A Review of Three Directed Acyclic Graphs Software Packages: MIM, Tetrad, and WinMine

Dominique HAUGHTON, Arnold KAMIS, and Patrick SCHOLTEN

This article offers a review of three software packages that estimate directed acyclic graphs (DAGs) from data. The three packages, MIM, Tetrad and WinMine, can help researchers discover underlying causal structure. Although each package uses a different algorithm, the results are to some extent similar. All three packages are free and easy to use. They are likely to be of interest to researchers who do not have strong theory regarding the causal structure in their data. DAG modeling is a powerful analytic tool to consider in conjunction with, or in place of, path analysis, structural equation modeling, and other statistical techniques.

KEY WORDS: Bayesian networks; Causality; Data mining; Indirect effects.

1. INTRODUCTION

This article reviews three packages for estimating directed acyclic graphs (DAGs) from data, namely, MIM (version 3.2.0.3 communicated to us directly from the author; MIM 2005), Tetrad III and IV (Tetrad III version 3.1, Tetrad IV version 4.3.5-0; Tetrad 2006) and WinMine (WinMine version 2.0; WinMine 2005). DAGs have been used to model a wide variety of phenomena, such as:

- U.S. unemployment (Awokuse and Bessler 2003)
- Web-based personalized marketing (Mussi 2003)
- Student behavior in distance learning (Xenos 2004)
- Forensic analysis of fires (Biedermann et al. 2005)
- Wealth of countries (Bessler and Loper, 2001)

In general, the use of DAG modeling packages, and this article, may interest any researcher who:

- would like to learn about different software packages that are designed to estimate DAGs from data;
- has some background knowledge or constraints, but no strong theory per se;
- is attempting to discover a causal relationship in a dataset but has no strong theoretical foundation for guidance;
- senses that structural equation modeling (LISREL, EQS, Amos, PLS, etc.) is appropriate, but that some preprocessing of causal structure is necessary.

The primary consideration in selecting the three software packages under review was cost, usability, and accessibility: all three packages are available at no cost and are easily downloadable at the Web sites listed in the references. Our overriding goal is to discuss the similarities, differences, and usability of software tools designed to discover DAGs. This review is intended to help guide researchers in choosing software packages that would be appropriate for their particular needs.

In general we find that all three packages are relatively easy to use. Tetrad III is the only package that has a command-line interface exclusively, while Tetrad IV is menu-driven. WinMine and MIM have a combination of command-line and graphical interfaces. All three packages offer a variety of tools with sensible default values. Although the graphical tools in MIM, Tetrad IV, and WinMine allow researchers to obtain visual insights from the data, the command line tools permit researchers to batch scripts and increase modeling efficiency with complex models.

It is important to stress that the packages discussed in this review differ from other packages—like LISREL, EQS, and AMOS—which provide coefficient estimates once the causal structure of the variables is specified. The primary objective of the three software packages reviewed is to discover the DAG in the data. Once the causal structure is identified with one of the software packages—Tetrad, WinMine, or MIM—the structure can be used as input to develop structural equation models (SEMs) and obtain coefficient estimates in software packages such as LISREL. Thus, the three software packages under review are ideal in environments where the causal structure of the variables is unknown to the researcher because of a lack of sufficient theoretical knowledge. Our review of the DAG software packages will be based on data from the Vietnam Living Standards Surveys of 1992 and 1998, an overview of which is provided below in Section 1.2. Our results indicate that each of the software packages produces similar but not identical results. The primary challenge in using any of the three software packages is in preparing the data, constraining the models with background knowledge and interpreting the results.

1.1 Brief Overview of the DAG Methodology

As mentioned earlier, the packages reviewed in this article rely on the concept of DAGs, which are directed graphs that contain no directed cycles. We give a brief introduction to this concept and refer the reader to, for example, Edwards (2000, chap. 7) for further information. Articles in Wikipedia mentioned in the references might also prove helpful.

In our context a directed graph is an ordered pair $G = (V, A)$ where V is a set of nodes or vertices (variables) and A is a set of ordered pairs of vertices (or a set of directed edges). For example, let $V = \{x, y, z\}$ and $A = \{(y, x), (z, x)\}$. Then, $G = (V, A)$ is a directed graph, represented in Figure 1(a). The directed graph G contains no directed cycles since for any vertex

Dominique Haughton is Professor of Mathematical Sciences, Arnold Kamis is Assistant Professor of Computing Information Systems, and Patrick Scholten is Assistant Professor of Economics, Bentley College, Waltham, MA 02452 (E-mail: *dhaughton@bentley.edu*).

DOI: 10.1198/000313006X117972

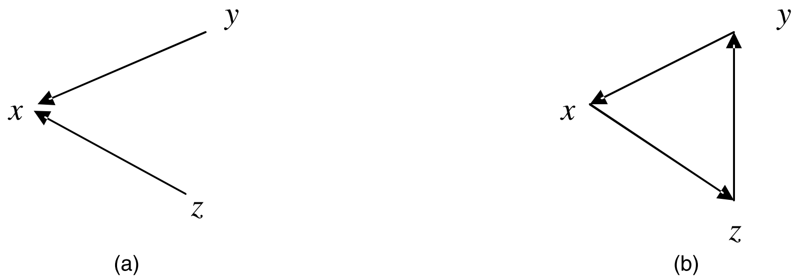


Figure 1. Examples of directed graphs.

x there does not exist a nonempty directed path that starts and ends with x. Therefore, G is a directed acyclic graph (DAG), as seen in Figure 1(a).

In contrast, the directed graph, $G' = (V, A')$, where $V = \{x, y, z\}$ and $A' = \{(y, x), (x, z), (z, y)\}$ is not an acyclic directed graph because there is a nonempty directed path that starts and ends with x, as can be seen clearly in Figure 1(b).

A Bayesian network is a particular type of DAG, where the variables in the network are represented by the nodes of the DAG and the set of directed edges represents conditional dependence relationships among the variables. When there is a directed edge from node x to y, then x is said to be a parent of y. Given this structure a Bayesian network is a representation of the joint density of all variables represented by the nodes of the graph. Let $V_1, \ldots, V_n$ denote the variables and *parent*(V_i) the set of parents of the variable V_i. We say (Pearl 2000) that a DAG represents the joint distribution f if the following decomposition holds:

$$f\left(V_1, \ldots, V_n\right) = \prod_{i=1}^{n} f\left(V_i \mid \text{parent}(V_i)\right),$$

where the product is calculated over all variables V_i in the Bayesian network (nodes in the DAG), and each term in the product refers to the conditional density of V_i given its parents. Because of a well-known theorem (see, e.g., Pearl 2000, theorem 1.2.7, p. 19), the fact that a DAG represents a joint distribution is equivalent to each variable being independent of its nondescendants given its parents.

Applied to the DAG in Figure 1(a), $y \rightarrow x \leftarrow z$, the equation above implies that y and z are unconditionally independent. When such a DAG occurs, by itself or as part of a larger DAG, x is referred to as a *collider*. Colliders block dependency between variables. In contrast, the DAG $y \leftarrow x \rightarrow z$ has no collider; hence dependency propagates through x, and y and z are independent, given x.

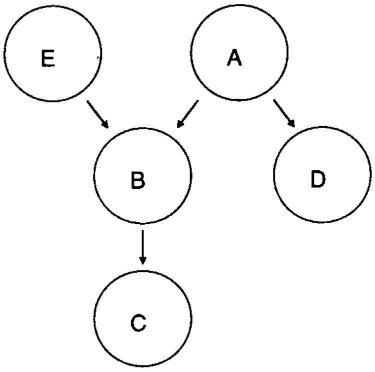


Figure 2. Example of a directed acyclic graph.

Consider a slightly more complex example as displayed in Figure 2.

In this example, B is a collider. Interestingly, by the theorem just mentioned, it follows that D is independent of C, given B. That is, once B is known, D and C are independent even though they have a common parent A.

1.2 Data Overview

In this article, we use a subset of data from the Vietnam Living Standards Surveys (VLSS) of 1992 and 1998 to provide a review of the three DAG packages. These surveys essentially follow the World Bank LSMS (Living Standard Measurement Study) framework, with questionnaires that have been adapted to the Vietnamese context and tested in the field. The survey sampling (stratified with two-level clustering) was performed carefully and competent data cleaning was implemented. Data from the VLSS are widely considered to be of high quality, and many publications have arisen which rely on them.

Table 1 gives a description and summary statistics of the variables that will be used in the article. Our analyses are based on a panel of 4,272 households interviewed both in 1992 and 1998. These data contain both continuous and discrete variables. For more details on the surveys, we refer the reader, for example, to Haughton, Haughton, and Phong (2001, chap. 1).

The remainder of the article is structured as follows. Sections 2, 3, and 4 are devoted to reviewing the MIM, Tetrad, and WinMine software packages, respectively. In each of these three sections, we introduce the package and describe how the data must be prepared for use, describe briefly the main idea behind its algorithm, present output with interpretation, and finally discuss the pros and cons of the package. Section 5 concludes with a summary of our findings.

