective is to estimate the precision matrix. Given m identically independent distributed observations $X^{(1)}, X^{(2)}, \ldots, X^{(m)}$ from $N_n(\mu, \Sigma)$ the log-likelihood function can be written as

$$\begin{aligned} l(\mu, \Sigma) &= C(-\frac{m}{2}log\,det(\Sigma) - \frac{1}{2}\Sigma_{i=1}^{m}(X^{(i)} - \mu)^T\Sigma^{-1}(X^{(i)} - \mu)) \\ &\propto \frac{m}{2}log\,det(\Sigma) - \frac{1}{2}tr(\Sigma^{-1}(\Sigma_{i=1}^{m}(X^{(i)} - \mu)(X^{(i)} - \mu)^T) \\ &= -\frac{m}{2}log\,det(\Sigma) - \frac{m}{2}tr(S\Sigma^{-1}) - \frac{m}{2}(\bar{X} - \mu)^T\Sigma^{-1}(\bar{X} - \mu), \end{aligned} \tag{3}$$

where C is a positive constant. In addition, given $\mu = \bar{X}$, the term $-\frac{n}{2}(\bar{X} - \mu)^T\Sigma^{-1}(\bar{X} - \mu)$ is constant. Thus, in order to solve Maximum Likelihood (ML) estimation, we have to maximize the term $-log\,det(\Sigma) - tr(S\Sigma^{-1})$ on all possible covariance matrices Σ or if we assign $K = \Sigma^{-1}$ then we should maximize $log\,det(K) + tr(S\Sigma^{-1})$ for K in the set of all possible precision matrices, where $S = \Sigma_{i=1}^{n}(X^{(i)} - \mu)(X^{(i)} - \mu)^T = \hat{\Sigma}$ (Lauritzen (1996)). The aim now is to construct a GM, G.

– *Model Selection:* So as to construct the GM, we must select the best model among all possible models, i.e., for a graph with v nodes we have to select the most likely graph fitting the model among all graphs with v nodes and possible $\binom{v}{2}$ edges.

  * *Stepwise Methods:* A collection of methods proposed for model selection are based on the stepwise approach. In this method, we start from a graph with no edge (or all possible edges) and follow sufficient forward (or backward) steps to construct the graph. We introduce two stepwise methods implemented in our package, below:

    · *Akaike Information Criterion (AIC):* Akaike Information Criterion is based on minimizing negative of log-likelihood, penalized by model complexity (Burnham, Anderson, and Huyvaert (2010)). The AIC of the model, with parameter k, is

    $$AIC(k) = -2l + k|E|, \tag{4}$$

    in which $|E|$, number of edges, stands for model complexity (or degree of freedom). In AIC, at each step of forward (or backward) stepwise algorithm, we add (or remove) the edge which minimizes the AIC (or decrease it over some threshold).

    · *Bayesian Information Criterion (BIC):* Bayesian Information Criterion is very similar to AIC except for k which has to be equal to $log(m)$ when we have m observations of each variable (Claeskens and Hjort (2008)).

  * *Thresholding:* A very simple method for constructing GGMs is to threshold partial correlations. In this method, the edge between two nodes with partial correlation above a threshold will be added to, otherwise will be eliminated from the graph. However, one should estimate the whole precision matrix prior to use this method and the estimation may not be feasible due to non-positive-definiteness of the covariance matrix.

  * *Significance Tests:* According to Gaussianity, one can test if a partial correlation is zero. To this end, consider the estimation of $\rho_{i,j}$ as the partial correlation between variables i and j, it can be shown that one has

    $$\hat{\rho}_{i,j} = S_{ij} - S_{i,V-\{i,j\}} S^{-1}_{V-\{i,j\},V-\{i,j\}} S_{V-\{i,j\},j}. \tag{5}$$

    Now consider Fisher's z-transform as

    $$\hat{z}_{i,j} = \frac{1}{2} log\Big(\frac{1+\rho_{i,j}}{1-\rho_{i,j}}\Big). \tag{6}$$

    The test statistic $T_m = \sqrt{m-p+2-3}|\hat{z}_{i,j}|$ can be used with a rejection of $R_n = (-F^{-1}(1-\alpha/2), F^{-1}(1-\alpha/2))$, with F, the cumulative distribution of standard normal, to a test of power α (Drton and Perlman (2007)).

  * *glasso:* glasso proposed to maximize the penalized log-likelihood function,

    $$l - \lambda|K|_1 \tag{7}$$

    were λ is penalizing parameter to determine model sparsity and $|K|_1$ is the sum of absolute values of off-diagonals of the precision matrix to control the model complexity. This leads to a convex programming problem which is straightforward to solve. Additionally, glasso algorithm can be applied well to high-dimensional settings (Friedman, Hastie, and Tibshirani (2008)).

- **Bayesian Networks (Causal Networks):** In every multivariate setting, an interesting investigation is to find causal effects among the variables, that is, to find which variable, measurement, or feature is a cause of another. Although yet there has been no algorithm proposed to capture the whole causal association set among a set of variables, there are some algorithms with satisfying efficiency, developed for causal inference (Drton and Perlman (2007)). The formalism of the problem is as follows:
Assume X to be an n-dimensional random variable with density function f. One can factorize f as

$$f_X(x) = \prod_{i=1}^{n} f_{X_i}(x_i|Pa(X_i)), \tag{8}$$

where $Pa(X_i) \subseteq \{X_1, X_2, \ldots, X_n\}$ (Pa stands for Parent). A Bayesian network is a directed graph like $G = (V, E)$, where V is the vertex set in which each vertex, like v_i, represents a variable, like X_i, and each edge, like e_{ij}, in the set edge E stands for conditional dependence between the two variables, like X_i and X_j, given all other variables. In G each edge ending at v_i is either directed from corresponding vertices of $Pa(X_i)$ to v_i or from v_i to v_j iff $X_i \in Pa(X_j)$. Accordingly, the direction of each edge implies probabilistic causality, since, given the states of parents, the probability distribution of child will be determined. It can be shown that this setting leads to a Directed Acyclic Graph (DAG) representing f due to **Markov property** (Drton and Perlman (2007)). Thus, in order to construct a Bayesian network one must first estimate such a factorization of f. To that end, one approach is to first find conditional (partial) independence structure of the multivariate model and then find directions (causal relations) among the structure. In the next section, we introduce a constraint-based algorithm we implemented in **muvis**, named FCI.

  - *Fast Causal Inference Algorithm:* FCI is a constraint-based algorithm first proposed in (Kalisch and Martin (2010)). This class of algorithms, aim to find some constraints given observed data that are necessary if the variables have a specific causal structure. Afterward, the causal structure will be estimated according to the set of constraints in hand. FCI is a generalization of **PC-algorithm** (Kalisch, Maechler, Colombo, and Kalisch (2010)). PC-algorithm consists of three prominent steps. In the beginning, the undirected **skeleton** of the graph is estimated. Then, for each edge like (u, v) which is present in the graph, the algorithm checks if there is any subset of nodes that can separate the two ends of the edge (u, v), i.e., if there is any subset like S which u & v are independent$|S$. This step can be carried out by checking the constraints mentioned before. In the next step, the $v1 - v2 - v3$ substructures (we will call such substructures, $v - structures$) are oriented with some rules (Kalisch *et al.* (2010)). Finally, the algorithms use some rules to orient further edges avoiding directed cycles (Kalisch *et al.* (2010)). In FCI is based upon PC-algorithm assuming the existence of some hidden variables. The first part of the FCI algorithm is the same as the PC-algorithm. In light of the existence of hidden variables, excluding edges due to some conditional subset separations is not sufficient. So the algorithm uses more rules to remove more edges due to the possibility of the presence of hidden variables. Description of the details of these algorithms is beyond the scope of this paper (readers can find the details in Kalisch and Martin (2010) and Kalisch, Hauser, Maathuis, and Mächler

(2018)). There is also a faster version of FCI which is computationally cheaper and is known as an approximation of FCI, named RFCI.

- **Minimal Forest for High-dimensional Modeling:** When the number of variables is too large (hundreds and thousands of variables), simple Graphical Modeling algorithms may fail both statistically (efficiency) and computationally (performance). Correspondingly, proper algorithms should be used in order to address these issues in high-dimensional settings. In the following, we will introduce an algorithm called minimal forest which is designed for high-dimensional Graphical Modeling.

  - *The Chow-Liu Algorithm:* Chow and Liu proposed an algorithm based on maximum weight spanning tree algorithm to find the maximum likelihood tree for multinomial discrete distributions. The algorithm is fast enough to be applied to high-dimensional data. The formulation comes in the following.

