Blauth, Pigeot:

GraphFitI - A computer program for graphical chain models

Sonderforschungsbereich 386, Paper 224 (2000)

Online unter: http://epub.ub.uni-muenchen.de/

GraphFitI – A computer program for graphical chain models

Angelika Blauth and Iris Pigeot
Department of Statistics, University of Munich, Ludwigstrasse 33, D–80539 Munich, Germany
blauth@stat.uni-muenchen.de, pigeot@stat.uni-muenchen.de

Summary

Fitting a graphical chain model to a multivariate data set consists of different steps some of which being rather tedious. The paper outlines the basic features and overall architecture of the computer program GraphFitI which provides the application of a selection strategy for fitting graphical chain models and for visualising the resulting models as a graph. It additionally supports the user at the different steps of the analysis by an integrated help system.

Keywords: *Chain graph; Conditional independence; Selection strategy; Software*

1 Introduction

Modern technologies make it nowadays impressively easy to collect huge amounts of data in course of an empirical study which results in the challenging demand to extract the substantial information. This task asks for adequate computational and statistical solutions allowing for an efficient and most informative analysis.

Such multivariate data sets usually imply a complex association structure which cannot be analysed in a straightforward manner based on standard statistical software without running the risk of loosing important information. Here, it seems to be a sensible approach to combine substantial knowledge about the underlying structure with sophisticated statistical methods. It is, for instance, common sense to formulate certain response variables which should be explained by potential influence variables to be identified by an appropriate statistical analysis. Thus, the variables are implicitly split into two groups and possible relationships are investigated among others by contingency table analyses or multivariate regression models. Such techniques, however, ignore indirect influences and the association structure among the explanatories which both may yield important additional insights into the data and its structure. Therefore, other than simple regression models are called for to meet these requirements which usually arise in e.g. biomedical, psychological, or social studies.

For this purpose, the so–called graphical chain models have been introduced mainly by Cox, Lauritzen, and Wermuth (Lauritzen, Wermuth, 1989; Wermuth, Lauritzen, 1990; Cox, Wermuth, 1993) which extend path analytical techniques to other types of multivariate data, which e.g. may contain simultaneously discrete and continuous variables, and to more complex association structures. The basic idea is to represent a multivariate model in a graph where each element is to be interpreted in a purely statistical sense. As additional advantage compared to results of other statistical methods, such a graph is much easier to communicate than endless tables and columns of numbers. The various steps of a statistical analysis to finally reach a graph are, however, much more complicated than a simple regression and no longer easy to handle. Standard software packages do not enable the user to conduct a multivariate analysis based on graphical chain models as a simple routine. Thus, although being a promising statistical tool for reasonably analysing high–dimensional data sets, graphical chain models are up to now not used in daily practice due to the enormous computational effort being still required.

This gap between a theoretically convincing method and its practical inconvenience is intended to be filled by the computer program GraphFitI (Graphical Models Fitting Interactions) which not only offers the possibility for calculating such models but also a help system which assists practitioners or statisticians being not familiar with this technique.

To give more details on the different features of GraphFitI we proceed as follows. The next section summarises the basic concepts of graphical models, before in Section 3 some of the most common computer programs and selection strategies for graphical models are briefly reviewed. Section 4 describes some basic aspects of GraphFitI, its domain and the overall architecture of the program. The design is addressed in somewhat more detail in Section 5. The final section summarises the major advantages of GraphFitI and also addresses a main problem related to model selection in general.

2 Graphical models

Conditional independencies form the key concept of graphical models. The independence structure of a multivariate data set is then displayed in a so–called conditional independence graph. As already mentioned, each element of the graph has a statistical meaning. The vertices represent the variables where each two of them are connected by an edge if they are directly associated. That is, if two vertices are unconnected the corresponding variables are conditionally independent given the remaining variables. Figure 2.1 may serve for illustrative purposes. It portrays the conditional independence structure of a five–dimensional random vector $\boldsymbol{X}= (X_1, \dots, X_5)'$, where the particular independencies depicted there are listed below. Here, for instance $X_1 \perp\!\!\!\perp X_3 \mid \{X_2, X_4, X_5\}$ means that X_1 and X_3, which are unconnected in the graph, are conditionally independent given X_2, X_4, and X_5.