2. MIM

2.1 Introduction and Data Preparation

MIM (Mixed Interaction Modeling) was created by David Edwards, and it can be obtained at no cost at *http://www.hypergraph.dk/*. It works on any reasonably current PC. The MIM package fits a variety of models to data, including DAGs. According to the documentation, MIM implements "a full range of statistical techniques based on the models, including maximum likelihood estimation, hypothesis testing, model selection, and much more."

Table 1. Variables Used and Summary Statistics

Variable	Description	Mean	Std. dev.
female92	Female? (yes = 1, no = 0)	0.26	0.44
Hgovt92	Head in government? (92) (yes = 1, no = 0)	0.02	0.14
HWcoll92	Head white collar? (92) (yes = 1, no = 0)	0.11	0.32
hhsize92	Household size (92)	5.03	2.17
kid92	Number of children (92)	2.02	1.56
eld92	Number of elderly (92)	0.40	0.66
nosick92	Number of sick (92)	1.39	1.56
domrem	Domestic remittances amount in ’000 VND (92)	145.10	821.36
oversrem	Overseas remittances amount in ’000 VND (92)	372.86	5558.70
domrem92	Received domestic remittances? (92) (yes = 1, no = 0)	0.18	0.39
ovsrem92	Received overseas remittances? (92) (yes = 1, no = 0)	0.05	0.22
domgive	Gave domestic remittance (92) (yes = 1, no = 0)	0.16	0.37
overgive	Gave overseas remittances (92) (yes = 1, no = 0)	0.00	0.03
savings	Amount of savings in ’000 VND (92)	1545.35	13446.61
sborr92	Amount of borrowing in ’000 VND (92)	1033.74	17773.07
lend92	Lent money? (92) (yes = 1, no = 0)	0.16	0.37
age	Age of head in years (in 1998)	48.33	13.81
age2	Age squared	2526.09	1437.98
educyr98	Number of years of education of head	6.90	4.31
urban98	Urban (98) (yes = 1, no = 0)	0.21	0.41
medy98I	Number of years of education of mother	2.61	1.50
medyr98M	Missing mother’s education? (yes = 1, no = 0)	0.66	0.47
fedy98I	Number of years of education of father	3.54	2.69
fedyr98M	Missing father’s education? (yes= 1, no = 0)	0.45	0.50
fatherlive	Father is alive? (yes = 1, no = 0)	0.28	0.45
motherlive	Mother is alive? (yes = 1, no = 0)	0.46	0.50
faWcoll92	Father white collar? (92) (yes = 1, no = 0)	0.11	0.31
moWcoll92	Mother white collar? (92) (yes = 1, no = 0)	0.10	0.30
rpcexp92adj	Real per cap. exp. in ’000 98 VND (92)	1290.83	1002.39
rpcexp98adj	Real per cap. exp. in ’000 98 VND (98)	2676.88	2190.78

NOTE: negative values for real per capita expenditures occur for about 20 cases, and are due to the fact that government transfers were subtracted from household expenditures. VND refers to the Vietnamese Dong, the Vietnamese currency.

Although the MIM help files are generally adequate, there is a companion book to the MIM software (Edwards 2000) that is extremely useful and includes sample code for various models. A discussion of causal inference (following Pearl’s model and Rubin’s model) is included in the book (chapter 8), but there is no claim that models estimated by MIM are in any way “causal.” Indeed, Edwards sets the tone on his view of causality in the opening quote of the book, which is attributed to Democritus: “I would rather discover a single causal relationship than be king of Persia.” Hence one can surmise that although the package enables the modeling of large, complex graphs, causality cannot in general be claimed.

There are three steps that need to be completed to read data into MIM. First, the data must be saved as a flat file in standard delimited format. Second, continuous variables are declared using the “Cont” command followed by single-letter, case-sensitive variable names; that is, {abc ... ABC}. Categorical variables are declared using the “Factor” command followed by single-letter, case-sensitive names of the variables that include the number of categories for each variable; that is, $\{a2\,b3\,C4\}$ indicates that variable “a” has two categories, “b” has three categories, and “C” has four categories. Finally, the “read” command followed by the path to the flat file and list of variables will read the data into MIM. The variable names are not user-friendly or conducive to easy recall; hence, variable labels can be added using the “label” command followed by the single-lettered variable name and the variable label in quotes. Importantly, the code used to read and label the data file into MIM must end with an exclamation point “!”. The MIM code to bring in our VLSS dataset is given in Figure 3.

Once the dataset is read into MIM, it is quite easy to view the variables with their labels and to view the data. It is also straightforward to save the workspace as a file which can be retrieved at the next session.

2.2 Main Ideas of the Algorithm

Within MIM, DAGs are built by defining blocks of variables and declaring the blocks in the order in which the variables are expected to influence one another. MIM will create directed edges according to this user-defined block structure. Within a block, the edges between variables—if any—will be undirected. An undirected graph, G, is an ordered pair $G = (V, E)$, where V is a set of nodes and E is a set of unordered pairs called *edges*. The defining characteristic of an undirected graph is that directional inferences cannot be made between variables.

MIM uses a variety of standard models (typically estimated by maximum likelihood) to determine variable edges among pairs of variables in different blocks or within a block. Other models are also possible. The details of the estimation procedures used in MIM are given in Appendix D of Edwards (2000).

2.3 Graphical and Numerical Output

We initially use a backward stepwise procedure to let MIM identify edges between pairs of variables on the full dataset.

```
Cont  abcdefghijklmnopqrstuvwxyzABCD
read 'c:\causality\vulnsmallMIM.raw'  abcdefghijklmnopqrstuvwxyzABCD
label a "age"; label b "urban98" label c "hhsize92";
label d "kid92"; label e "eld92"; label f "lend92";
label g "sborr92"; label h "borr92"; label i "educyr98";
label j "fatherlive"; label k "motherlive"; label l "fedyr98M";
label m "medyr98M"; label n "age2"; label o "fedy98I";
label p "medy98I"; label q "female92"; label r "nosick92";
label s "oversrem"; label t "domrem"; label u "overgive";
label v "domgive"; label w "savings"; label x "faWcoll92";
label y "moWcoll92"; label z "Hgovt92"; label A "HWcoll92";
label B "ovsrem92"; label C "domrem92"; label D "DLogExpadj";
!
```

Figure 3. Sample code to read and label a data file into MIM.

Therefore, in the undirected model, we define no blocks. The command "SatModel" runs a saturated full model on a set of variables declared by the user. In this case, we perform a single stepwise selection procedure as indicated by the command "stepwise o." If the variable set is omitted, all variables are used in the model. The code and partial output from this operation are given in Figure 4.

The (partial) output in Figure 4 contains a χ^2 test to determine whether an edge between two variables is statistically significant at the default 5% level. This level of significance can be reset to other levels by the user. Statistically significant edges are marked with a "+" sign. In the context of the output reported in Figure 4, the edges between variables $[AC]$ and $[Aq]$ are statistically significant at the 5% level. That is, there exists a statistically significant relationship between the variables "HWcoll92" and "domrem92" and "HWcoll92" and "female92". This means that there is a statistical association between the fact that the head of household is a white collar worker and the fact that the household received domestic remittances in 1992, and between gender and the fact that the head of household is a white collar worker.

Figure 4 provides a statistical test to determine edges between variables. MIM is then able to generate an independence graph depicting these edges between variables ("Graph" command). By default, the variables of an independence graph are arranged in a circle by alphabetical order. The variables (nodes) of the independence graph are drawn as circles with the corresponding variable name (letter) inside; a hollow circle indicates a continuous variable and filled-in circle depicts a categorical variable. Variable labels appear next to the corresponding hollow or filled-in circle. Undirected models return solid lines that represent statistically significant edges while nonstatistically significant edges are shown as dashed lines. The graph from the undirected model is given in Figure 5. Note that the information contained in Figure 5—albeit somewhat cumbersome for this particular model—contains the same information as Figure 4. The graph feature is substantially more beneficial when the number of variables is more tractable. The initial positioning of the nodes on the graph is MIM's default; it is possible to drag and move nodes

```
MIM->Retrieve C:\Causality\VULN32.MIM
MIM->satmod;  stepwise o
Coherent Backward Single-step Selection.
Fixed edges: none.
Critical value:   0.0500
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Model: //ABCDabcdefghijklmnopqrstuvwxyz
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge         Test
Excluded   Statistic DF          P
    [AB]      0.0404  1      0.8407
    [AC]      7.2909  1      0.0069 +
    [AD]      0.2320  1      0.6300
. . .
. . .
. . .
    [Ap]      1.2553  1      0.2625
    [Aq]     75.7206  1      0.0000 +
    [Ar]      0.0688  1      0.7931
```

Figure 4. MIM undirected model (partial output).