    Given an $m \times n$ dataset with m observations of n discrete variables, we aim to fit a maximum likelihood tree to the variables. Suppose that V is the set of variables (nodes) and E is the set of associations (edges). Chow and Liu (1968) showed that the probability of observing $V = v$ can be written as

    $$P(v) = \frac{\prod_{(V_i,V_j)\in E} P(V_i = v_i, V_j = v_j)}{\prod_{i\in V} P(V_i = v_i)^{d_i - 1}} \tag{9}$$

    were P is the probability distribution function and d_i is the degree of the node v_i in the tree. It can be shown that the maximum log-likelihood is the summation of mutual information between each pair of variables. Where the mutual information between V_i and V_j is defined as

    $$I_{i,j} = \sum_{v_i,v_j} \sum I(V_i = v_i, V_j = v_j) log \frac{\sum I(V_i = v_i, V_j = v_j)}{\sum I(V_i = v_i) \sum I(V_j = v_j)}, \tag{10}$$

    where I is the indicator function. Thus, if we use $I_{i,j}$ as the weight of the edge (V_i, V_j), applying the maximum spanning tree algorithm on the graph will lead to the maximum likelihood tree (Chow and Liu (1968)). As mutual information can be also defined for continuous distributions, this method can be extended to continuous variables and also mixed distributions of continuous and discrete variables (Edwards, de Abreu, and Labouriau (2010b)). Finally, so as to construct the maximum spanning tree from a connected graph, the Kruskal's algorithm can be used (Kruskal (1956)).

  - *AIC/BIC minimal forest:* Although extracting maximum-likelihood spanning tree gives a spare representation of the associations within the set of variable, it will force the representation to be connected, which may not be true for some settings. Respectively, extending the tree to a forest will address this problem. Edwards *et al.* (2010b) proposed a penalized mutual information measure based on AIC and BIC. In that paper they introduced $I^{AIC}_{i,j} = I_{i,j} - k|E|$ and $I^{BIC}_{i,j} = I_{i,j} - log(m)|E|$ as the alternative weights of the edge (V_i, V_j). After filtering out the edges with negative weights the Kruskal's algorithm can be employed to select the maximum spanning tree of the graph (Edwards *et al.* (2010b)).

2.2. Variable-wise KL-divergence

Kullback-Leibler Divergence: Kullback-Leibler divergence (KL-divergence) is a measure of dissimilarity of one probability distributions from another distribution (Fig. 1a). Given $D_{\mathrm{KL}}(P||Q)$ as the KL-divergence of distribution Q with respect to distribution P, $D_{\mathrm{KL}}(P||Q)$ indicates a measure of error, assuming Q when the real distribution is P (Kullback and Leibler (1951)). KL-divergence for discrete probability distributions P and Q is defined as

$$D_{\mathrm{KL}}(P\|Q) = \sum_{i} P(i)\ \log \frac{P(i)}{Q(i)}. \tag{11}$$

Moreover, KL-divergence can be applied to continuous distributions as

$$D_{\mathrm{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x)\ \log \frac{p(x)}{q(x)}\, dx. \tag{12}$$

It can be proved that KL-divergence is non-negative. One can use KL-divergence in a symmetric manner in order to find the distance between two distributions. Thus, $D_{\mathrm{sym}}(P, Q)$ may be defined as

$$D_{\mathrm{sym}}(P, Q) = D_{\mathrm{KL}}(P\|Q) + D_{\mathrm{KL}}(Q\|P). \tag{13}$$

- **Variable-wise Kullback-Leibler Divergence:** The Symmetric KL-divergence which is defined in the previous section can be used to compute the divergence of two groups of observations of a single measurement. In other words, if the measurement can discriminate the two groups efficiently. One can generalize this to a set of variables (measurements) with two fixed groups of observations (Fig. 1b). With such an approach, KL-divergence can be respected as a measure of importance along variables, thus, sorting variables regarding their KL-divergence values orders the list of variables in terms of importance. We call this approach Variable-wise KL-divergence (VKL).

- **Find Violating Variables with VKL:** Assuming two measurements (variables) to have a linear association, one can simplistically fit a line to the pair of variables. Outliers of a linear regression can be identified regarding their residuals. The observations with the most absolute values of residuals can be defined as outliers, by setting a cut-off. Subsequently, looking for differences (in measurements) of such outliers may lead to some valuable information (Fig. 1c), i.e., these outliers should have some features which block (violate) the expected linear association. For instance, suppose that we have a dataset including some laboratory (continuous) measurements and a number of clinical demographic (discrete) data collected from a population and we want to look if there are some features that may violate the positive (linear) association of Body Mass Index (BMI) as a measure of obesity and Diastolic Blood Pressure (DBP). Accordingly, looking for differences between up-outliers (relative high BMI and low DBP) and low-outliers (relative low BMI and high DBP) in every other measurement we have will lead us to the most important features that are potential to violate the expected association. To this end, we can use VKL-divergence in order to find the most different features of the two groups of outliers. Such features may be very informative because of the ability to block the expected linear association. In this specific example, one may be interested to see which features can block high DBP in relatively obese individuals, and also which features can cause high DBP in relatively slim individuals (Fig. 1c).

- **Significance Levels of KL-divergence:** In order to compute the significance level of KL-divergence, one can permute the members between two groups of observations to see if the KL value for the variable is significant (size of the two groups should be fixed). In this setting, after creating a lot of permuted groupings, the true KL-divergence value can be compared to the empirical distribution constructed by permuting groups (Fig. 2).

3. Package Implementation

3.1. Preprocessing

The `data_preproc` function can be used to preprocess raw datasets. The function is designed to address outlier-detection and imputation of missing data. In order to find outliers, `data_preproc` uses an anomaly-detection algorithm from Vallis, Hochenbaum, and Kejariwal (2014) for time series data (Fig. 3). For each variable, it sorts the observations in a decreasing (or increasing) order and it defines the anomalies detected by the algorithm, as outlier data (Fig. 4). The function then removes the outliers and behave them as missing observations. The missing observations are imputed by the mean or the median of the whole set of observations for the measurement if it is continuous or categorical, respectively. The method gets a parameter, `levels`, an integer value indicating the maximum number of levels of a categorical variable. The method returns the dataset with continuous variables as `numeric` and categorical variables as `factor` data types (see 4.2 for the practical example).

3.2. Test Associations

The function `test_pair` implements *Pearson's Chi-squared*, *ANOVA*, and *correlation* tests for categorical-categorical, categorical-continuous, and continuous-continuous pairs of variables, respectively. One can easily use `test_pair` (`test_assoc`) in order to test any desired association between two (multiple) variables. Additionally, `test_assoc` implements multiple hypothesis tests, using False Discovery Rate correction of Benjamini and Hochberg.

3.3. Plot Associations

The function `plot_assoc` is implemented to facilitate single- and pairwise-variable visualizations. For a single continuous or categorical variable, it creates a bar plot and density plot, respectively. Pairwise-variable visualizations consist of a boxplot of the continuous variable for different levels of the categorical one, a scatter plot for two continuous variables and a heatmap illustrating the relation of different levels of two categorical variables. There is also a logical parameter `interactive` indicating if the output plot a **highcharter** object (Kunst (2017)). It will output a print-friendly plot using R package **ggplot2** when the parameter is set to `False` (Wickham (2010)).

3.4. Graphical Models

As mentioned in 2.1, GMs are efficient for computation and interpretation of the whole structure of associations among the variables. However, each set of variables requires a proper

