

# Manual

*Jolanda J. Kossakowski, BSc*

*17 August 2015*

## Contents

<b>Introduction</b>	<b>1</b>
<b>Uploading Data</b>	<b>1</b>
<b>Network Analysis</b>	<b>3</b>
<b>Network Esthetics</b>	<b>7</b>
<b>Centrality Analyses</b>	<b>7</b>
<b>Clustering Analyses</b>	<b>7</b>
<b>Network Comparison</b>	<b>7</b>
<b>References</b>	<b>7</b>

## Introduction

Network analysis has gained popularity in recent years. By the development of R-packages like *qgraph* (Epskamp et al. 2012) and *igraph* (Csardi and Nepusz 2006), it has become easier to visualize network structures and apply different estimation methods. The drawback of this is that users have to learn R in order to use these packages. The goal of this application was to design and programme a web application that enables users to visualize and analyze networks without having to know the R-language itself. This manual will aid users and explain all the functions and options that are currently available within the application.

## Uploading Data

In the grey left panel of the application, users can upload either a “.txt” or a “.csv” file. Users only have to specify the type of data that they are uploading, how missing values, if present, are coded within the data file, and whether character variables (variables that contain words instead of numbers) should be coded as *factors*. In the case that the user discriminated between different scores by using words, this option should be checked.

☐ **Strings as factors**

## Data Types

**Specify the kind of data that is uploaded:**

Raw Data ▲

Raw Data

Adjacency Matrix

Edgelist

The user can enter three different types of data. The first option is “raw data”. When you have raw data, your datafile consists of columns, which indicate the individual items or variables, and rows that signify each individual participant or measurement of those items or variables. Below is a simplified example of what raw data looks like.

##	participant	Item1	Item2
## 1	1	1	2
## 2	2	5	1
## 3	3	1	5
## 4	4	7	2
## 5	5	6	3
## 6	6	2	4
## 7	7	2	3
## 8	8	5	3
## 9	9	7	4
## 10	10	6	3

The next available option as a data type is the “adjacency matrix”. When you use an adjacency matrix, the data will have the individual items or variables on both the rows and the columns, and each cell indicate whether there is a relation between the two variables, and if so, how strong this relation is. To illustrate, let us take the following adjacency matrix:

##	Item1	Item2	Item3	Item4
## 1	0	1	2	1
## 2	1	0	3	2
## 3	2	3	0	1
## 4	1	2	1	0

In this example, it can be seen that there are four variables, as there are four rows and columns. Between variable 1 and 2, a relation exists with strength 1. In this example, the diagonal consists of zeroes, which is not obligatory; it is possible to construct a network with so-called *self-loops*: edges that start and finish at the same node without the interference of some other node. Furthermore, the example above depicts a symmetrical matrix, but this does not have to be the case: networks can be constructed in which the relation from A to B is different from the relation from B to A.

The last available data type is the “edge list”. An edge list is characterized by two columns that depict the starting and ending node for each individual edge, which are represented by the rows. Take for example the edge list below:

```
##   from to
## 1    1  2
## 2    2  3
## 3    3  4
## 4    4  5
## 5    5  1
```

In this example, an edge originates in node 1, and ends in node 2, without the interference of some other node.

In the examples that were previously used, a header existed that shows the variable name for each column. Note that this may help in the identification of nodes when a network is constructed, but that a header is not necessary in order to construct a network.


## Missing Data

After you have specified your data, you can specify the symbol used for missing data, if these exist in your data file. If you do not have any missing data, you can leave this box empty.

**Missing value coding:**

## Example Data

If you do not have any data (yet) that can be used to construct networks, you can download an example data set, which is a raw data file that contains the first 25 variables of the NEO-PI-R. This data is made available via the *psych* package.

 **Download Example Data**

## Network Analysis

After entering all specifications with respect to the data, a network is automatically constructed and displayed on the right, in the tab *Network*:

## Network App

The options below are needed to specify how your file looks like. If you do not have any data but you want to see how the application works, click "Demo version" and a dataset is automatically updated that is available via the psych package. The example dataset comprises of 25 NEO-PI-R items: 5 items per trait.

**Choose CSV or TXT File**  
 no file selected

☒ Demo Version

**Specify the kind of data that is uploaded:**  

Raw Data

☐ Strings as factors

**Missing value coding:**

Network

Centrality Plot

Centrality Table

Clustering Plot

Clustering Table

Network Comparison

This network is constructed by means of the default values that exist. Almost everything can be changed with respect to the network esthetics, and the estimation of the network. I will explain all options in more detail below.

### Network Estimation

There are various methods for estimating a network structure. The NetworkApp has employed 7 methods that are used often. Note that the estimation methods only with when you use raw data. As the absence/presence of edges and/or their strength is already specified in an adjacency matrix or edge list, the estimation methods become redundant.