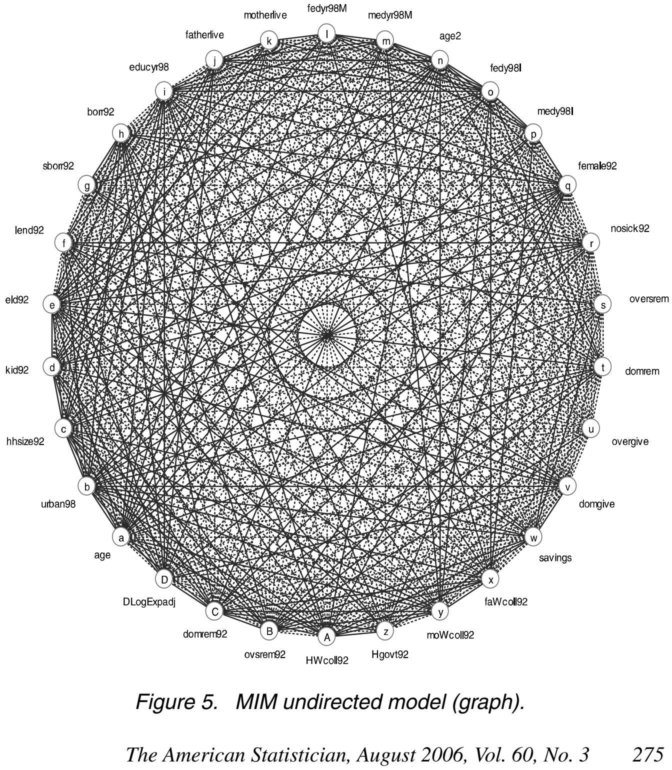


Figure 5. MIM undirected model (graph).

```
MIM->satmod;  stepwise o
Coherent Backward Single-step Selection.
Fixed edges: none.
Critical value:   0.0500
Block no.  1
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Block no.  2
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Model: //Cbcdfhijmrv
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge         Test
Excluded   Statistic DF           P
    [bc]      73.5375  1      0.0000 +
    [bd]     132.9679  1      0.0000 +
. . .
. . .
. . .
    [rC]       7.7566  1      0.0054 +
    [vC]       0.0907  1      0.7632
Block no.  3
Decomposable mode, Chi-squared tests.
DFs adjusted for sparsity.
Model: //CDbcdfhijmrv
Deviance:   0.0000 DF:   0 P:  1.0000
    Edge         Test
Excluded   Statistic DF           P
    [bD]      25.9137  1      0.0000 +
    [cD]      18.0657  1      0.0000 +
. . .
. . .
. . .
    [vD]       7.3013  1      0.0069 +
    [CD]       8.4991  1      0.0036 +
MIM->SaveOutput
C:\Causality\MIMOutputUndirectedandDirected
```

Figure 6. MIM directed model (code and output).

and labels. The primary benefit of running an undirected model is that it identifies statistically significant edges that can aid the user in defining manageable variable blocks.

Once the block structure has been identified using the edges from the undirected model, this structure is set using the "SetBlocks" command. Once set, the "SatModel" command initiates a block-recursive model that can be defined to run a DAG. Figure 6 contains the code and (partial) output for the directed model. In this case, the graphical analysis contained in Figure 7 provides a more readable format in which to interpret the directed model. For instance, the first block, variable "b" labeled "Urban98" in Figure 7, has no edge that feeds into it, as set by the user. Instead, Urban98 has directed edges that feed into other variables from the second and third blocks such as educyr98 (i), kid92 (d), and Dlogexpadj (D) to name a few. The statistically significant edges identified within a block—for instance, variables i and d—are undirected and are represented by the solid lines. Similarly, variables r and v are contained within the same block but the edge connecting these variables is not statistically significant at the 5% level as indicated by the dashed line connecting these variables. The last block, named "D" with label Dlogexpadj, indicates that our dependent variable (difference in logged expenditure per capita) does not have an edge that connects into any other variable, as set by the user. Significant links and their graphical representation reveal that variables medyr98M, fatherlive, educyr98, borr92, urban98, nosick92, lend92, kid92, domgive, hhsize92, and domrem92 have directed edges into the dependent variable.

In past analyses of this dataset where regressions were conducted for the difference in logged expenditure per capita, all independent variables were on the same level, all allowed to potentially connect into the dependent variable. By contrast, our MIM analysis implies, for example, that the urban location of a household (in 98) links into the dependent variable both directly and via the variable on the number of years of education of the head of household.

We conclude this section with a discussion of the pros and cons of the MIM software and a reference to past published work based on a MIM model. The combination of command line and graphical user interface works well. The user interface is mostly command-line, not graphical, but is straightforward to use, and the output is in both textual and graphical form, depending on

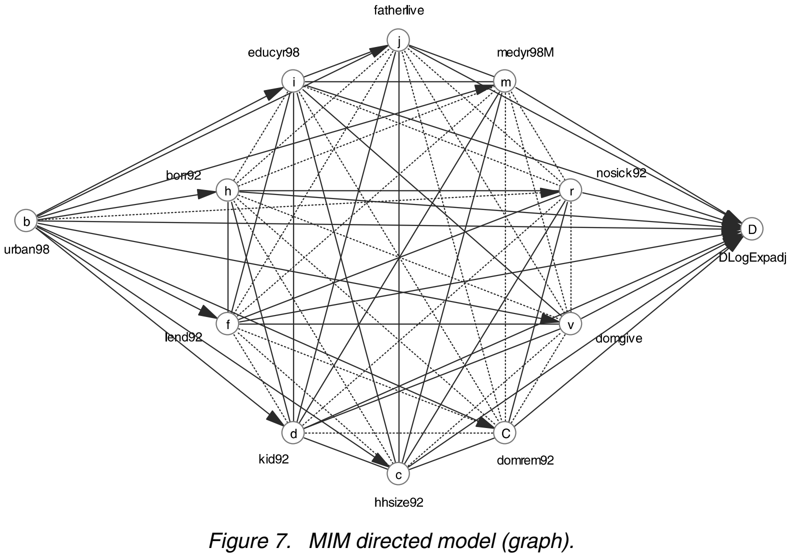


Figure 7. MIM directed model (graph).

the particular tool. Bringing raw data into MIM is a bit awkward; it would be somewhat easier if variable names could be read in directly when entering the data.

The researcher can use background knowledge for blocking the variables, which is helpful. On the other hand, that implies that the direction of the links has to be provided by the user, and in that respect MIM differs from the two other packages. A graph can include both discrete and continuous variables and it may be undirected, directed, or chained (a combination of undirected and directed). MIM readily provides clear and attractive graphs to represent estimated models, as seen above. MIM implements a range of statistical techniques, including maximum likelihood estimation, hypothesis testing, and model selection, and as such is quite flexible; for example, one model can be used to model links from one block to another, and another model to model links between another pair of blocks.

We refer the reader to an interesting published article where a MIM model was natural because background knowledge did make it possible for the researchers to define suitable MIM blocks. In Mohamed, Diamond, and Smith (1998), the authors investigated determinants of infant mortality in Malaysia. The MIM analysis made it possible to identify intermediate and direct determinants of infant mortality. For instance, the year of birth of the child was found to act as a direct as well as indirect (via prematurity and type of drinking water) determinant of infant mortality.

3. TETRAD III AND IV

This section describes the application of Tetrad III and IV to our dataset. The results from both versions of Tetrad are identical, up to a very small difference (which will be pointed out below) probably due to a slight difference in the tests applied or to round-off error (as determined by a personal communication with Clark Glymour, one of the authors of Tetrad). Tetrad III is command-based and runs somewhat faster, while Tetrad IV provides a GUI, inclusive of a graphical representation of the DAG built by the package. Note that Tetrad IV requires Java Runtime Environment version 1.4.1 or higher.

3.1 Introduction and Data Preparation

Tetrad is a package created by Spirtes, Glymour, and Scheines (1993) at Carnegie Mellon University. For an overview of this methodology, we refer the reader to the Tetrad homepage (Tetrad 2006), where the software can be obtained at no cost (by following the Tetrad III link and downloading—for Windows—the file wintet3.exe, and by following the instructions on the Tetrad IV link). It works on any reasonably current PC or SPARCstation. It estimates directed acyclic graphs and, under some very restrictive conditions, derives causal models from data. Conceptually, the relationships between variables to be considered can be summarized in a graph, where a directed arrow links x to y if x is a cause of y.

For an introduction to applications of the Tetrad methodology, we refer the reader to the work of Bessler and Loper (2001) in economics. In particular, Bessler (2003) serves as a good introduction to the subject. Tetrad cannot in general infer causality because that would require, among other assumptions, a very strong assumption of "causal sufficiency," which would require any common cause of any pair of variables in the set of variables to also be part of this set. This is very seldom the case. However, even in cases where causal sufficiency does not hold, Tetrad has been used to identify pairs of variables which are dependent given all other variables in the set, and to infer the existence of directional links between variables.