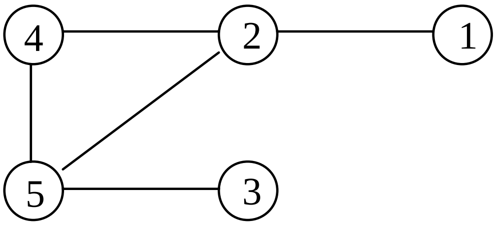


Figure 2.1: *Conditional independencies* $X_1 \perp\!\!\!\perp X_3 \mid \{X_2, X_4, X_5\}$, $X_1 \perp\!\!\!\perp X_4 \mid \{X_2, X_3, X_5\}$, $X_1 \perp\!\!\!\perp X_5 \mid \{X_2, X_3, X_4\}$, $X_2 \perp\!\!\!\perp X_3 \mid \{X_1, X_4, X_5\}$, $X_3 \perp\!\!\!\perp X_4 \mid \{X_1, X_2, X_5\}$ *of a five–dimensional random vector* $\boldsymbol{X}$.

The variables in Figure 2.1 are regarded on equal footing, i.e. each two of them being connected are assumed to influence each other, but none of them is regarded as potentially causal to another variable. In case that such a "causal ordering" seems to be reasonable due to e.g. subject matter knowledge or time it has to be reflected in the graph. This is for instance already called for in simple regression models where response and possible explanatory variables are distinguished. To account for situations where it is sensible to fix a priori a recursive structure among the variables the whole set of variables has to be partitioned accordingly into subsets. The variables of each subset form a block and are depicted within a box in the graph. The blocks are ordered to build up a chain. Per convention, the box to the right contains the pure explanatories and the one to the left the pure responses. In between, we find the so–called intermediates which are responses to variables in previous blocks and simultaneously influential to future blocks. Variables within one block are assumed to be on equal footing, i.e. no explanatory–response association is justified within one subset. Thus, each associated pair of variables is connected by an undirected edge, whereas associations between blocks are assumed as directed and represented as directed edges pointing from the explanatories in previous blocks to responses in the current or future blocks. Due to the inclusion of intermediate variables also indirect influences can be modelled and represented in a graph. Such graphical chain models imply a multivariate distribution which can be factorised as a marginal distribution and a number of conditional distributions where conditioning is always to variables in the current and previous blocks. Let us pick up again Figure 2.1, but now assuming a recursive ordering of the five variables, where X_1 is considered as pure explanatory, $\{X_2, X_3\}$ is the only set of intermediates, and finally $\{X_4, X_5\}$ is the set of pure responses. The resulting structure is displayed in Figure 2.2, where due to this ordering the former undirected edges e.g. from X_2 to X_4, X_5 are replaced with directed edges from the intermediate X_2 to the two responses. The path from X_1 to X_4 via X_2 is an example for an indirect influence here of X_1 on X_4. Finally, the missing edge between X_2 and X_3 means that these variables are independent conditioned only on X_1.

