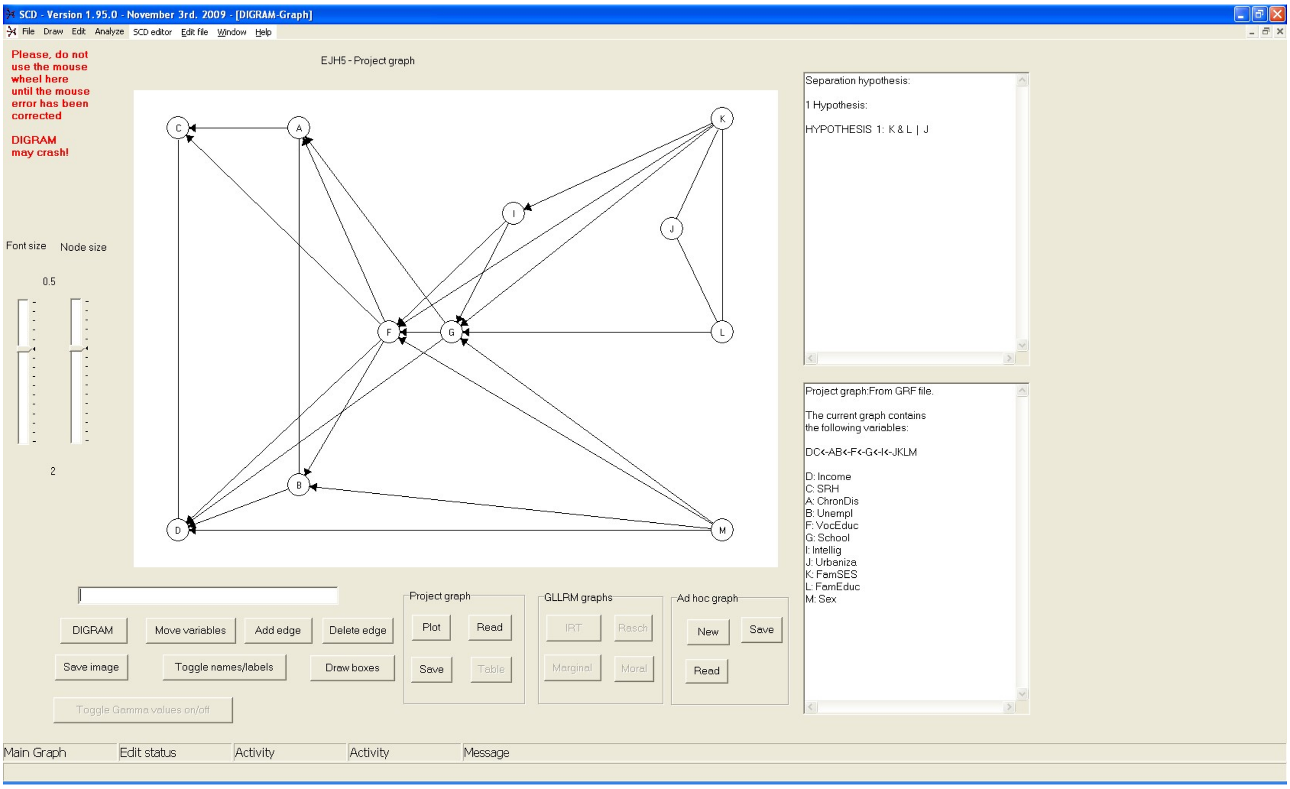
A guided tour through DIGRAM 2.0:

The graph module

Take a look at Figure 1. It shows DIGRAM’s Graph window; the window where you can see the Markov graphs of the statistical models that you work with when you analyze data or – for that matter – any graph that you care to construct with or without data and an underlying statistical model. The model you can see in Figure 1 is the same model that was used (or will be used) as the starting point for the guided tour through DIGRAM’s analysis of multidimensional contingency tables by chain graph models.



Please, do not use the mouse wheel here until the mouse error has been corrected

DIGRAM may crash!

Separation hypothesis:

1 Hypothesis:

HYPOTHESIS 1: K & L | J

Project graph:From GRF file.

The current graph contains
the following variables:

DC<-AB<-F<-G<-I<-JKLM

D: Income
C: SRH
A: ChronDis
B: Unempl
F: VocEduc
G: School
I: Intellig
J: Urbaniza
K: FamSES
L: FamEduc
M: Sex

Figure 1. DIGRAM’s Graph window.