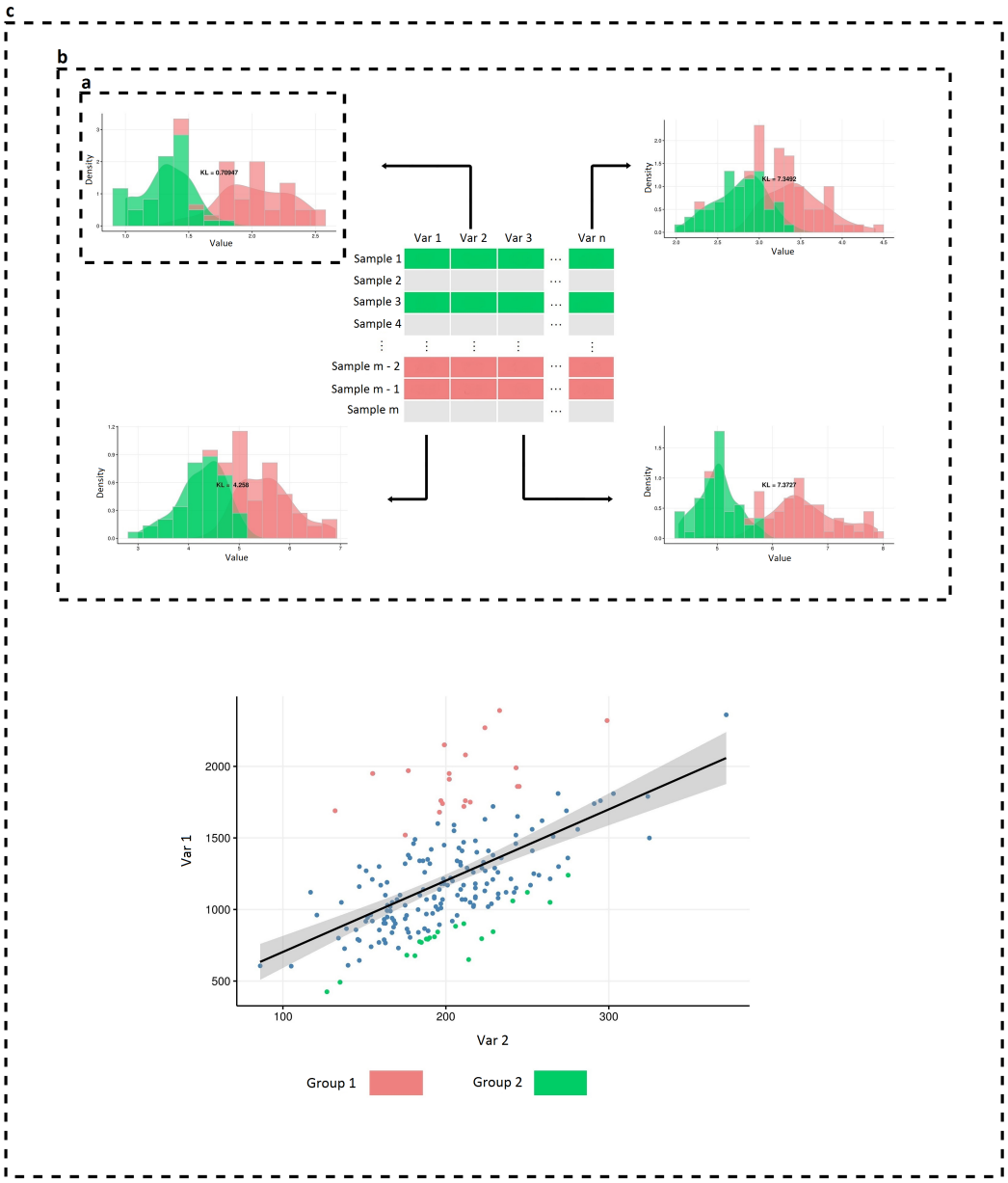


Figure 1: **Variable-wise KL-based methods.** Given a multivariate dataset with m samples and n variables, **a)** simple (symmetric) KL-divergence (KL) can be used in order to find the distance between probability distributions of two interesting groups of sample, colored in green and red, for a single variable; **b)** Variable-wise KL-divergence (VKL) can be used so as to calculate KL-divergence between two groups of samples for each variable, by calculating KL-divergence between the two groups on each variable; **c)** by fitting a linear model on two variables of interest, Violating Variable-wise KL-divergence (VVKL), finds the outliers with most absolute residuals in the linear model, the upper group is colored in red and the lower one is in green and these two groups of outliers will be then passed to VKL to find the violating (blocking) variables for the linear association expected on the two variables.

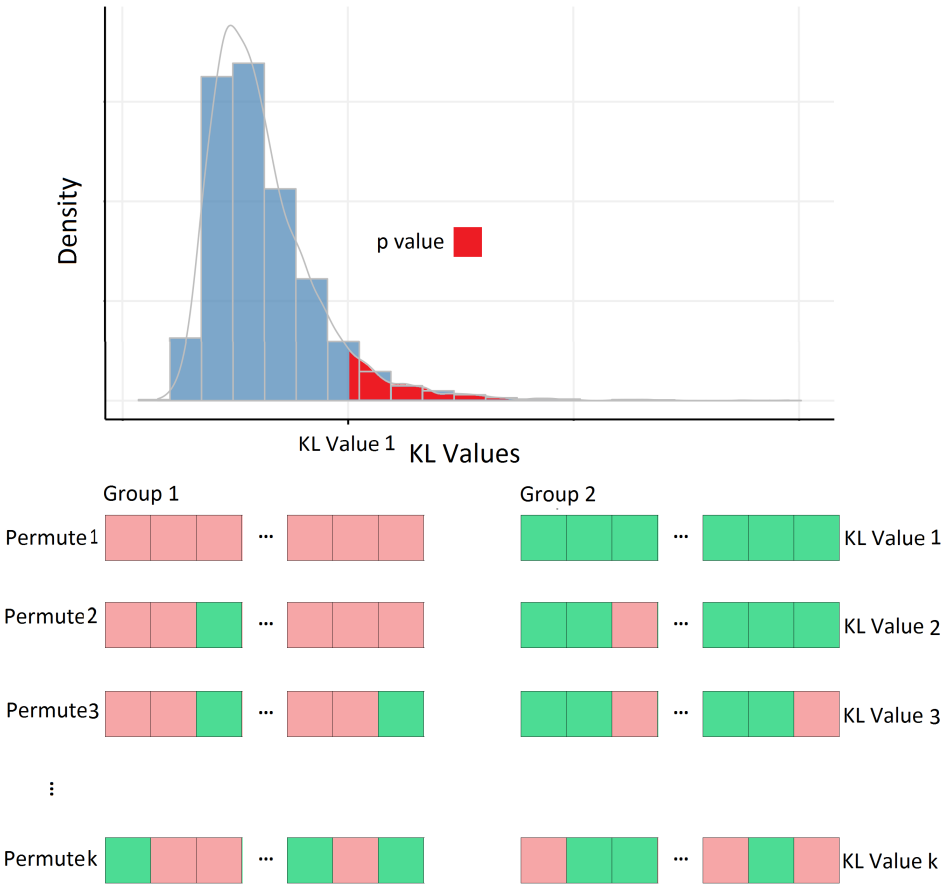


Figure 2: **The significance level of KL-divergence**. In order to find the significance level of a KL-divergence (KL Value 1) of a variable between two groups of samples (Permute 1), one can permute the samples between groups for k times and compute the KL-divergence between new couple of groups (Permute 2, Permute 3, ..., Permute k). Afterward, one can find the significance level of the KL-value by considering the empirical distribution of permuted KL-values as it is illustrated in the upper plot.

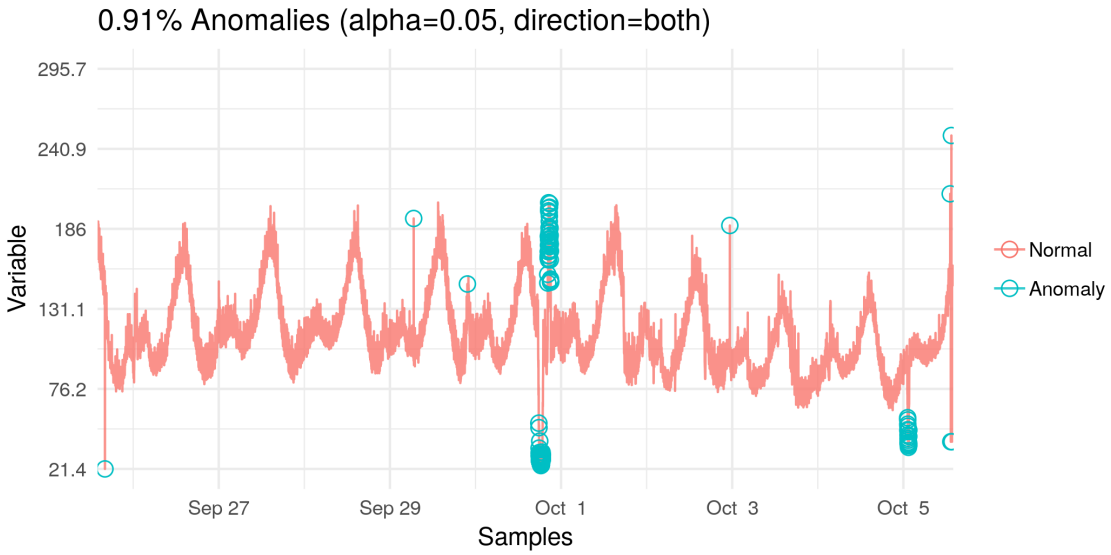


Figure 3: **Anomaly Detection by Vallis *et al.* (2014)**. The points in time-series data which do not follow the major trend of data are recognized as anomalies. The normal samples are colored in red and the anomalies are in blue

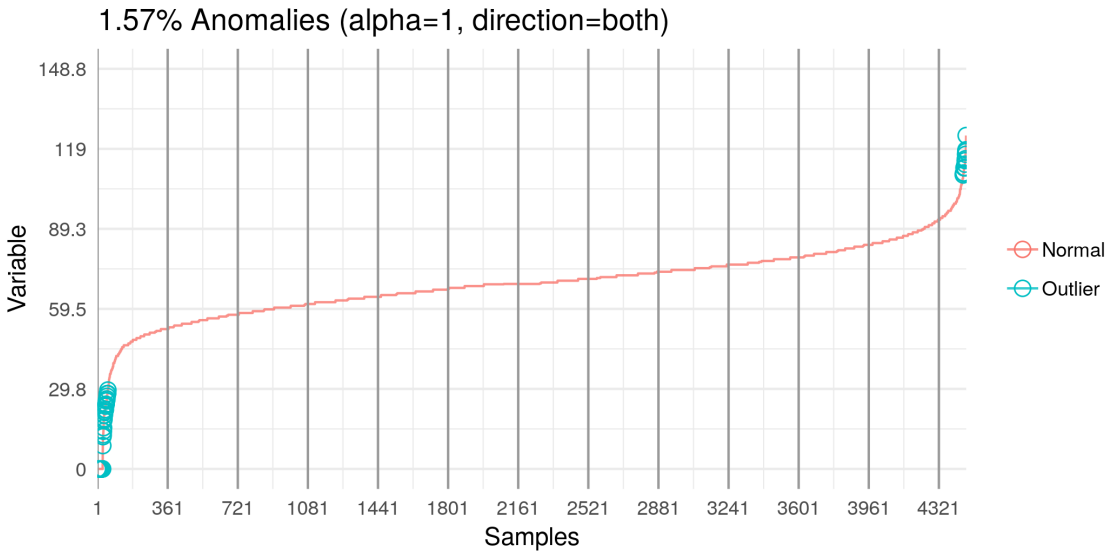


Figure 4: **Outlier Detection Algorithm**. After sorting data points in an increasing manner for the variable of interest, the outliers are detected after applying the anomaly detection algorithm (blue samples).