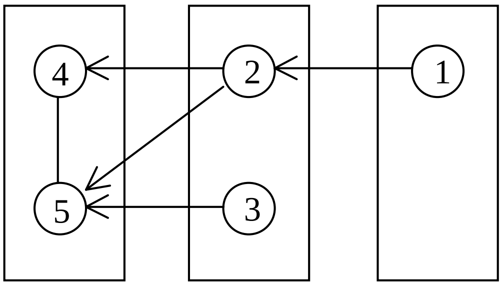


Figure 2.2: *Graphical chain model with following conditional independencies* $X_1 \perp\!\!\!\perp X_3 \,|\, \{X_2\}$, $X_1 \perp\!\!\!\perp X_4 \,|\, \{X_2, X_3, X_5\}$, $X_1 \perp\!\!\!\perp X_5 \,|\, \{X_2, X_3, X_4\}$, $X_2 \perp\!\!\!\perp X_3 \,|\, \{X_1\}$, $X_3 \perp\!\!\!\perp X_4 \,|\, \{X_1, X_2, X_5\}$ *of a five–dimensional random vector* $\boldsymbol{X}$.

The main task is now to fit a graphical chain model to a given data set. Various approaches can be thought of. In the next section, we will sketch different computer programs and selection strategies without claiming completeness.

3 State of the art

Let us first review the main software packages for calculating graphical models. Note that the various programs come up with certain limitations due to the type of graphical model or the scaling of the variables.

If focus is on undirected graphs regardless whether the variables are discrete, continuous or mixed, MIM (cf. Edwards, 1987, 2000) can be recommended. MIM can also cope with missing data and offers different selection strategies addressed later in this section. Its latest version additionally allows for the calculation of graphical chain models but it is still restricted to simple models involving few variables. If larger models are to be considered, all calculations have to be carried out using standard software in statistics.

In contrast, DIGRAM (Kreiner, 1987, 1992) fits chain graphs but it is restricted to discrete data. In this respect, however, it is rather convincing since various exact tests for checking the hypothesis of conditional independence are incorporated.

Contingency tables may be analysed using CoCo (Badsberg, 1992), where also missing values can be handled and a large number of exact conditional tests for decomposable models is included.

The interactive modelling of discrete data via loglinear models is enabled by TURNER (Lauer, 1998), which are displayed as hierarchical loglinear models.

TETRAD (Spirtes, Glymour, Scheines, 1993) basically calculates directed acyclic graphs.

In the field of Bayesian belief networks the program HUGIN (Hugin Expert A/S) has to be mentioned. It gives the possibility to construct model decision support systems in domains characterised by inherent uncertainty. The model is organised as a direct acyclic graph which can handle discrete and to some extent also mixed data.

A program which is not directly connected to graphical models but which also uses concepts of these modelling technique is BUGS (Spiegelhalter, Thomas, Best, 1999). Complex statistical problems for which no exact analytic solution is at hand are treated within a Bayesian framework, where inference is then carried out via MCMC methods. Graphical models come here into play for representing the conditional assumptions regarding the independence structure of the underlying statistical problem.

All these programs assist the user in conducting a model search, where the space is restricted due to the given limitations of the software. For doing so, different types of selection strategies or algorithms are embedded. TETRAD, for instance, uses the PC–algorithm which has been developed for directed acyclic graphs. Most of the other softwares base their model search upon forward and backward selections using p–values or the AIC or BIC as criteria as for instance CoCo. MIM additionally searches the model space using the Edwards–Havránek procedure (Edwards, Havránek, 1985, 1987). This procedure fits a series of models where the accepted ones are stored in a certain subset and the rejected ones in another. The third subset contains those models which are undetermined. It continues until all models in the considered family are either accepted or rejected. For graphical chain models with mixed variables, Cox and Wermuth (1996) have proposed to use a more heuristic strategy which combines screenings for interactions and nonlinear relations with forward and backward selections in a system of univariate regressions. Alternatively, it is also possible to perform backward and forward selections in course of a model fit based on the ME algorithm (Edwards and Lauritzen, 1999), which takes the original chain graph structure of the model into account. This algorithm is incorporated in MIM. Besides these more classical approaches, there are also some Bayesian strategies as for instance those implemented in BUGS and HUGIN where e.g. BUGS uses MCMC–algorithms based on the Gibbs sampler. Recent approaches exploit the reversible jump MCMC–algorithm originally introduced by Green (1995). For instance, Giudici and Green (1999) have proposed an MCMC–algorithm which allows a model selection in undirected decomposable Gaussian models only. This algorithm has then been extended to discrete data by Giudici, Green, and Tarantola (1999).