Bessler and Loper (2001) illustrated the advantages of using DAG techniques. In modeling the GDP (gross domestic product) growth of a set of developing countries, the authors found that openness to trade connects directly to GDP growth, but that agricultural productivity growth connects to GDP growth only via openness to trade (at level 0.20) or not at all (at level 0.10). The authors were thus able to shed light on past assumptions in the economics literature about the development process.

Data can be entered into Tetrad either in the form of a raw flat file in standard format, or in the form of a correlation matrix if all variables are assumed to be continuous as is the case here. It is possible to build a Tetrad model on the basis of categorical data, in which case the data need to be entered as a table of cell counts or as a raw file. In that latter case, restrictions exist on the number of categories (a maximum of 10) in each variable.

Figure 8a describes the input file needed for Tetrad III in the case of continuous variables and Figure 8b displays the user interface for entering data into Tetrad IV. The Tetrad III code is straightforward, and well documented in the Tetrad III user manual, which is available with the package. After the correlation matrix, a section on "temporal knowledge" allows the user to indicate which variables should precede other variables in directional links. Figure 8c displays the user interface for entering this temporal knowledge into Tetrad IV. We do not allow any variable to link into urban98, age (or its square), or gender, and we do not allow our dependent variable to link into any other variable. Aside from these restrictions, Tetrad first establishes which pairs are dependent conditionally on other variables, and then uses an algorithm known as the PC algorithm (described in an Appendix in the Tetrad III user manual) to derive directions for links. Interestingly, Tetrad in fact estimates a class of models compatible with the data, and proposes a directional link (denoted by $- >$) when all models in the class agree on the direction. If they do not, Tetrad reports a link with an unclear direction, denoted as —. When Tetrad determines that a pair of variables probably has a "latent cause" (a variable linking into both variables in the pair but not available in the dataset), it uses the notation $<>$.

Dealing with input and output files in Tetrad III is straightforward, as well as running Tetrad IV and accessing its log files (where the text output resides).

3.2 Main Ideas of the Algorithm

The Tetrad III Build module and the Tetrad IV analysis used in this article both rely on the PC algorithm, which assumes *causal sufficiency*, stating that any latent cause common to two variables in the DAG must also be part of the DAG. When causal sufficiency is not assumed, Tetrad (III or IV) uses a more complicated algorithm, the FCI algorithm, described by Spirtes, Glymour, and Scheines (1993, chap. 6).

We give here a brief idea of how the PC algorithm works. First, a complete undirected graph is created with each variable

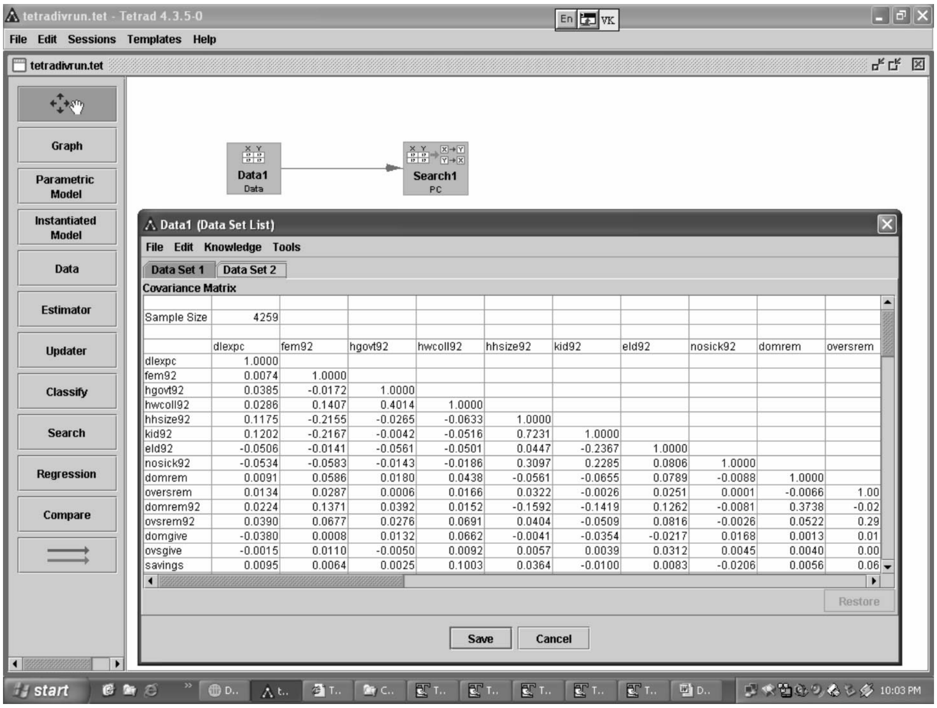
```
/covariance
4259
dlexpc fem92 hgovt92 hwcoll92 hhsize92 kid92 eld92 nosick92 domrem oversrem
domrem92 ovsrem92  domgive ovsgive savings sborr92 borr92 lend92 age age2
edyr98 urban98  medy98i medyr98m fedy98i fedyr98m fathlive mothlive fawc92
mowc92
1
0.0074     1
0.0385     -0.0172
0.0286     0.1407      0.4014     1
. . .
. . .
. . .

0.0325    0.0452     0.0493     0.1708     -0.019     -0.0406     -0.0542    -0.0164    0.0541     0.0409     0.0594     0.1103
          0.0274     0.0097     0.0499     -0.0004    -0.0246     0.0084     -0.0154    -0.0179    0.1575     0.2203     0.1719
          -0.1156    0.2338     -0.1253    0.0154     0.0236      1
0.0395    0.0857     0.0405     0.1827     0.0231     -0.0305     -0.0356    -0.0092    0.0722     0.0489     0.0659     0.178
          0.0282     -0.0114    0.0636     -0.0024    -0.0654     0.0084     0.041      0.0385     0.074      0.3292     0.0975
         -0.0916     0.0712     -0.0725    -0.0301    -0.0422     0.3361     1

/knowledge
addtemporal
1  fem92  age age2 urban98
2  hgovt92 hwcoll92 hhsize92 kid92 eld92 nosick92 domrem oversrem domrem92
 ovsrem92  domgive ovsgive savings sborr92 borr92 lend92  edyr98
 medy98i medyr98m fedy98i fedyr98m fathlive mothlive fawc92 mowc92
3  dlexpc
```

Figure 8a. Input of data into Tetrad. Input file for Tetrad III for continuous variables.

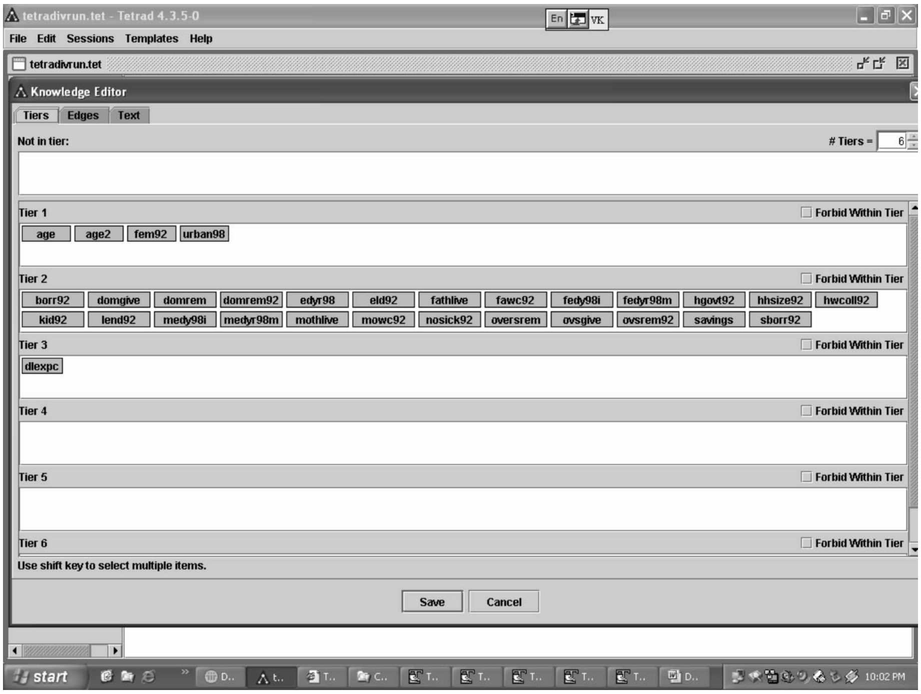
corresponding to a vertex. Then, edges are removed in pairs with variables which are independent, either unconditionally or conditionally on a subset of the remaining variables. Independence is tested with standard correlation tests (for continuous data assumed to be multivariate normal) and by using a G^2 test of independence in contingency tables (for categorical data). We note that Tetrad currently allows for either continuous data, or categorical data, but not for a mixture of both types of data. In that latter case, the user would need to provide information on which pairs of variables are independent conditionally on other