GM. Package **muvis** implements three types of GMs: (i) `ggm` implements five different methods (i.e., AIC-based, BIC-based, partial correlation thresholding, significance testing, and glasso) for GGM structure learning, based on R packages **gRim**, **glasso**, **SIN**, and **gRbase**. (ii) `dgm` is for constructing Directed (causal) GMs (DGMs) using **pcalg** package. (iii) `min_forest` is designed for High-dimensional Graphical Modeling and is based on its implementation in the package **gRapHD** (Edwards, de Abreu, and Labouriau (2010a)). For all of the mentioned methods, there is a parameter `community` which indicates if the user wants to apply community detection algorithms on the graph to find the modules within the graph. To this end, we used Louvain method from R package **igraph** (Csardi (2010); Blondel, Guillaume, Lambiotte, and Lefebvre (2008)). There is another parameter `plot` which is a logical parameter indicating if the user wants to plot the graph. The R package **qgraph** is used to implement this graph visualization (Epskamp, Cramer, Waldorp, Schmittmann, and Borsboom (2012)).

3.5. KL-based Functions

The functions `VKL` and `VVKL` implement the KL-methods. The parameter `permute` indicates the number of permutations as described in 2.2 and is used to find the significance of the KL values. We used the function `KL.plugin` from R package `entropy` to calculate KL values (Hausser and Strimmer (2014)). As this function works on discrete data, prior to calculating KL for continuous variables, we discretize and consider them like discrete data (see 4.7).

4. The NHANES 2005-2006 dataset

The National Health and Nutrition Examination Surveys (NHANES) (The United States Department of Health and Human Services. Centers for Disease Control and Prevention. National Center for Health Statistics (2012)) is a program of studies about health and nutrition for US residents. We examined the functionality of **muvis** on NHANES 2005-2006 dataset which contains 7449 variables and 10,348 samples (see `https://www.icpsr.umich.edu/icpsrweb/ICPSR/studies/25504/summary` for more details).
Based on the number of missing values, we selected 161 variables including one ID, 74 continuous, and 86 categorical variables (having two to fifteen levels) and 4461 individuals (samples) aged from 20 to 85 years, including about 1% missing values. See Table 1 for description of the variables that are mentioned in this article. Complete list of 161 variables is available in `NHANES` dataset of the package. The next parts describe the analysis of this dataset using **muvis**.

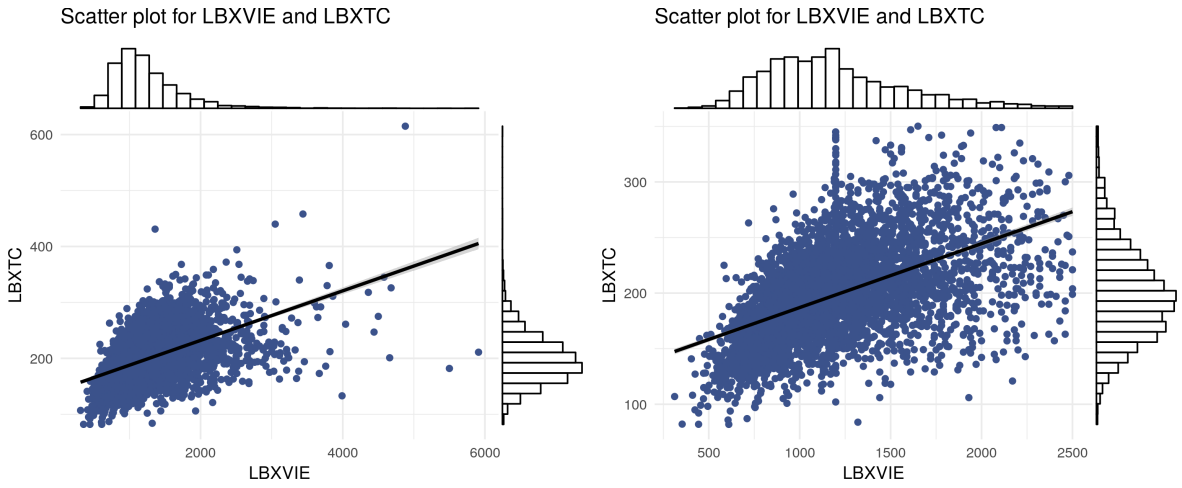
4.1. Loading the package and data

muvis is available at `https://github.com/bAIo-lab/muvis`. Once the package is installed and loaded into the R environment, the `NHANES` dataset can be loaded as a `dataframe` by a call to `data` as in the following.

```
> library(muvis)
> data("NHANES")
```

4.2. Preprocessing

We use `data_preproc` function for preprocessing of `NHANES`. The `detect.outliers` option is



(a) (b)

Figure 5: **Scatter plot** of `LBXTC` (total cholesterol. mg/dL) and `LBXVIE` (vitamin E. ug/dL) for NHANES dataset (a) with outliers and (b) without outliers. Each blue point indicates a sample and the black line shows the regression line fitted on two variables.

used to exclude outliers for each variable. As a first step through interpretation, we plot the relation between `LBXVIE` (a variable in the dataset indicating the amount of vitamin E) and `LBXTC` (the amount of total cholesterol) with and without outliers. The difference has been shown in Fig. 5.

```
> nhanes_with_outliers <- data_preproc(NHANES, levels = 15, alpha = 0.5)
> nhanes <- data_preproc(NHANES, levels = 15, detect.outliers = TRUE,
alpha = 0.5)
> plot_assoc(nhanes_with_outliers, vars = c("LBXVIE", "LBXTC"))
> plot_assoc(nhanes, vars = c("LBXVIE", "LBXTC"))
```

4.3. GGM for continuous data

We construct a GGM for continuous variables using `ggm` function. In this example, we construct it by intersecting `glasso` and `sin` algorithms. The largest connected component of the estimated graph is visualized by `graph_vis` function (Fig. 6).

```
> nhanes$SEQN <- NULL
> nhanes_ggm <- ggm(nhanes, significance = 0.05,
rho = 0.15, community = TRUE, methods = c("glasso", "sin"), plot = F)
> grph_clustrs <- clusters(nhanes_ggm$graph)
> new_ggm <- induced.subgraph(nhanes_ggm$graph,
V(nhanes_ggm$graph)[which(grph_clustrs$membership == which.max(grph_clustrs$csize))])
> ggm_vis <- graph_vis(new_ggm, plot = T,
filetype = "png", filename = "nhanes_ggm")
```

Investigating each community one can appraise the efficiency of the model: Community 1 are all about body measurements (e.g., BMI, waist circumstance, height, etc.); community 2 explains red blood cell profile; community 3 contains measurements about different types of

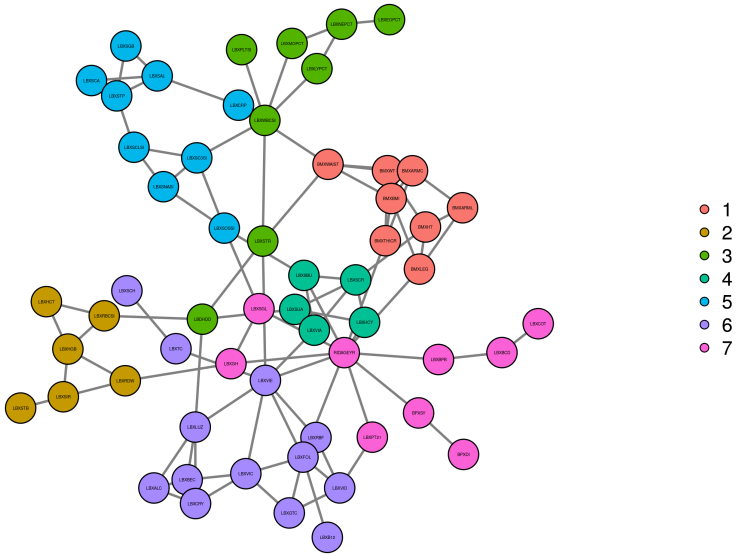


Figure 6: **GGM**. The graph is constructed using `ggm` function with the intersection of the models estimated by `sin` and `glasso` methods. The largest connected component of the graph is shown. The nodes represent continuous variables in the data and are colored according to their community.

white blood cells; community 4 accommodates uric acid, vitamin A, Urea, Creatinine, and Homocysteine; community 5 includes variables describing body biochemistry profile; community 6 contains some vitamins and chemical compounds (e.g. carotenoids, folate, etc.); and finally community 7 contains age, blood pressure, lead, and Parathyroid hormone.