Comparing now the above computer programs, a gap is obviously left to be filled by a program which enables the user to fit graphical chain models to mixed multivariate data in a convenient and user–friendly way.

4 Basic features of GraphFitI

To give an idea of the domain of GraphFitI, one can roughly say that the program provides the user with a statistical analysis based on graphical chain models and with an integrated help tool for assisting him/her at the different steps of the analysis.

Fitting a graphical chain model to a high–dimensional data set is currently still rather cumbersome since mixed data with a large number of variables cannot be handled with the existing computer programs especially designed for graphical models. Even the algorithm recently integrated in MIM is in practice restricted to only a few variables since it is very time consuming. GraphFitI offers the fit of graphical chain models using a more heuristic strategy introduced by Cox and Wermuth (1996). This strategy is mainly based on the calculation of univariate regression models where it is roughly divided in two steps. First, a screening is performed regarding possible second–order interactions and nonlinear relations (Cox, Wermuth, 1994). The search is carried out visually inspecting graphical representations of the corresponding estimated coefficients calculated for the particular regression model. These figures remind of normal–probability–plots. GraphFitI offers different possibilities for conducting the screening, which will be explained in some more detail in the next section. The second step consists of forward and backward regressions depending on the scale of the response variable. For a more detailed discussion see for instance Caputo, Heinicke, and Pigeot (1999). This strategy is implemented in GraphFitI and can be run automatically, but also interactively interrupted if the obtained numerical results are in no way consistent with common substantive knowledge.

As already mentioned, besides the pure calculation of a graphical chain model, one of its challenging features is the link to the presentation of the model in a graph. But drawing the graph by hand is again inconvenient. Thus, GraphFitI simultaneously depicts the fitted model in a graph where this graphical representation can be modified interactively which possibly implies that parts of the model have to be recomputed. This is of course indicated by the program as a warning. Edges added or deleted by the user are marked in different colours to make them easily identified. To improve the visualisation different commands are offered as for instance zooming, highlighting or hiding parts of the graph, or additional information on the strength or direction of the detected associations.

The third component of GraphFitI includes a help module which does not only explain how to use the program itself but also assist users with limited experience in the field of graphical models.

A user–friendly graphical interface provides the user with a convenient handling of the different steps of the statistical analysis from the data input, the construction of the rough dependence chain up to the visualisation of the final chain graph. It should, however, be noticed that GraphFitI mainly aims at analysing graphical chain models, i.e. although it also allows for fitting undirected graphical models there are faster alternatives in this case. Since the statistical procedures are also part of the program, GraphFitI is a stand–alone system and not an interface for an existing statistical software package. Figure 4.1 illustrates the overall system architecture of GraphFitI.

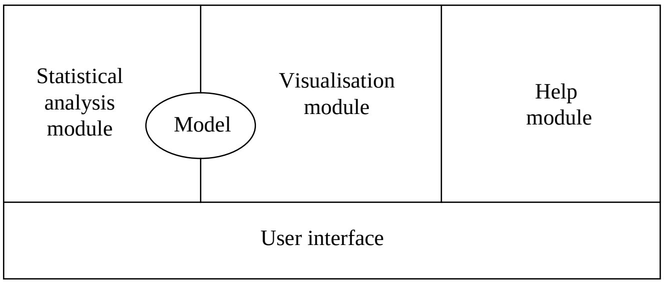


Figure 4.1: *Overall system architecture of GraphFitI.*

5 Design

GraphFitI is written in Java to allow it to run under several platforms. As already mentioned the program consists of three main modules which are all connected by the user interface. In the following we give a short description of the several modules and of the most important data structure, the statistical model.