Covariance Matrix

Sample Size	4259									
	dlexpc	fem92	hgovt92	hwcoll92	hhsize92	kid92	eld92	nosick92	domrem	oversrem
dlexpc	1.0000									
fem92	0.0074	1.0000								
hgovt92	0.0385	-0.0172	1.0000							
hwcoll92	0.0286	0.1407	0.4014	1.0000						
hhsize92	0.1175	-0.2155	-0.0265	-0.0633	1.0000					
kid92	0.1202	-0.2167	-0.0042	-0.0516	0.7231	1.0000				
eld92	-0.0506	-0.0141	-0.0561	-0.0501	0.0447	-0.2367	1.0000			
nosick92	-0.0534	-0.0583	-0.0143	-0.0186	0.3097	0.2285	0.0806	1.0000		
domrem	0.0091	0.0586	0.0180	0.0438	-0.0561	-0.0655	0.0789	-0.0088	1.0000	
oversrem	0.0134	0.0287	0.0006	0.0166	0.0322	-0.0026	0.0251	0.0001	-0.0066	1.00
domrem92	0.0224	0.1371	0.0392	0.0152	-0.1592	-0.1419	0.1262	-0.0081	0.3738	-0.02
ovsrem92	0.0390	0.0677	0.0276	0.0691	0.0404	-0.0509	0.0816	-0.0026	0.0522	0.29
domgive	-0.0380	0.0008	0.0132	0.0662	-0.0041	-0.0354	-0.0217	0.0168	0.0013	0.01
ovsgive	-0.0015	0.0110	-0.0050	0.0092	0.0057	0.0039	0.0312	0.0045	0.0040	0.00
savings	0.0095	0.0064	0.0025	0.1003	0.0364	-0.0100	0.0083	-0.0206	0.0056	0.06

Figure 8b. Input of data into Tetrad. User interface for reading data into Tetrad IV.



Not in tier:

Tiers = 6

Tier 1 □ Forbid Within Tier

age age2 fem92 urban98

Tier 2 □ Forbid Within Tier

borr92 domgive domrem domrem92 edyr98 eld92 fathlive fawc92 fedy98i fedyr98m hgovt92 hhsize92 hwcoll92
kid92 lend92 medy98i medyr98m mothlive mowc92 nosick92 oversrem ovsgive ovsrem92 savings sborr92

Tier 3 □ Forbid Within Tier

dlexpc

Tier 4 □ Forbid Within Tier

Tier 5 □ Forbid Within Tier

Tier 6 □ Forbid Within Tier

Use shift key to select multiple items.

Figure 8c. Input of data into Tetrad. Temporal knowledge in Tetrad IV.

variables. Tetrad differs from the other packages reviewed in this respect.

In order to orient the surviving links, the PC algorithm proceeds as follows: for each triplet x, y, z such that both pairs (x, y) and (y, z) are linked but the pair (x, z) is not linked, if y does not appear in any set which, when conditioned on, makes x and z independent, then the triplet x, y, z is oriented as $x \rightarrow y \leftarrow z$, which makes y a collider. Once all such colliders are identified, the algorithm proceeds as follows: if $x \rightarrow y$, y and z are linked and x and z are not linked, and if there is no arrowhead at y, then (y, x) is oriented as $y \rightarrow z$. We recommend Appendix B of the Tetrad III user manual to users who would like to follow how the algorithm unfolds on a particular example.

3.3 Reporting Results

Tetrad III output (displayed in Figure 9a) consists of a listing of the correlation matrix and its p values, and a summary of user choices, for example, forbidden links. A list follows of all removed pairs (the first few removals are italicized in Figure 9a) , and of the conditioning variables, from the unconditional to the conditional on more and more variables. The user can then identify whether a pair was removed because of a nonsignificant unconditional correlation, or because of a nonsignificant conditional correlation. The output also identifies the variables that were used in the conditioning. Finally a list follows with three types of links which emerged from the PC algorithm (links into our dependent variable are italicized in Figure 9a). Links where the orientation can be established from the algorithm are marked with a directed arrow. Links where the orientation is ambiguous are marked with an m-dash: —. Links marked with "<>" have a common latent cause according to Tetrad.

Tetrad IV allows the user to request text output with various levels of details. At the highest level of detail, the Tetrad IV output is very similar to that of Tetrad III, but also includes useful information on the orientation of edges and the creation of colliders. An abbreviated display of this output is provided in Figure 9b. Our Tetrad IV analysis yields a DAG almost identical to that of Tetrad III, except for the absence in Tetrad IV of the link between lend92 and dlexpc (these two variables are determined to be independent conditionally on nosick92 and domgiv, with a p value of 0.0635, see italicized row in Figure 9b).

We note the following pros and cons of Tetrad. The command-line Tetrad III interface is simple, and the GUI in Tetrad IV is attractive and straightforward. Data input is fairly easy; one needs only a correlation matrix or raw data and optionally a few parameters for background knowledge. The output is easy to interpret, as shown above.

The GUI in Tetrad IV is very handy; in particular we display in Figure 9c the graphical representation of the DAG provided by Tetrad IV, and note that it is very easy to move the labels around by clicking and dragging, as we have done here. Apart from the choice of a significance level (0.05 in our case), a depth parameter (0 or -1) can be selected by the user; a value of 0 means that Tetrad IV will eliminate edges by looking at only unconditional dependence among pairs of variables.

The learning curve for using all the tools and options properly is somewhat steep because of the variety of modeling tools and the delicate choices between algorithms a user faces. Of course, this tends to arise because of the inherent difficulty in building DAGs in general. The error messages are perhaps a bit terse for some of the tools.

```
Output file: c:\causality\tetrad3.out
Data file: c:\causality\tetrad3.dat
Temporal Tier file: c:\causality\tetrad3.dat

Parameters:
  Sample Size: 4259
  Continuous Data

Covariance Matrix
  dlexpc    fem92     hgovt92   hwcoll92  hhsize92  kid92     eld92
nosick92  domrem    oversrem  domrem92  ovsrem92  domgive   ovsgive   savings
sborr92   borr92    lend92    age       age2      edyr98    urban98   medy98i
medyr98m  fedy98i   fedyr98m  fathlive  mothlive  fawc92    mowc92
 1.0000
 0.0074    1.0000
. . .

Correlation Matrix
  dlexpc    fem92     hgovt92   hwcoll92  hhsize92  kid92     eld92
nosick92  domrem    oversrem  domrem92  ovsrem92  domgive   ovsgive   savings
sborr92   borr92    lend92    age       age2      edyr98    urban98   medy98i
medyr98m  fedy98i   fedyr98m  fathlive  mothlive  fawc92    mowc92
 1.0000
 0.0074    1.0000
. . .

P-value for Correlations
  dlexpc    fem92     hgovt92   hwcoll92  hhsize92  kid92     eld92
nosick92  domrem    oversrem  domrem92  ovsrem92  domgive   ovsgive   savings
sborr92   borr92    lend92    age       age2      edyr98    urban98   medy98i
medyr98m  fedy98i   fedyr98m  fathlive  mothlive  fawc92    mowc92
 0.0000
 0.6294    0.0000
 . . .

Significance:      0.0500
  Settime:          Unbounded

Forbidden edges:

    dlexpc -> fem92   dlexpc -> hgovt92   dlexpc -> hwcoll92   dlexpc -> hhsize92
    dlexpc -> kid92   dlexpc -> eld92
    dlexpc -> nosick92   dlexpc -> domrem   dlexpc -> oversrem   dlexpc -> domrem92   dlexpc -> ovsrem92
. . .
```

Figure 9a. Tetrad output. Tetrad III partial output.

There are more tools in Tetrad than were tested in this review where we focused on the DAG building capabilities of the packages; they are detailed in the readily available user manuals.

4. WINMINE

4.1 Introduction and Data Preparation

WinMine was created by Microsoft Corporation, and a noncommercial version can be obtained for free at *http://research.microsoft.com/~dmax/winmine/tooldoc.htm*. It works on any reasonably current PC. The WinMine package constructs a variety of statistical models, including DAGs, from data.

4.1.1 Loading Data

This package has a Data Conversion Wizard that helps get data into WinMine-readable format. It permits the user to use three different data-source formats: SQL Server Database, Raw text file, and DST text file. The format of the input file needs to be comma-, tab-, or space-delimited with or without column headers (names). An output file name is specified using the .xdat extension. Once the source, input and output files are specified an input file characteristics dialog box pops up to provide the user with an opportunity to modify the file characteristics. The user interface is presented in Figure 10.