4.4. Causal network for continuous data

The causal (directed) network of continuous variables is constructed using `dgm` function with parameter `dtype = "gaussian"`. The largest connected component of the estimated graph is shown in Fig. 7.

```
> nhanes_dgm <- dgm(nhanes, dtype = "gaussian", alpha = 1e-15)
> grph_clustrs <- clusters(nhanes_dgm$graph)
> new_dgm <- induced.subgraph(nhanes_dgm$graph,
V(nhanes_dgm$graph)[which(grph_clustrs$membership == which.max(grph_clustrs$csize))])
> dgm_vis <- graph_vis(new_dgm, plot = T, directed = T, filename = "nhanes_dgm",
filetype = "png")
```

4.5. Minimal forest for mixed data

Using `min_forest` function we estimate the minimal forest with `BIC` method and detect communities in the graph. The estimated minimal forest and some of its communities are illustrated in Fig. 8.

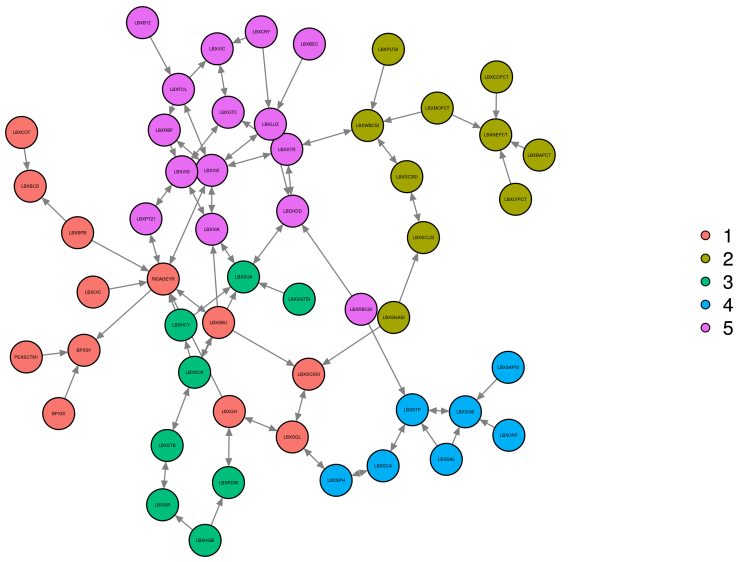


Figure 7: **Causal graph**. The largest connected component of the graph is shown. Nodes indicate variables in the dataset and are colored based on their community number. Edges are directed based on the estimated causal relationship so that each edge is directed from cause to effect.

```
> nhanes_mf <- min_forest(nhanes, stat = "BIC", community = T, plot = F)
```

```
> mf_vis <- graph_vis(nhanes_mf$graph, plot = T, filetype = "png",
filename =  "nhanes_mf_bic", plot.community = T)
```

4.6. Community analysis of the minimal forest

In the following section, we inspect communities 1, 3, 5, 6, 7, and 8 of the minimal forest graph to demonstrate the functionally of the algorithm.

Most of the nodes in community 1 (Fig. 8b) are related to red blood cells and body biochemistry profile. There is an association between `LBDHDD` (direct HDL-cholesterol) and `RIAGENDR` (gender) in this community, shown in Fig. 9a.

```
> signif <- nhanes_mf$significance
> signif[signif$edges.from == 'RIAGENDR' & signif$edges.to == 'LBDHDD', ]
```

```
   edges.from edges.to statistics      p.value
   RIAGENDR   LBDHDD   619.5036 1.737264e-13
```

```
> plot_assoc(nhanes, vars = c("LBDHDD", "RIAGENDR"))
```

Fig. 10a shows another relation in community 1, which is between nodes `LBXHGB` (hemoglobin) and `LBXSIR` (refrigerated iron in blood). Community 3 (Fig. 8c) mostly includes chemical compounds and vitamins (i.e. E, C, B12, and D). As an example, the scatter plot of two

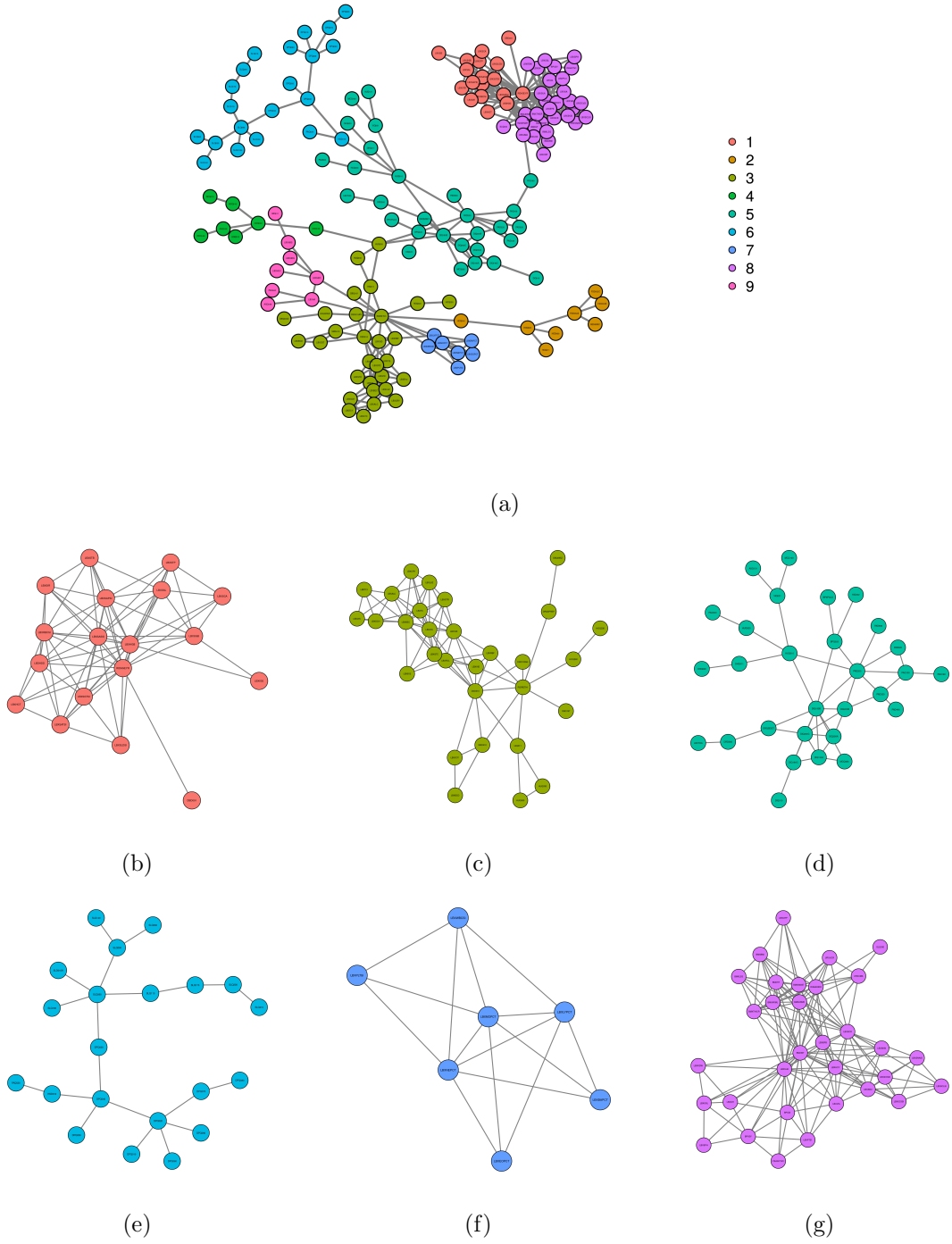


Figure 8: **Minimal forest**. (a) The graph is constructed using `min_forest` function with `BIC` method. The nodes indicate variables in the dataset and are colored with respect to their community. (b) community 1: red blood and body biochemistry profile. (c) community 3: chemical compounds and vitamins. (d) community 5: physical activity, health condition, and diabetes. (e) community 6: depression and sleeping status. (f) community 7: white blood cells. (g) community 8: body measurements and blood pressure.

variables LBXVIE (vitamin E) and LBXTC (totoal cholesterol) is illustrated in Fig. 10b. There are also two connected nodes SMD410 (smoking in household) and LBXCOT (cotinine) in this community that their relation is shown in Fig. 9c. Nodes in community 5 (Fig. 8d) indicate three groups of variables, namely, physical activity, health condition, and diabetes. There is a chain relation that connects DRD360 (eating fish in past 30 days) to diabetes group through node DRQSDIET (being on a special diet). DRD360 is also connected to LBXTHG (total mercury in blood). The relation is plotted in Fig. 9b. DPQ030 (trouble with sleeping) connects sleep disorder nodes to variables about depression in community 6 (Fig. 8e). Community 7 (Fig. 8f) consists of different types of white blood cells. Body measurement variables and blood pressure are located in community 8 (Fig. 8g).