5.1 The model

The model is the central data structure in the system. Hidden from the user, it holds information which is needed both by the analysis module and by the graphical representation module. There are two ways of information flow. First, the model is built by the selection strategy of the analysis tool. From this model the graph is drawn by extracting information like if and how strong two variables are dependent, in which chain components the variables are placed, and so on. Since it is possible to interfere with the selection strategy through the graph a message flow from the graph to the analysis module is implemented as well. There is a well–defined interface to the model so that is possible to connect several selection strategies with it.

5.2 The statistical analysis module

The statistical analysis module mainly consists of classes which are needed during the selection strategy. There exist classes which perform normal regression, logistic regression, and multivariate logistic regression. These classes are glued together by the overall selection strategy performing the model selection in the framework of Cox and Wermuth (1996). In this class mainly the forward and backward selection is implemented. Besides that it has the task to choose the correct variables of the regression as well as the adequate type of regression depending on in which step the algorithm is at the moment. Last, also the non–visual part of the screening belongs to this module.

In detail, the strategy works as follows. First, a screening of the data for interactions and nonlinear relations is performed to get a hint which of them may need consideration (see also Section 5.4). Then, the current target is chosen from the model and the correct type of regression needed for this target. This information is used to initialise the regression, which estimates the models during the backward and forward selection steps. In the next step, a forward selection decides which of the interactions and nonlinear relations found in the screening are of such importance that they have to be included in the model. For the resulting model a backward selection is performed to come to a first reduction of the model. Next, all possible qualitative and mixed interaction terms in the current model are calculated and again a backward selection based on the extended model is carried out. In the last step, all quantitative nonlinear effects and interactions of the actual model are added, and after a final backward selection we get the final model for the current target. This information is used to update the model, and we go on with applying this strategy to the next target until no variable is left. Figure 5.1 serves for illustrating these steps.

5.3 The visualisation module

The graphical representation is designed such that it is independent of the underlying selection strategy to allow an easy extension of both modules without changing the other one. All information needed to draw a corresponding graph is hold in the model data structure. Besides giving a visualisation of the model the user can interact with the graph (see also Section 4). The possibility of changing the assumed or calculated statistical model by a visual modification of the graph offers the user a very comfortable way for realising such changes without the necessity of written commands.

Note, that until now, only the design of that module is finished and has not been completely implemented in the program yet. Currently, a very detailed protocol gives information about the results of the strategy and the resulting graph is summarised in an adjacency matrix.