The wizard typically will correctly “guess” whether the first row contains column names, but if it does not “First Row Contains Variable Names” can be checked. Similarly, if the

```
Temporal Tier: 1
fem92 age age2 urban98
Temporal Tier: 2
hgovt92 hwcoll92 hhsize92 kid92 eld92 nosick92 domrem oversrem domrem92 ovsrem92 domgive
ovsgive savings sborr92 borr92 lend92 edyr98 medy98i medyr98m fedy98i fedyr98m fathlive mothlive
fawc92 mowc92
Temporal Tier: 3
dlexpc
}
{-------------------------------------------------------
List of vanishing (partial) correlations that made
TETRAD remove adjacencies.

  Corr. :  Sample (Partial) Correlation
  Prob. :  Probability that the absolute value of the sample
           (partial) correlation exceeds the observed value,
           on the assumption of zero (partial) correlation in
           the population, assuming a multinormal distribution.

Edge             (Partial)
Removed          Correlation                 Corr.    Prob.
-------          -----------                 -----     -----
Edge             (Partial)
Removed          Correlation                 Corr.    Prob.
-------          -----------                 -----     -----
dlexpc - fem92  rho(dlexpc fem92)            0.0074   0.6294
dlexpc - hwcoll92rho(dlexpc hwcoll92)        0.0286   0.0620
dlexpc - domrem rho(dlexpc domrem)           0.0091   0.5533
dlexpc - oversremrho(dlexpc oversrem)        0.0134   0.3824
. . .
. . .
. . .
fem92 - ovsgive rho(fem92 ovsgive)           0.0110   0.4734
fem92 - savings rho(fem92 savings)           0.0064   0.6764
fem92 - sborr92 rho(fem92 sborr92)           0.0215   0.1612
fem92 - fedy98i rho(fem92 fedy98i)           0.0044   0.7740
. . .

#: no orientation consistent with assumptions


Significance Level =  0.0500
/Pattern
hhsize92 -> dlexpc
kid92   -> dlexpc
nosick92 -> dlexpc
lend92  -> dlexpc
edyr98  -> dlexpc
urban98 -> dlexpc
medyr98m -> dlexpc
. . .
. . .
. . .
eld92   <> nosick92
eld92   <> ovsrem92
age     -> eld92
age2    -> eld92
. . .
```

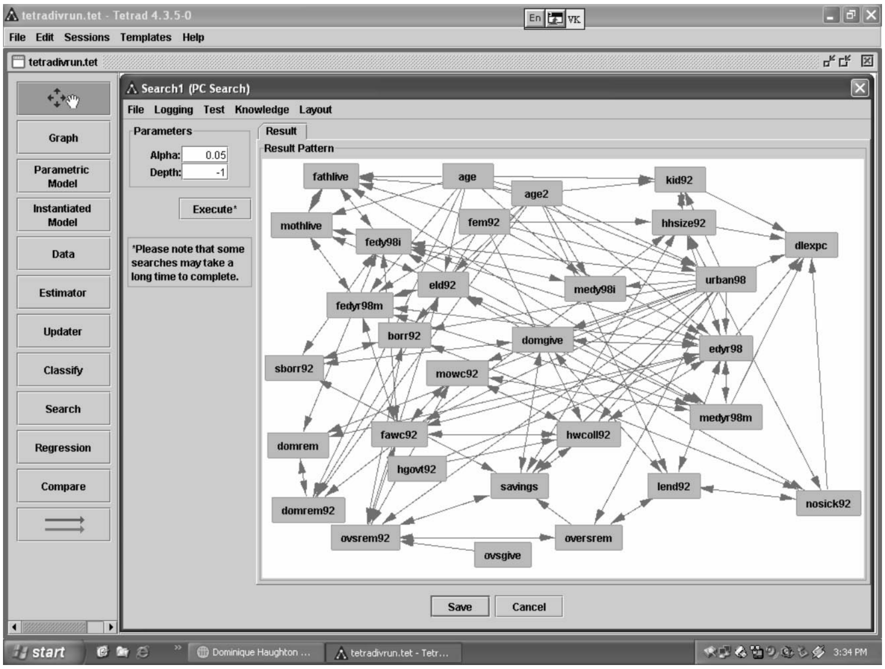
Figure 9a (continued). Tetrad output. Tetrad III partial output.

```
Starting PC algorithm.
Independence test = Fisher's Z, alpha = 0.0500.
Starting Fast Adjacency Search.
Depth = Unlimited
Independence accepted: dlexpc _||_ fem92 p = 0.6293
Independence accepted: dlexpc _||_ hwcoll92 p = 0.0620
. . .
Independence accepted: dlexpc _||_ lend92 | nosick92, domgive p = 0.0635
. . .
Independence accepted: age _||_ edyr98 | hgovt92, hwcoll92, domgive, lend92, age2, urban98, fedyr98m
p = 0.0568
Finishing Fast Adjacency Search.
Staring PC Orientation.
Starting BK Orientation.
Orienting edge: age2 --> kid92
Orienting edge: urban98 --> oversrem
. . .
Finishing BK Orientation.
Starting Collider Orientation:
Orienting collider: edyr98 *-> dlexpc <-* nosick92
Orienting collider: edyr98 *-> dlexpc <-* kid92
Orienting collider: medyr98m *-> dlexpc <-* hhsize92
. . .
Returning this graph:

Nodes:
1. dlexpc
2. fem92
3. hgovt92
. . .
Edges:
1. age --- age2
2. urban98 --> medy98i
3. urban98 --> hwcoll92
4. age --> mothlive
. . .
97. fawc92 <-> mowc92

Elapsed time = 20.43 s
Finishing PC algorithm.
```

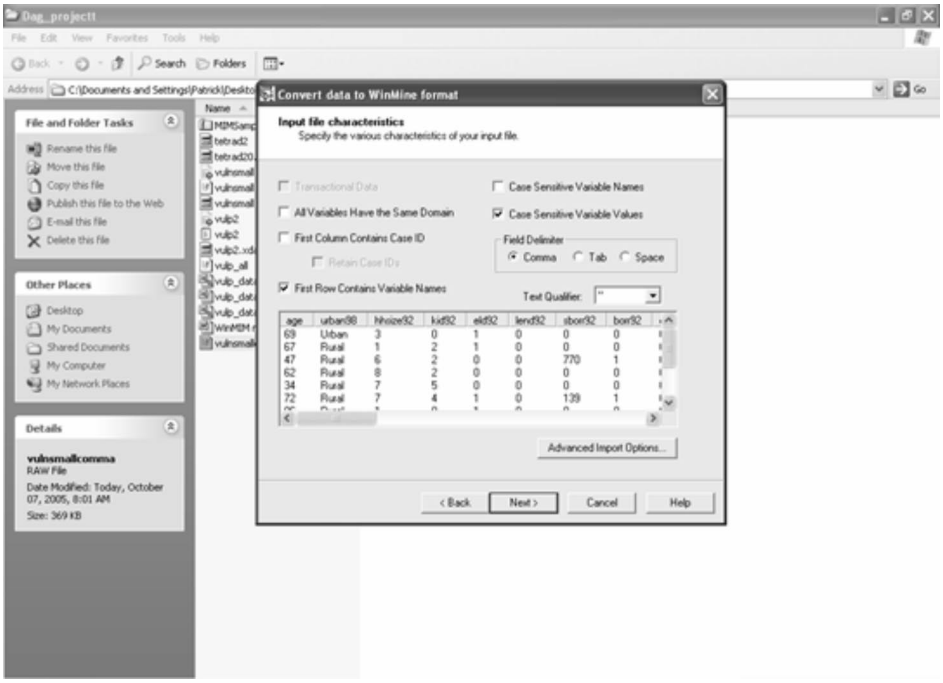
Figure 9b. Tetrad output. Tetrad IV partial text output.



Alpha: 0.05
Depth: -1

*Please note that some searches may take a long time to complete.

Figure 9c. Tetrad output. Tetrad IV graphical representation of DAG.



vulnsmallcomma
RAW File
Date Modified: Today, October 07, 2005, 8:01 AM
Size: 369 KB

Convert data to WinMine format

Input file characteristics
Specify the various characteristics of your input file.

□ Transactional Data
□ All Variables Have the Same Domain
□ First Column Contains Case ID
□ Retain Case IDs
☑ First Row Contains Variable Names
□ Case Sensitive Variable Names
☑ Case Sensitive Variable Values

Field Delimiter
● Comma ○ Tab ○ Space

Text Qualifier: "

age	urban98	hhsize92	kid92	eld92	lend92	sborr92	borr92
69	Urban	3	0	1	0	0	0
67	Rural	1	2	1	0	0	0
47	Rural	6	2	0	0	770	1
62	Rural	8	2	0	0	0	0
34	Rural	7	5	0	0	0	0
72	Rural	7	4	1	0	139	1

Figure 10. WinMine data conversion wizard.

wizard does not correctly "guess" the delimited format of the file, the user can check the appropriate box. The next screen scans the data and guesses whether the variable type is numerical or categorical. Once the data are defined, the user is given an opportunity to review and change the variable type by launching the Variable Editor application. In this dialogue box, the user can change variable types to numerical, categorical, or select to not import the variable. This dialogue box also provides the user with an opportunity to change the names of variables. The wizard allows the user to define how null states are labeled. Once satisfied with the variable definitions, the user can click through the data conversion process and finish the data conversion wizard. WinMine's data conversion tool makes it easy to import data. In addition, WinMine provides data join and data split tools which allow the user to merge datasets, and to splits datasets into analysis and validation files.

4.1.2 Model Preparation

WinMine provides a GUI interface tools package—PlanEditor—to build a plan that instructs the learning algorithm how to model each variable. The learning algorithm can use three pieces of information for each variable.