4.7. Variable-wise KL-divergence

Focusing on the variable PAD590 (TV usage) we select two groups of samples: (i) The participants who watch TV less than an hour (g1) and (ii) those who watch more than 5 hours (g2) a day . We use VKL function with permute = 1000 so as to find the most different features between these two groups. In the following code, we get five variables with the highest KL-divergence values excluding PAD590.

```
> g1 <- which(nhanes$PAD590 == 1)
> g2 <- which(nhanes$PAD590 == 6)
> KL <- VKL(nhanes, group1 = g1, group2 = g2, permute = 1000)
> KL[2:6, ]
```

```
               KL variable p.value
HSD010   0.2581593   HSD010   0.001
PAQ520   0.2193065   PAQ520   0.001
HUQ010   0.2157753   HUQ010   0.001
RIDAGEYR 0.2094270 RIDAGEYR   0.001
LBXALC   0.2066657   LBXALC   0.001
```

HSD010 (general health condition) is the most different variable between g1 and g2 based on KL-divergence values. See Table 1 for description of the other variables.

4.8. Violating Variable-wise Kulback-Leibler Divergence

As Fig. 10b shows, there is an approximately linear relationship between vitamin E and total cholesterol. Using VVKL function with permute = 100, we find the most important (categorical) variables violating such linear relationship. The result suggests DSD010 (taking dietary supplements) and DRQSPREP (type of table salt used) as variables with the highest KL-divergence. Two groups of samples with the highest absolute residual values (top 5 %) with respect ot the fitted line are remarked as outliers and highlighted in Fig. 11.

```
>  KL <- VVKL(nhanes[, 75:160], var1 = nhanes$LBXVIE, var2 = nhanes$LBXTC,
plot = T, var1.name = "LBXVIE", var2.name = "LBXTC", permute = 100)
> head(KL$kl)
> KL$plot
```

```
               KL variable p.value
DSD010   0.1485264   DSD010    0.01
DRQSPREP 0.1466014 DRQSPREP    0.01
```

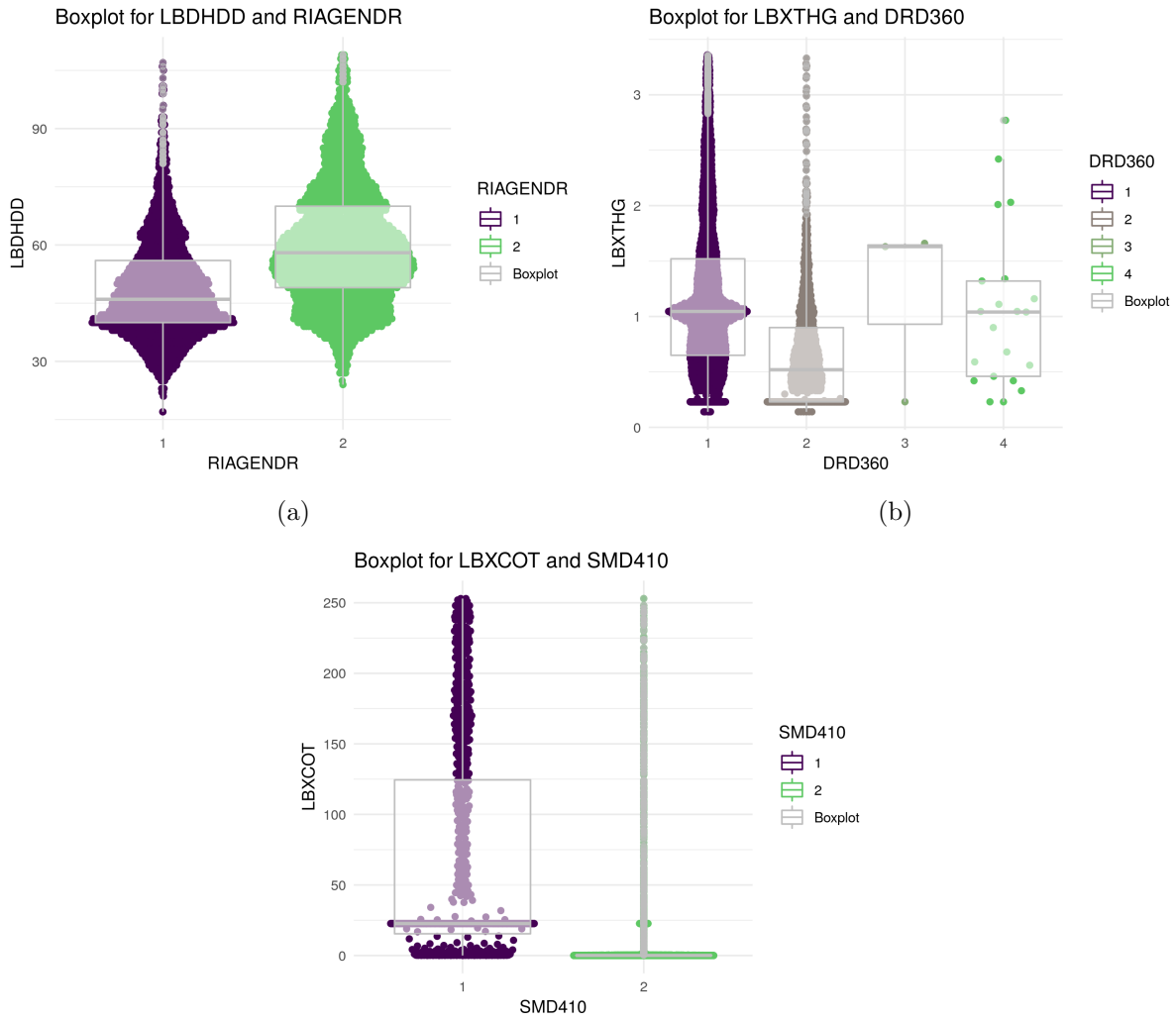
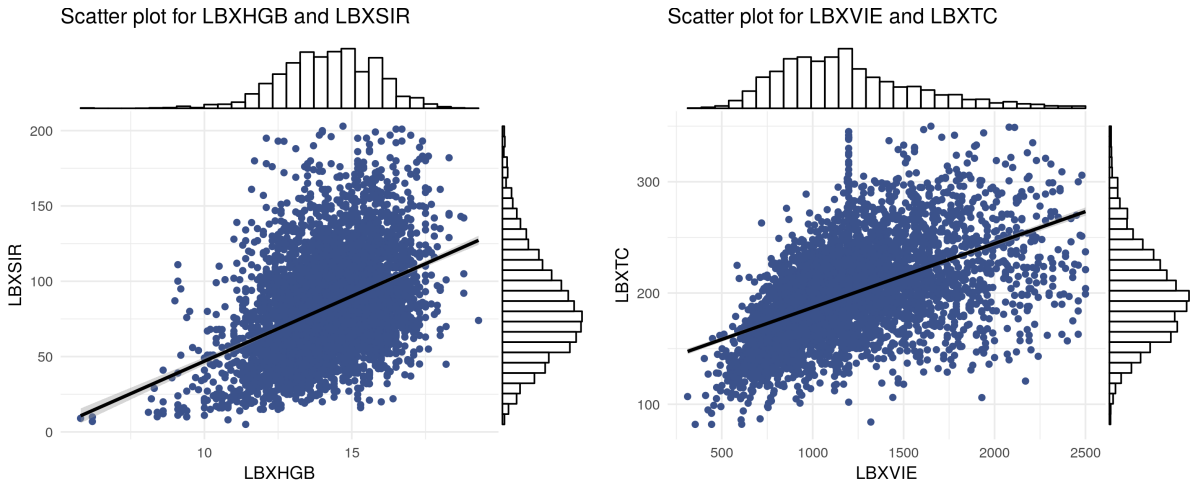


Figure 9: **Boxplot** of (a) `LBDHDD` (direct HDL-cholesterol. mg/dL) and `RIAGENDR` (gender. 1 for Male and 2 for Female), (b) `LBXTHG` (total mercury in the blood. ug/L) and `DRD360` (eating fish in past 30 days. 1 for Yes, 2 for No, 3 for Refused and 4 for Don't know), (c) `LBXCOT` (cotinine. ng/mL) and `SMD410` (smoking in household. 1 for Yes and 2 for No). The plots are colored according to the different levels of their categorical variable. points in these plots represent samples in the data.