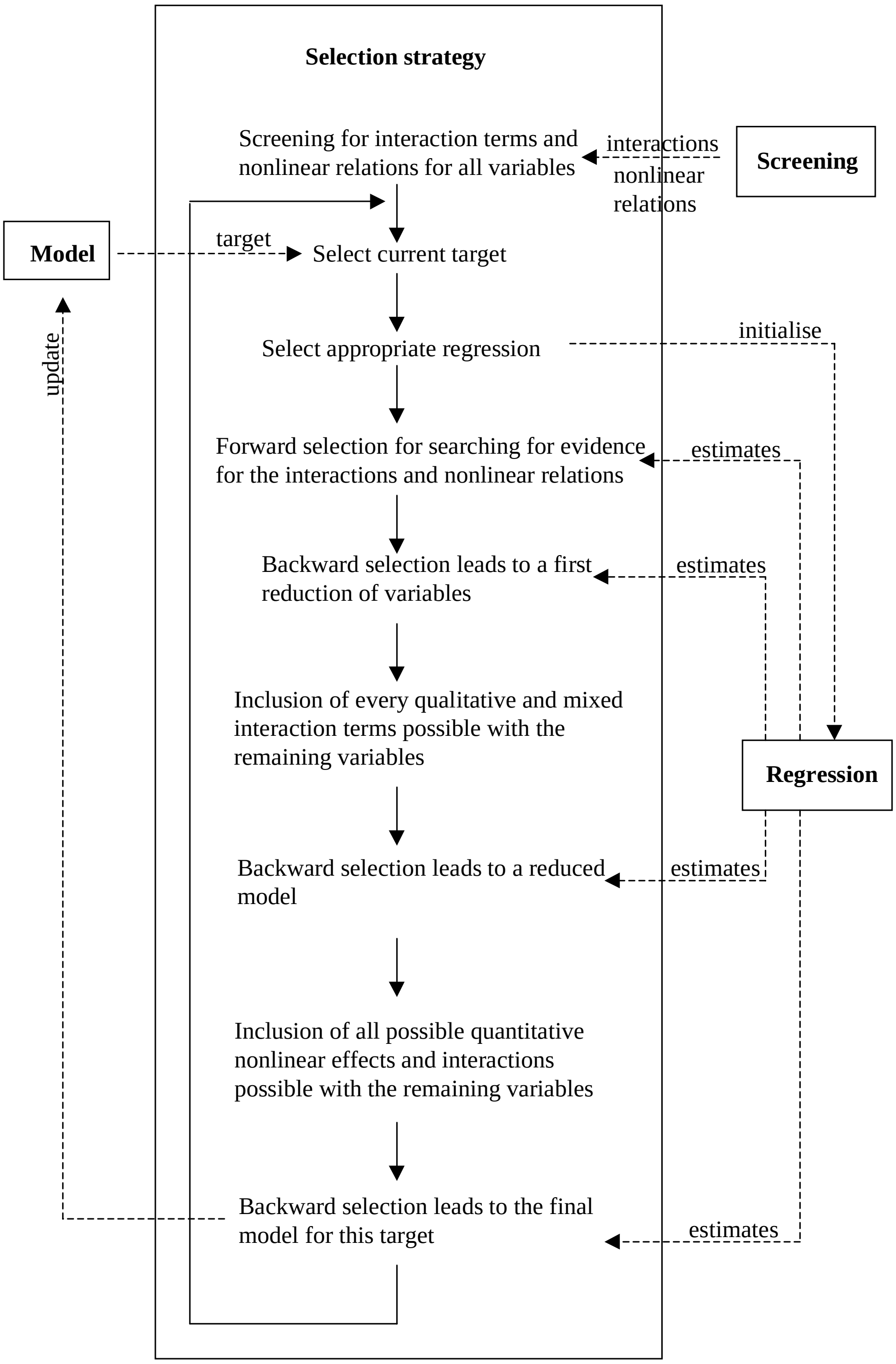
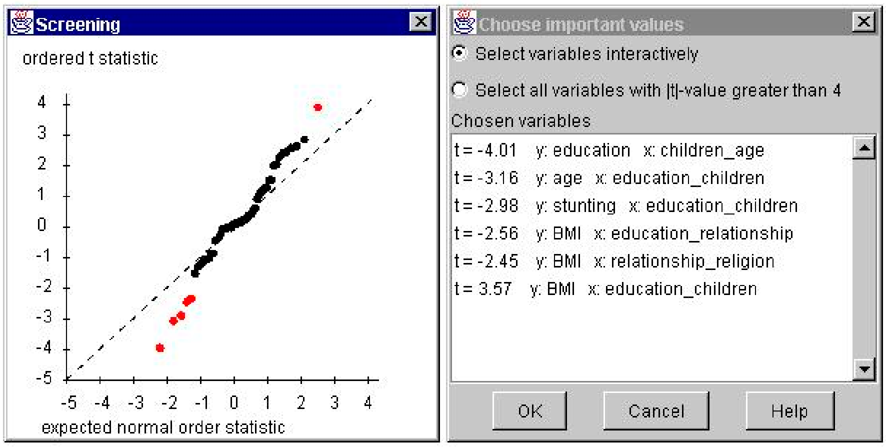