### Network Estimation Method:

Partial Correlation

FDRnetwork

GLASSO

IC-Algorithm: DAG

IC-Algorithm: Skeleton

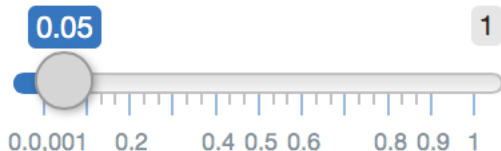
Partial Correlation

Pearson Correlation

VAR-model

**FDR Network** The *FDR Network* estimation method estimates a network based on the False Discovery Rate (FDR). By setting a cutoff value between 0 and 1, edges with an FDR higher than the cutoff value will be removed from the network.

### Cut-off value FDR network:



There are three different methods that can be used to create an FDR network:

### Method for FDR Network:

None

Local FDR
None
p-value
q-value

One can use a local false discovery rate, the  $p$ -value, or the  $q$ -value. When estimating an FDR network, one is obliged to specify the method and a cut-off value.

**GLASSO** The *GLASSO* estimation method is used to control for spurious correlations that may arise due to multiple testing. To create a sparse network, the EBICglasso, which uses penalized maximum likelihood estimation, is applied to set spurious connections to zero. The result is a network in which connections between nodes represent a nonzero partial correlation, while controlling for all other nodes in the network (Kossakowski et al. 2015).

No further action is required of the user to construct a GLASSO network: by selecting GLASSO, the application will automatically construct this type of network.

**IC-Algorithm: DAG** The *IC-Algorithm: DAG* estimation method can be used to construct a *Directed Acyclic Graph* (DAG) by using the *Inferred Causation (IC) Algorithm* (Pearl 2009). This DAG is created by means of the *pcalg* package (Kalisch et al. 2012). This function estimates the equivalence class of a DAG, which means that multiple networks can describe the same set of conditional independencies. More specifically, a *Completed Partially Directed Acyclic Graph* (CPDAG) is estimated, which is a network that may contain both directed and undirected edges. A directed edges is drawn when this edge is present in all DAGS in the equivalence class, and an undirected edge is drawn when at least one DAG contains the edge with direction  $a \rightarrow b$ , and at least one DAG that contains the same edge, but with direction  $b \rightarrow a$ .

When the user wants to estimate a DAG, one also has to specify which method is to be used to test the conditional independencies:

## Method for Conditional Independence

### IC-Algorithm:

None

Binary

Discrete

D-separation

Gaussian

None

The choice of method mainly depends on the data: If the data contains Gaussian data, the user would choose Gaussian. The same holds for Binary data and Discrete (multinomial) data. D-separation is a special method that states that two variables are conditionally independent controlling for a third variable if that third variable blocks ALL paths between the initial two variables.

### IC-Algorithm: Skeleton

### Partial Correlation

### Pearson Correlation

### VAR-model

### Small World Index

The *Small World Index* (SWI) is an index that tells you whether a network is in fact a small world or not. A Small World Network is a network in which the average distance between two nodes resembles that of a Random Network, and where the average ratio between the amount of existing edges and possible edges (*clustering*) is high in comparison to a Random Network (Watts and Strogatz 1998).

### The Small-World Index of this network is

[1] 1

A network is considered a Small World Network when the SWI is higher or equal to 3. The SWI is automatically calculated for each network that is constructed, and will be re-calculated when parameters are changed.

## Download Network

Each network that is constructed can be download as a .pdf file by clicking on the corresponding button:



## Network Esthetics

## Centrality Analyses

## Clustering Analyses

## Network Comparison

## References

Csardi, Gabor, and Tamas Nepusz. 2006. "The Igraph Software Package for Complex Network Research." *InterJournal, Complex Systems* 1695: 1–9. <http://igraph.org>.

Epskamp, Sacha, Angélique O. J. Cramer, Lourens J. Waldorp, Verena D. Schmittmann, and Denny Borsboom. 2012. "qgraph: Network Visualizations of Relationships in Psychometric Data." *Journal of Statistical Software* 48: 1–18. <http://www.jstatsoft.org/v48/i04/>.

Kalisch, Markus, Martin Mächler, Diego Colombo, Marloes H Maathuis, and Peter Bühlmann. 2012. "Causal Inference Using Graphical Models with the R Package Pcalg." *Journal of Statistical Software* 47 (11): 1–26.

Kossakowski, Jolanda J, Sacha Epskamp, Jacobien M Kieffer, Claudia D van Borkulo, Mijke Rhemtulla, and Denny Borsboom. 2015. "The Application of a Network Approach to Health-Related Quality of Life (HRQoL): Introducing a New Method for Assessing HRQoL in Healthy Adults and Cancer Patient."

Pearl, Judea. 2009. *Causality*. Cambridge university press.

Watts, Duncan J, and Steven H Strogatz. 1998. "Collective Dynamics of 'Small-World' Networks." *Nature* 393. Nature Publishing Group: 440–42.