The first piece of information is each variable's role in the model: an input variable, output variable, input-output variable, marginal-ignored variable, marginal-input variable, or ignored. An input-variable is one used to predict other variables and an output variable is one that is only predicted by other variables. An input-output variable is both predicted and used to predict other variables. Ignored variables are not used in the model. Marginal-ignored and marginal-input variables are similar to ignored or input, but a marginal distribution is constructed for designated variables.

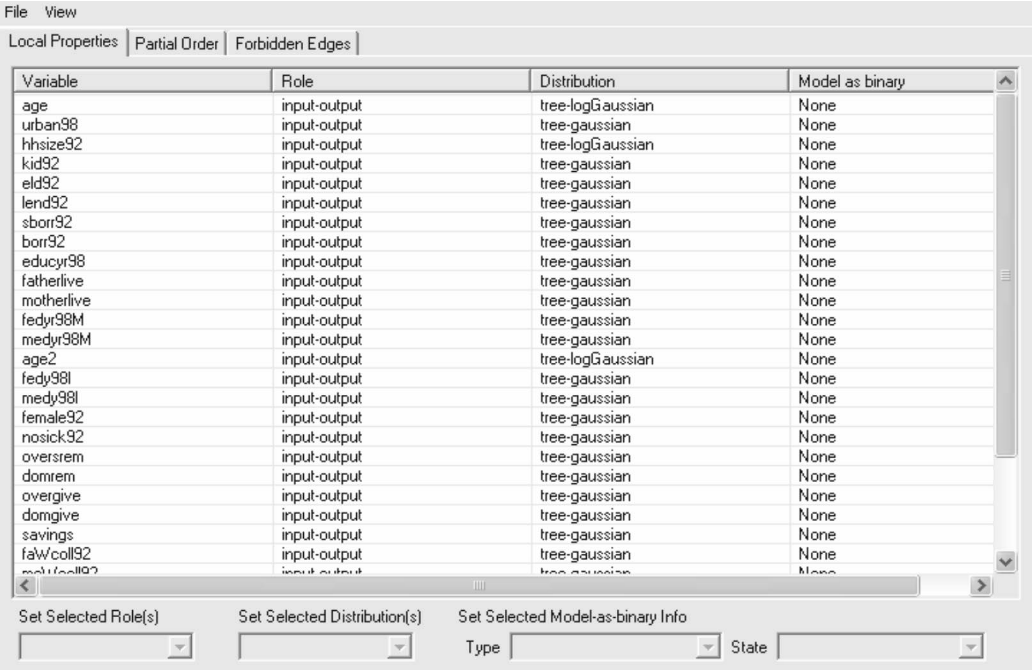
The second piece of information used by the learning algorithm to build a model is to specify a distribution for each variable. Distributions are represented by either a table or a tree and a specific functional form for the distribution. A user can specify either a multinomial or binary-multinomial distribution for discrete variables. Continuous variables can be modeled with the following distributions: Gaussian, binary-Gaussian, log-Gaussian, or binary-log-Gaussian. The "binary" version of the leaf distributions allows the user to model missing versus not missing, and then to use the given distribution on the nonmissing part.

Finally, when modeling binary variables (tree distributions only) the user can request that the learning algorithm focus on binary versions of variables. This may occur, for example, when one is modeling a continuous variable that has both missing and nonmissing values. The model-as-binary information option would model the missing versus nonmissing status of the continuous variable.

WinMine's PlanEditor toolkit permits users to specify partial-order constraints for a Bayesian network or other constraints that forbid the learning algorithm from considering certain dependencies, that is, forbidden edges. When PlanEditor is not used prior to running a model, all variables are specified as input-output variables, tree distributions are used for all variables, and the functional distribution is automatically selected using the values in the data.

4.1.3 Model Specification

Two learning models are possible using the WinMine package: a dependency network or a Bayesian network. A dependency network consists of a set of conditional probabilities for



Variable	Role	Distribution	Model as binary
age	input-output	tree-logGaussian	None
urban98	input-output	tree-gaussian	None
hhsize92	input-output	tree-logGaussian	None
kid92	input-output	tree-gaussian	None
eld92	input-output	tree-gaussian	None
lend92	input-output	tree-gaussian	None
sborr92	input-output	tree-gaussian	None
borr92	input-output	tree-gaussian	None
educyr98	input-output	tree-gaussian	None
fatherlive	input-output	tree-gaussian	None
motherlive	input-output	tree-gaussian	None
fedyr98M	input-output	tree-gaussian	None
medyr98M	input-output	tree-gaussian	None
age2	input-output	tree-logGaussian	None
fedy98I	input-output	tree-gaussian	None
medy98I	input-output	tree-gaussian	None
female92	input-output	tree-gaussian	None
nosick92	input-output	tree-gaussian	None
oversrem	input-output	tree-gaussian	None
domrem	input-output	tree-gaussian	None
overgive	input-output	tree-gaussian	None
domgive	input-output	tree-gaussian	None
savings	input-output	tree-gaussian	None
faWcoll92	input-output	tree-gaussian	None

Figure 11. WinMine plan editor.

all output or input-output variables. In a similar way, a Bayesian network corresponds to a joint distribution of the variables such that no cycle appears in the network.

4.2 Main Ideas of the Algorithm

The main ideas for the WinMine algorithm were provided to us by Max Chickering, the author of WinMine, in a personal correspondence. We focus here on the algorithm based on decision trees; one alternative is the algorithm based on contingency tables when all variables are categorical, but the decision tree approach allows for more types of data. For decision-tree distributions, one can either learn a dependency network or a Bayesian network, the only difference being the acyclicity constraint. Decision trees are grown greedily using hold-one-out splits for discrete predictors, and a quantile method for continuous predictors (see Chickering, Meek, and Rounthwaite 2006).

At any stage of the model-building process, a split is added to a decision tree predictive of a particular variable to optimize

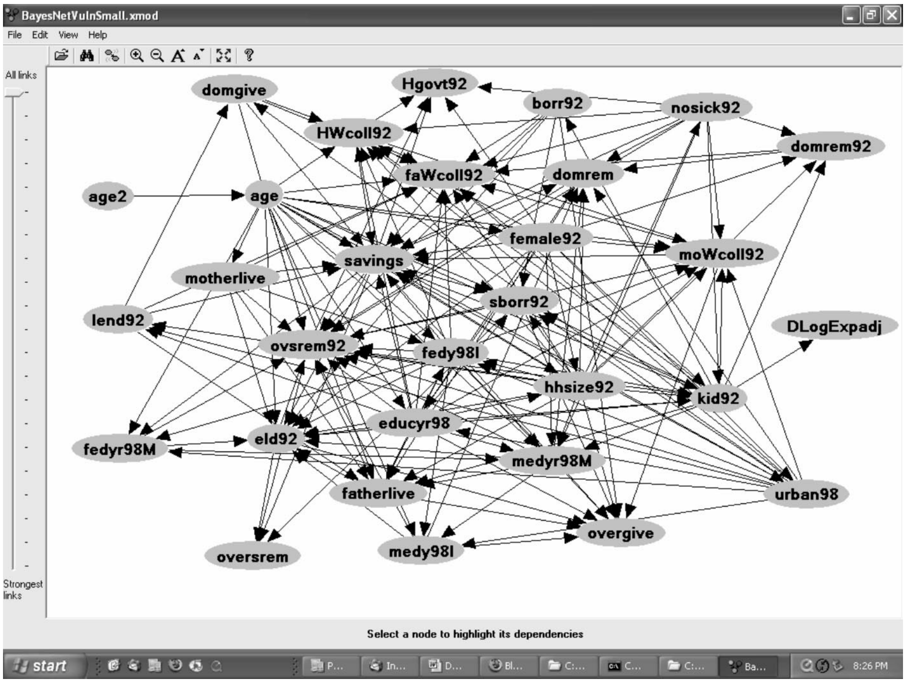


Figure 12. WinMine graphical output.

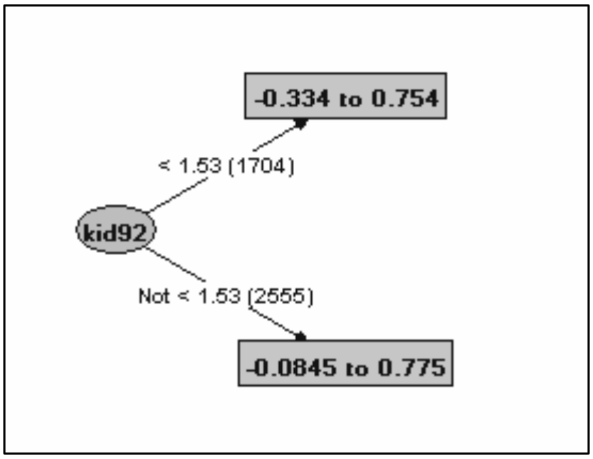


Figure 13. Result of double-clicking DLogExpadj.

a scoring function selected by the user, all the while making sure that the added split does not create a cycle (as in the case of tables, one can use a number of different scoring functions). Note that this approach (aside from the acyclicity constraint) is similar to that used in decision trees algorithms such as CART (classification and regression trees) and CHAID (chi-squared automatic interaction detector), albeit with a potentially different objective function, and is reminiscent of the process used in a stepwise regression.