Selection strategy
Screening for interaction terms and nonlinear relations for all variables
interactions
nonlinear relations
Screening
Model
target
Select current target
Select appropriate regression
initialise
update
Forward selection for searching for evidence for the interactions and nonlinear relations
estimates
Backward selection leads to a first reduction of variables
estimates
Inclusion of every qualitative and mixed interaction terms possible with the remaining variables
Regression
Backward selection leads to a reduced model
estimates
Inclusion of all possible quantitative nonlinear effects and interactions possible with the remaining variables
Backward selection leads to the final model for this target
estimates

Figure 5.1: *The selection strategy.*

5.4 The user interface

Much attention has been paid to the design of a user–friendly interface. Each action is performed via a dialog to avoid a complicated command line system. Besides that, each dialog uses as many graphical representations as possible to allow an intuitive use of GraphFitI. The screening for interactions and nonlinear relations may serve as an example. If e.g. the entry screening for interactions is chosen from the menu two windows appear which are shown for a given data set in Figure 5.2. In the left window, the expected normal statistics are plotted against the t–values resulting from a trivariate regression for each possible interaction (cf. Section 4). The size of this window can be adjusted to the number of points appearing in the picture by dragging the bottom right corner. In the right window, it can be selected whether the program should choose the striking points automatically or whether the user wants to do it interactively. In the first case, each point belonging to a $\|t\|$–value greater than 4 is selected. In the second case, a rectangle can be dragged around the striking points in the left window. In both cases, the selected points are marked and listed in a box, which means that the corresponding t–values and the regression models for which the test statistics have been calculated are given. If the points are selected interactively it is possible to remove an entry from the list. The screening for nonlinear relationships is based on the same principle.



● Select variables interactively

○ Select all variables with |t|-value greater than 4

Chosen variables

t = -4.01 y: education x: children_age
t = -3.16 y: age x: education_children
t = -2.98 y: stunting x: education_children
t = -2.56 y: BMI x: education_relationship
t = -2.45 y: BMI x: relationship_religion
t = 3.57 y: BMI x: education_children

Figure 5.2: *Screening for interactions.*

The program also follows current standards in the error handling. The user is prevented as much as possible from wrong inputs by the program. Thus, dialogs are typically conducted by using list boxes or buttons to avoid such wrong inputs. This implies that the user is informed by the program if his/her action leads to essential interventions in the current process. In addition, the dialogs contain an early check of the input with respect to possible errors. In case of an error, a routine is called describing the error as precisely as possible.

5.5 The help module

The help module, which is written in the Java-help system, gives information both for using the program and on the statistical background. The idea is that even users who are not familiar with the background of graphical models should be able to use the program. From each dialog of the program it is possible to jump via a help button to the help entry describing this dialog. From that, there are references which lead to the statistical background information being relevant for the actions in this dialog. There also exist a table of contents, an index, and a search function to directly jump to a topic of interest.

6 Discussion

The task of fitting a graphical chain model to a multivariate data set implies different steps and different sources of knowledge. Subject–matter knowledge is needed to formulate the rough dependence chain on which all future steps of the statistical analysis have to be based. The final structure is then derived from the data using an appropriate selection strategy. GraphFitI now supports the user in selecting the variables, fixing their measurement scales, and building the dependence chain by providing a user–friendly interface (cf. Blauth, Pigeot, Bry, 2000). But the major feature of GraphFitI is the visualisation of the fitted chain graph at each step of the implemented selection strategy and the possibility of interactively modifying the graph or the model, respectively. Such modifications are again based on subject–matter knowledge where the remaining parts require statistical knowledge on the theory and appropriate use of graphical models. With this respect, a user being non–familiar with this statistical analysis tool is supported by an integrated help system.

It should, however, be emphasised that even the results of such an analysis may be convincing from a substantive point of view, there are typically other statistical models which are also consistent with the given data set. These models may result from other selection strategies or just from simply adjusting the implemented criteria of the applied one.