(a) (b)

Figure 10: **Scatter plot** of (a) `LBXHGB` (hemoglobin. g/dL) and `LBXSIR` (refrigerated iron in blood. ug/dL), (b) `LBXTC` (total cholesterol. mg/dL) and `LBXVIE` (vitamin E. ug/dL). Each point in the plots indicates a sample in the data.

```
PAD440   0.1273735   PAD440    0.01
BPQ020   0.1253917   BPQ020    0.01
RIAGENDR 0.1169647 RIAGENDR    0.01
DBQ095Z  0.1115717  DBQ095Z    0.01
```

4.9. Clustering with minimal forest

In order to show the functionality of minimal forest in clustering, we select a subsample of size 200 and apply `min_forest` function on the subsampled dataset. Therefore the transposed dataset is passed to the funciton. Fig. 12 illustrates the stratified population of the samples.

```
> t_nhanes <- as.data.frame(sapply(as.data.frame(t(nhanes[1:200, ])),
function(x) as.numeric(as.character(x))))
> clusters_mf <- min_forest(t_nhanes)
> clusters_vis <- graph_vis(clusters_mf$graph, plot = T,
filename = "clusters", filetype = "png")
```

Dimensionality reduction algorithms can be used in order to get a proper visualization of data in low (i.e., two or three) dimensions. We use `dim_reduce` function with `tsne` and `umap` methods to plot the aforementioned subpopulation in two dimensions. The points are colored according to their communities within the minimal forest in Fig. 12. As one can see in Fig. 13 the samples within each community are clustered together, in a fairly accurate manner, particularly with UMAP.

```
> communities <- clusters_mf$communities
> communities <- communities[match(c(1:200), as.integer(names(communities)))]

## Using 'umap' method
```

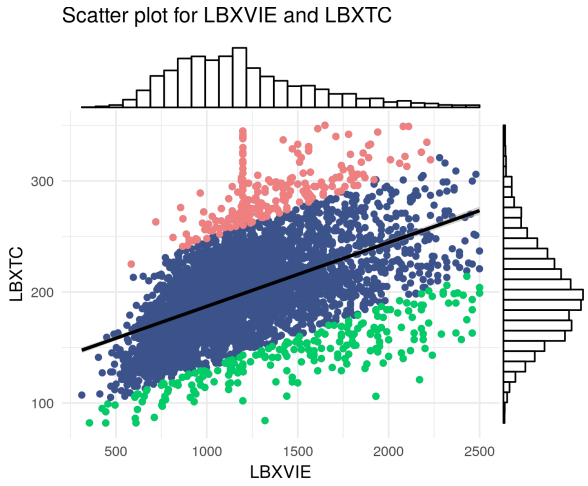


Figure 11: **Scatter plot** of `LBXTC` (total cholesterol. mg/dL) and `LBXVIE` (vitamin E. ug/dL). A line is fitted to the variables and the samples with absolute residual values within the highest 5 percentile are considered as the outliers. The two outlier groups are colored with red and green colors.

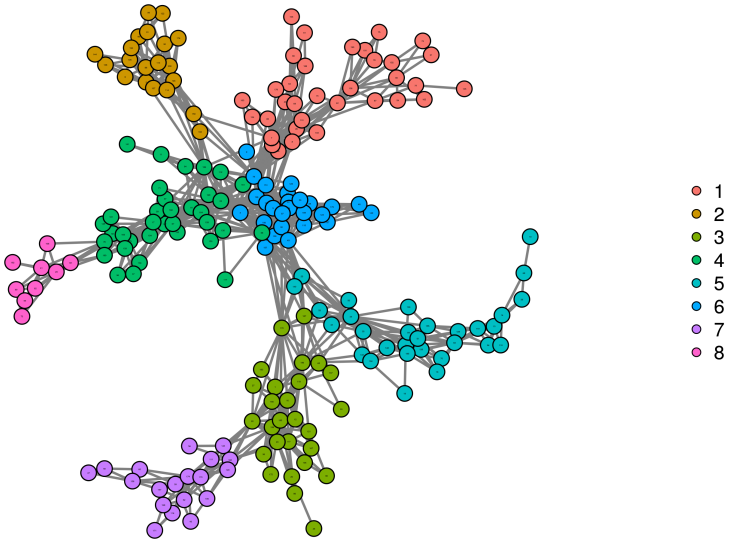


Figure 12: **Population stratification (clustering) with minimal forest**. The graph is constructed using `min.forest` function with `BIC` method on the transposed dataset. The nodes represent samples and are colored according to the graph communities.

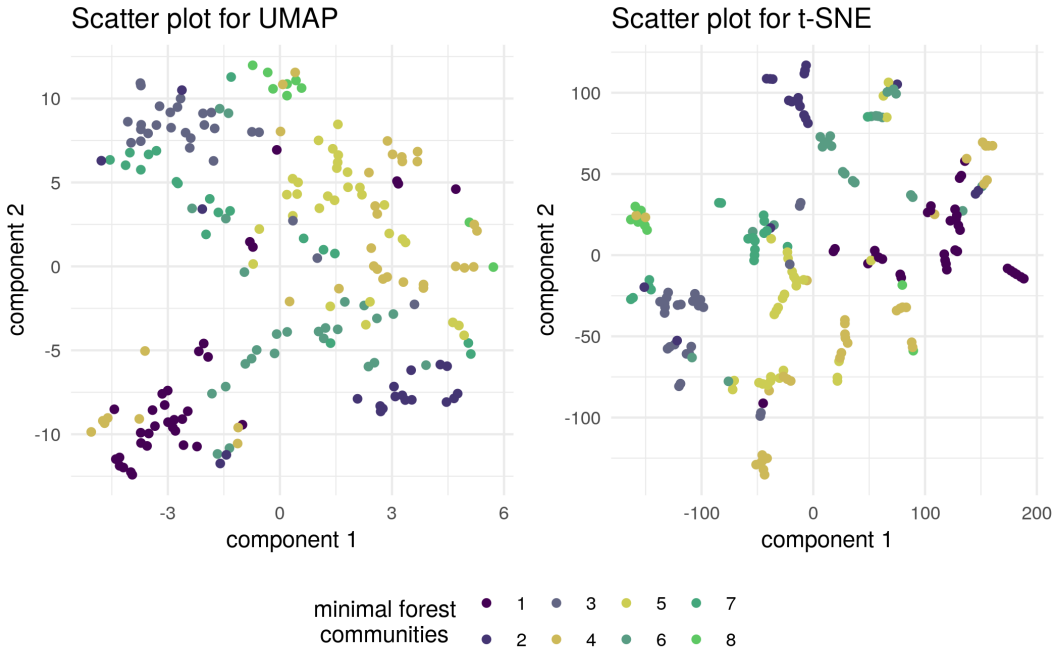


Figure 13: **Two-dimensional visualization of the subsampled dataset**. The funciton `dim_reduce` is used with parameter `method` set to `umap` (on left) and `tsne` (on right) to perform dimensionality reduction. The nodes are colored based on their community in the minimal forest illustrated in Fig. 12.

```
> ump <- dim_reduce(nhanes[1:200,], method = "umap", annot1 = as.factor(communities)
, annot1.name = "minimal forest\n communities")

## Using 'tsne' method
> tsn <- dim_reduce(nhanes[1:200,], method = "tsne", annot1 = as.factor(communities)
, annot1.name = "minimal forest\n communities")

## Using cowplot to plot with shared legend
> require(cowplot)
> require(ggplot2)
> leg <- get_legend(ump + theme(legend.position = "bottom"))
> plt <- plot_grid(ump + theme(legend.position = "none"),
tsn + theme(legend.position = "none"))
> plot_grid(plt, leg, ncol = 1, rel_heights = c(1, .2))
```

5. Conclusions

In this document, we introduced **muvis** as a set of tools for analysis and visualization of multivariate data. The package provides the users with an easy-to-use and end-to-end analysis pipeline. The methods implemented in **muvis** can be applied on a wide range of datasets with high number of variables and observations. It can be used for data preprocessing

(`data_preproc`), statistical analysis (`VKL`, `VVKL`, `test_assoc`, `ggm`, `dgm`, and `min_forest`), and visualization (`plot_assoc`, `graph_vis`, and `dim_reduce`). As demonstrated in section 4, the results assert the functionality of the package on real data: preprocesing method could effectively detect and eliminate the outliers; the GMs were efficient to simultaneously estimate the structure of associations and interpretations; and the visualization method could easily visualize associations among variables. We also introduced novel KL-based methods for determining important variables that could explain surprising observations, including violation of expected linear associations. This work can be extended in several directions: providing predictive models (e.g. elastic nets, ensemble methods, etc.), non-Gaussian GMs, and enhanced imputation methods.

6. License

Table 1: Description of mentioned variables.