4.3 Reporting Results

Figure 12 presents graphical output from WinMine. Although not shown here, clicking on a node shows its antecedents, consequents, and nodes that can be both. Clicking on kid92 shows that DlogExpAdj is only one of 12 consequents. WinMine's color-coded graphs can reveal immediate patterns in a convenient way. Note that WinMine identified only one link into our dependent variable, namely kid92. This is somewhat counter-intuitive, a point we will return to in the conclusion.

The output of the model is viewable in the DnetBrowser, which provides a decision-tree diagram for each variable in the domain. The output displays a set of all trees with arrows showing the direction of the relationships. Double-clicking on any variable provides a detailed leaf distribution (Figure 13).

We note that the range given for the node in Figure 13 lacks summary statistics.

WinMine, as other packages reviewed here, is easy to use. It is easy to split a dataset into training and testing subsets. There is a graphical slider for dynamically adjusting the statistical significance level. However, it does not show tick marks for the significance levels (0.05, 0.01, 0.005, etc.). It is also easy to explore the decision trees discovered, as they are displayed explicitly when a node is double-clicked. One can provide an explicit, optional plan file for designating distributional assumptions. Without the plan file, WinMine makes intelligent guesses (based on the data) about each variable's role, distribution, and need to model-as-

Table 2. Summary of Features

	Strengths	*Weaknesses*	*Distinctive features*
MIM 3.2	*Graphs may be undirected, directed, chain (a combination of undirected and directed) * Supports maximum likelihood estimation, hypothesis testing, model selection	*Good documentation, but user is well advised to also buy the book. * Directions of links must be provided by the user, via the creation of blocks; directional links are not possible within a block.	* Supports modeling of discrete and continuous variables.
Tetrad III/IV	* Can generate data for input to various SEM packages. * Largest variety of complementary tools * Easy to use GUI in Tetrad IV * Good documentation * Strong functionality, modularity and flexibility	* Significant learning curve to use all the tools and options properly.	* Algorithm optimizes globally and less conservatively than WinMine.
WinMine	* Ease of use	* It is not clear where to set the significance strength slider, as it lacks numeric grid marks.	* Explicit decision trees provided.
	* Ease of dataset splitting	* A large decision tree is hard to print, since fitting it to screen makes fonts too small to read. Must print it in pieces.	* Optional plan file makes intelligent guesses (based on data) about variable role, distribution, and need to model-as-binary.
	* GUI Slider for adjusting statistical significance level * Easy to sort columns and find variables in large graph. * Good tutorial	* Good tools for data checking, plan making.	* With table-distribution Bayesian networks, the browser indicates which edges are reversible or compelled. * Dnetbrowser is excellent for interactive exploration: viewing the whole graph and drilling into individual variables to see their tree.

binary. With table-distribution Bayesian networks, the browser indicates which edges are reversible or compelled, and they are color-coded. In short, Dnetbrowser, the primary tool of WinMine is highly interactive, allowing one to see the network and double-click individual variables to see the classification/decision tree. The GUI makes it easy to perform many modeling iterations, adjust significance levels to an appropriate level of stringency, and drill down into decision trees.

We also note that the user can use background knowledge for partial ordering and link forbidding. Although one could use the DAG discovered by WinMine as input into another package, WinMine is a self-contained package. The data preparation tools are quite handy. The tools for building and viewing dependency or Bayesian networks are useful. There is also a tool for computing the predictive accuracy of a model, which is useful for determining how accurately the model would perform on a new dataset.

5. CONCLUSIONS

Each of the packages is a second-generation statistical tool for modeling complex phenomena where strong theory is missing or in short supply. The differences among the packages lie primarily in the fact that they rely on different algorithms, tending to view the building of DAGs in somewhat different contexts, and secondarily in optional features. MIM supports the modeling of complex graphs, including DAGs, using standard statistical tests. Tetrad and WinMine support the modeling of DAGS. Tetrad was less conservative than WinMine in that even at a significance level of 0.0001, Tetrad found links that WinMine at a significance level of 0.05 did not. We found that Tetrad's algorithm optimizes holistically, whereas WinMine's algorithm optimizes locally, at every node in a graph. Interestingly, WinMine identified only one link into our dependent variable. This is somewhat counter-intuitive, since one would expect from past experience that urban98, for example, should link into the dependent variable.

Interestingly, in past work on this dataset involving spline regressions of the same dependent variable, we found in a parsimonious model that our predictors coincided with the variables Tetrad selected to link into the dependent variable. What one gains from a DAG analysis are some insights regarding indirect predictors that may not appear at all in a traditional regression. For instance, the gender of the head of household does link into the dependent variable as determined by Tetrad, but indirectly via the size of the household, the urban location of the household, and interestingly the number of years of education of the head.

We summarize the various features of the packages in Table 2. Finally, we note that for each of the three packages, the authors have been extremely responsive to questions and queries, by replying promptly to e-mails even on weekends, and even fixing bugs in the program where applicable. We refer the reader to Kevin Murphy's very useful Web site (Murphy 1998), which includes a comparison table featuring a number of packages, including the three reviewed here, as well as references and a good tutorial on Bayesian networks and graphical modeling.

REFERENCES

Awokuse, T. O., and Bessler, D. A. (2003), "Vector Autoregressions, Policy Analysis, and Directed Acyclic Graphs: An Application to the U.S. Economy," *Journal of Applied Economics*, 6, 1–24.

Bessler, D. (2003), "On World Poverty: Causal Graphs from the 1990s," available online at *http://agecon2.tamu.edu/people/faculty/bessler-david/WebPage/FAO2.ppt#1*, accessed March 2, 2006.

Bessler, D., and Loper, N. (2001), "Economic Development: Evidence from Directed Graphs," *Manchester School*, 69, 457–476.

Biedermann, A., Taroni, F., Delemont, O., Semadeni, C., and Davison, A. C. (2005), "The Evaluation of Evidence in the Forensic Investigation of Fire Incidents (Part I): An Approach Using Bayesian Networks," *Forensic Science International*, 147, 49–57.

Chickering, D. M., Meek, C., Rounthwaite, R. (2006), available online at *http://research.microsoft.com/~dmax/publications/splits.pdf*.

Edwards, D. (2000), *Introduction to Graphical Modeling* (2d ed.), New York: Springer-Verlag.

Friedman, N., Linial, M., Nachman, I., Pe'er, D. (2000), "Using Bayesian Networks to Analyze Expression Data," *Journal of Computational Biology*, 7 601–620.

Haughton, D., Haughton, J., and Nguyen, P. (eds.), (2001), "Living Standards During and Economic Boom; the Case of Vietnam, United Nations Development Program and General Statistical Office," Statistical Publishing House, Hanoi. Available online at *http://www.undp.org.vn/undp/docs/2001/living/lse.pdf*, accessed March 2, 2006.

MIM (2005), The MIM Website, *http://www.hypergraph.dk/*, accessed March 2, 2006.

Mohamed, W. N., Diamond, I., and Smith, P. (1998), "The Determinants of Infant Mortality in Malaysia: A Graphical Chain Approach," *Journal of the Royal Statistical Society*, 161, 349–366.

Murphy, K. (1998), "A Brief Introduction to Graphical Models and Bayesian Networks," available online at *http://www.cs.ubc.ca/~murphyk/Bayes/bayes.html*, accessed March 2, 2006.

Mussi, S. (2003), "Providing Websites with Capabilities of One-to-One Marketing," *Expert Systems*, 20, 8–19.

Pearl, J. (2000), *Causality*, Cambridge: Cambridge University Press.

Spirtes, P., Glymour, C., and Scheines, R. (1993), *Causation, Prediction, and Search*, Springer-Verlag Lecture Notes in Statistics 81, New York: Springer Verlag.

Tetrad (2006), Tetrad Project Homepage, *http://www.phil.cmu.edu/projects/tetrad/index.html*, accessed March 2, 2006.

WinMine (2006), WinMine Toolkit Homepage, *http://research.microsoft.com/~dmax/winmine/tooldoc.htm*, accessed March 2, 2006.

Xenos, M. (2004), "Prediction and Assessment of Student Behaviour in Open and Distance Education in Computers Using Bayesian Networks," *Computers and Education*, 43, 345–359.

ADDITIONAL READING

Wikipedia (2006), "Directed Acyclic Graph," Wikipedia, the free encyclopedia, *http://en.wikipedia.org/wiki/Acyclic_directed_graph*, accessed March 2, 2006.

——— (2006), "Graph (mathematics)," Wikipedia, the free encyclopedia, *http://en.wikipedia.org/wiki/Directed_graph*, accessed March 2, 2006.