Thus, our current research interest is not only on extending the theory of graphical models to other types of data, but also on developing alternative selection strategies e.g. based on reversible jump MCMC or genetic algorithms. The idea is to also implement other strategies in GraphFitI to provide the user with possibly different competing models which may lead to a fruitful platform for detailed discussions on the research problem.

References

Badsberg, J.H. (1992). Model search in contingency tables by CoCo. In: Dodge, Y., Whittaker, J. (eds) *Computational Statistics, CompStat 1992 Neuchâtel.* Physica–Verlag, Heidelberg, 251–256.

Blauth, A., Pigeot, I. & Bry, F. (2000). Interactive analysis of high–dimensional association structures with graphical models. *Metrika* **51**, 53–65.

BUGS. `http://www.mrc-bsu.cam.ac.uk/bugs/`

Caputo, A., Heinicke, A. & Pigeot, I. (1999). A graphical chain model derived from a selection strategy for the sociologists graduates study. *Biometrical Journal* **41**, 217–234.

Cox, D.R. & Wermuth, N. (1993). Linear dependencies represented by chain graphs (with discussion). *Statistical Science* **8**, 204–283.

Cox, D.R. & Wermuth, N. (1994). Tests of linearity, multivariate normality and the adequacy of linear scores. *Applied Statistics* **43**, 347–355.

Cox, D.R. & Wermuth, N. (1996). *Multivariate Dependencies – Models, Analysis and Interpretation.* Chapman & Hall, London.

Edwards, D. (1987). A guide to MIM. *Research Report* **87/1**. Statistical Research Unit, University of Copenhagen, Denmark.

Edwards, D. (2000). *Introduction to graphical modelling.* 2nd ed., Springer–Verlag, New York.

Edwards, D. & Havránek, T. (1985). A fast procedure for model search in multidimensional contingency tables. *Biometrika* **72**, 339–351.

Edwards, D. & Havránek, T. (1987). A fast model selection procedure for large families of models. *Journal of the American Statistical Association* **82**, 205–213.

Edwards, D. & Lauritzen, S.L. (1999). The ME algorithm for maximizing a conditional likelihood function. *Research Report* **R–99–2015**. Department of Mathematical Sciences, Aalborg University, Denmark.

Giudici, P. & Green, P.J. (1999). Decomposable graphical Gaussian model determination. *Biometrika* **86**, 785–801.

Giudici, P., Green, P.J. & Tarantola, C. (1999). Efficient model determination for discrete graphical models. *submitted.*

Green, P.J. (1995). Reversible jump markov chain Monte Carlo computation and bayesian model determination. *Biometrika* **82**, 711–732.

Hugin Expert A/S. `http://www.hugin.dk`

Kreiner, S. (1987). Analysis of multidimensional contingency tables by exact conditional tests: techniques and strategies. *Scandinavian Journal of Statistics* **14**, 97–112.

Kreiner, S. (1992). *Notes on DIGRAM, Version 2.10.* The Danish Institute of Educational Research, Copenhagen, Denmark.

Lauer, S. (1998). Interactive modelling of categorical data. In: B. Marx B, H. Friedl (eds). *Proceedings of the 13th International Workshop on Statistical Modeling,* New Orleans, July 27-31, 1998, 443–446.

Lauritzen, S.L. & Wermuth, N. (1989). Graphical models for associations between variables, some of which are qualitative and some quantitative. *Annals of Statistics* **17**, 31–57.

Spiegelhalter, D.J., Thomas, A. & Best, N.G. (1999). *WinBUGS Version 1.2 User Manual.* MRC Biostatistics Unit.

Spirtes, P., Glymour, C. & Scheines, R. (1993). *Causation, Prediction, and Search.* Springer–Verlag, New York.

Wermuth, N. & Lauritzen, S.L. (1990). On substantive research hypotheses, conditional independence graphs and graphical chain models (with discussion). *Journal of the Royal Statistical Society, Series B* **52**, 21–72.