Variable	**Label**
SEQN	Respondent sequence number
LBDHDD	Direct HDL-Cholesterol (mg/dL)
DRQSDIET	On special diet?
DRQSPREP	Salt used in preparation?
LBXVIC	Vitamin C (mg/dL)
LBXALC	Alpha-carotene (ug/dL)
DRD360	Fish eaten during past 30 days
LBXVIE	Vitamin E (ug/dL)
PAD590	# hours watch TV or videos past 30 days
DSD010	Any Dietary Supplements Taken?
LBXFOL	Folate, serum (ng/mL)
PAD440	Muscle strengthening activities
PAQ100	Tasks around home/yard past 30 days
PAD200	Vigorous activity over past 30 days
HUQ050	# times receive healthcare over past year
LBXSATSI	Alanine aminotransferase ALT (U/L)
LBXSAL	Albumin (g/dL)
PAQ180	Avg level of physical activity each day
LBXSCH	Cholesterol (mg/dL)
SMD410	Does anyone smoke in home?
LBXHCT	Hematocrit (%)
LBXHGB	Hemoglobin (g/dL)
LBXSIR	Iron, refigerated (ug/dL)
LBXMOPCT	Monocyte percent (%)
LBXRBCSI	Red blood cell count (million cells/uL)
LBXRDW	Red cell distribution width (%)
LBXSTR	Triglycerides (mg/dL)
LBXSUA	Uric acid (mg/dL)
LBXWBCSI	White blood cell count (1000 cells/uL)
BMXBMI	Body Mass Index (kg/m**2)
LBXBCD	Cadmium (ug/L)
LBXCOT	Cotinine (ng/mL)
BPQ020	Ever told you had high blood pressure
HUQ010	General health condition
LBXTHG	Mercury, total (ug/L)
LBXTC	Total Cholesterol (mg/dL)
RIDRETH1	Race/Ethnicity
RIAGENDR	Gender
PAQ520	Compare activity w/others same age
HSD010	General health condition
DBQ095Z	Type of table salt used

References

Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E (2008). "Fast unfolding of communities in large networks." *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10). ISSN 17425468. doi:10.1088/1742-5468/2008/10/P10008. 0803.0476.

Burnham KP, Anderson DR, Huyvaert KP (2010). "AIC model selection and multimodel inference in behavioral\necology: some background, observations, and comparisons." *Behavioral ecology and sociobiology*, **65**(1), 23–35. ISSN 0340-5443.

Chow CK, Liu CN (1968). "Approximating Discrete Probability Distributions with Dependence Trees." *IEEE Transactions on Information Theory*, **14**(3), 462–467. ISSN 15579654. doi:10.1109/TIT.1968.1054142.

Claeskens G, Hjort NL (2008). *Model selection and model averaging*. ISBN 978-0-521-85225-8. doi:10.1080/02664760902899774.

Coghlan A (2014). "A little book of R for multivariate analysis. Release 0.1." *Wellcome Trust Sanger Institute, Cambridge, UK*.

Cohen P, West SG, Aiken LS (2014). *Applied multiple regression/correlation analysis for the behavioral sciences*. Psychology Press.

core team RR (2017). "R: A language and environment for statistical computing." *R Foundation for Statistical Computing, Vienna, Austria*. ISSN 16000706. doi:http://www.R-project.org/. /www.R-project.org.

Csardi G (2010). "Package igraph." *Cran*, pp. 1–187. doi:10.1177/001316446902900315.

Drton M, Perlman MD (2007). "Multiple testing and error control in Gaussian graphical model selection." *Statistical Science*, **22**(3), 430–449 ST – Multiple testing and error control i. ISSN 0883-4237. doi:10.1214/088342307000000113. 0508267v3.

Edwards D, de Abreu GC, Labouriau R (2010a). "High-dimensional Graphical Model Search with gRapHD R Package." *Journal of Statistical Software*, **37**(1), 20. ISSN 15487660. doi:http://dx.doi.org/10.18637/jss.v037.i01. 0909.1234.

Edwards D, de Abreu GC, Labouriau R (2010b). "Selecting high-dimensional mixed graphical models using minimal AIC or BIC forests." *BMC Bioinformatics*, **11**. ISSN 14712105. doi:10.1186/1471-2105-11-18.

Epskamp S, Cramer AOJ, Waldorp LJ, Schmittmann VD, Borsboom D (2012). "qgraph : Network Visualizations of Relationships in Psychometric Data." *Journal of Statistical Software*. doi:10.18637/jss.v048.i04.

Esbensen KH, Guyot D, Westad F, Houmoller LP (2002). *Multivariate data analysis: in practice: an introduction to multivariate data analysis and experimental design*. Multivariate Data Analysis.

Everitt B, Hothorn T (2011). *An Introduction to Applied Multivariate Analysis with R*. Springer New York, New York, NY. ISBN 978-1-4419-9649-7. doi:10.1007/978-1-4419-9650-3. URL http://link.springer.com/10.1007/978-1-4419-9650-3.

Friedman J, Hastie T, Tibshirani R (2008). "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, **9**(3), 432–441. ISSN 14654644. doi:10.1093/biostatistics/kxm045. 0708.3517.

Hausser J, Strimmer K (2014). "entropy: Estimation of Entropy, Mutual Information and Related Quantities." *R Environment*, pp. https://cran.r–project.org/web/packages/entropy/in.

Højsgaard S, Edwards D, Lauritzen S (2012). "Graphical Models with R." *Graphical Models*, pp. 27–50. ISSN 1461422981. doi:10.1007/978-1-4614-2299-0. arXiv:1011.1669v3, URL http://www.springerlink.com/index/10.1007/978-1-4614-2299-0.

Kalisch AM, Maechler M, Colombo D, Kalisch MM (2010). "Package âĂŸ pcalg '." *Discovery*. 1307.5636.

Kalisch M, Hauser A, Maathuis M, Mächler M (2018). "An Overview of the pcalg Package for R."

Kalisch M, Martin M (2010). "Journal of Statistical Software Causal Inference using Graphical Models with the R Package pcalg." *Journal Of Statistical Software*, **VV**(Ii), 189–204. URL http://cran.r-project.org/web/packages/pcalg/vignettes/pcalgDoc.pdf.

Koller D, Friedman N, Getoor L, Taskar B (2007). "Graphical Models in a Nutshell." *Introduction to Statistical Relational Learning*, p. 43. doi:10.1.1.146.2935. URL http://www.robotics.stanford.edu/{~}koller/Papers/Koller+al:SRL07.pdf.

Kruskal JB (1956). "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem." *Proceedings of the American Mathematical Society*, **7**(1), 48. ISSN 00029939. doi:10.2307/2033241. arXiv:1011.1669v3, URL http://www.jstor.org/stable/2033241?origin=crossref.

Kullback S, Leibler RA (1951). "On information and sufficiency." *The annals of mathematical statistics*, **22**(1), 79–86.

Kunst J (2017). *highcharter: A Wrapper for the 'Highcharts' Library*. R package version 0.5.0, URL http://jkunst.com/highcharter.

Lauritzen SL (1996). "Graphical Models." *Graphical Models*.

Lê S, Josse J, Husson F (2008). "FactoMineR : An R Package for Multivariate Analysis." *Journal of Statistical Software*. ISSN 1548-7660. doi:10.18637/jss.v025.i01. arXiv:0908.3817v2.

Rue H (2005). "Gaussian Markov Random Fields: Theory and Applications." *Hand The*, **104**(1960), 263 p. ISSN 0026-1335. doi:10.1007/s00184-007-0162-3. URL http://www.amazon.com/dp/1584884320.

The United States Department of Health and Human Services Centers for Disease Control and Prevention National Center for Health Statistics (2012). "National Health and Nutrition Examination Survey (NHANES), 2005-2006." http://doi.org/10.3886/ICPSR25504.v5. Accessed 3 August 2018.

Timm NH (ed.) (2004). *Applied Multivariate Analysis*. Springer Texts in Statistics. Springer New York, New York, NY. ISBN 978-0-387-95347-2. doi:10.1007/b98963. URL http://link.springer.com/10.1007/b98963.

Tsagris M (2016). *Multivariate data analysis in R. A collection of R functions for multivariate data analysis v9.1*. Department of Computer Science, University of Crete, Herakleion.

Uhler C (2017). "Gaussian Graphical Models: An Algebraic and Geometric Perspective." *arXiv preprint arXiv:1707.04345*.

Vallis O, Hochenbaum J, Kejariwal A (2014). "A Novel Technique for Long-term Anomaly Detection in the Cloud." *Proccedings of the 6th USENIX Workshop on Hot Topics in Cloud Computing (USENIX '14)*, pp. 1–6.

Véron J, Rohrbasser JM (2003). "Wilhelm Lexis: the normal length of life as an expression of the "nature of things"." *Population*, **58**(3), 303–322. ISSN 0032-4663. doi:10.2307/3271331.

Wasserman L (2013). *All of statistics: a concise course in statistical inference*. Springer Science & Business Media.

Wickham H (2010). "ggplot2: elegant graphics for data analysis." *J Stat Softw*, **35**(1), 65–88.