We must remember, that the Markov graph of a graphical model first of all is a mathematical model of the statistical model and that the main idea is to analyzer and reanalyze the graph by graph theoretical algorithms during the analysis in order to identify properties of the probabilistic model that may be useful during the analysis. The visual graph may, however, also have a role to play. First, it may help us to grasp what the model is actually telling us about the data and the variables that we are analyzing if it is drawn up in the right way. And if it is not drawn in the most transparent way, redrawing the graph will also help us focus on the subject matter problems rather than the statistical technicalities of the analysis. And second, you may find it easier to modify the model by clicking on the parts of the model that you want to change.

The purpose of this guided tour is to take you through most of these features.

Displaying the graph

Let us first take a look on the different ways that that graph may look. Figure 1 shows the default view:

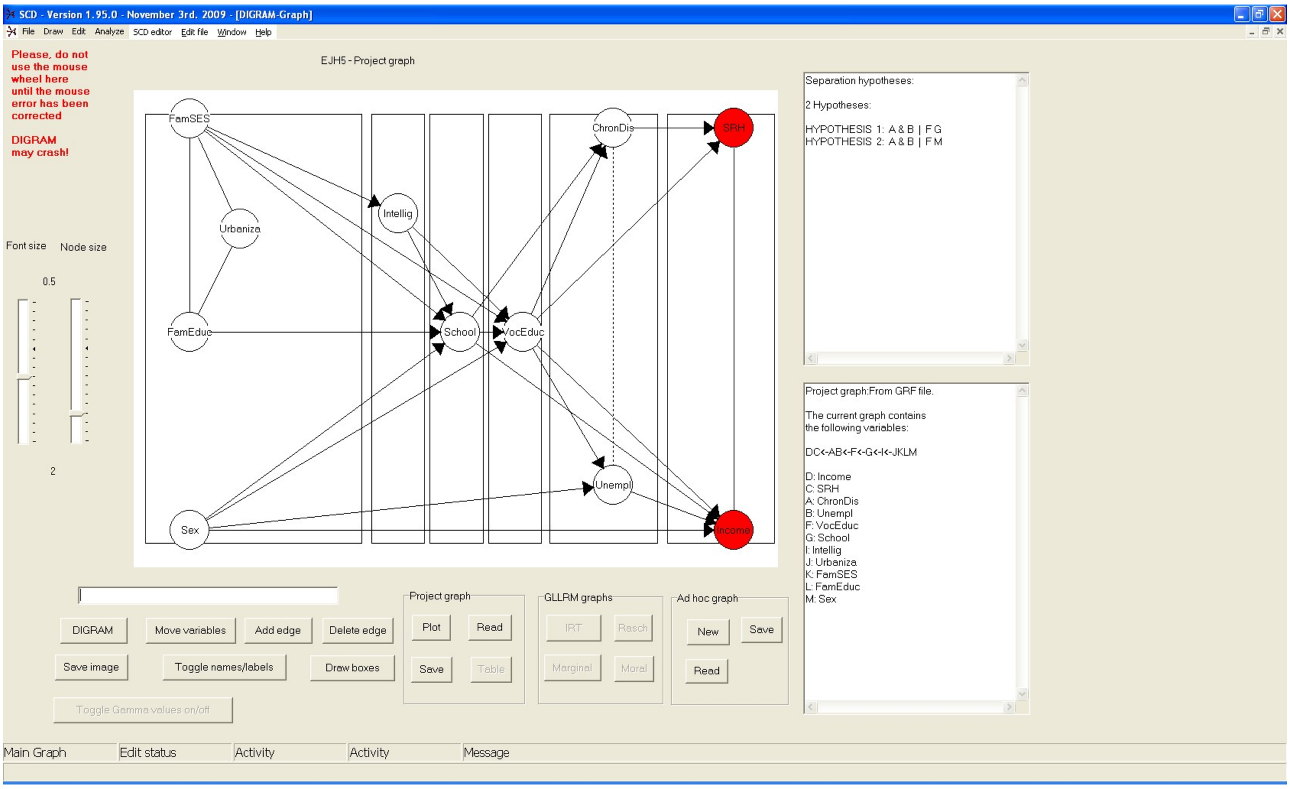
- Variables are shown as nodes with project labels rather than variable names, but you can see the variable names in the text box at the lower right side of the window.
- All nodes are white and all edges are fixed edges.
- The size of the nodes is the minimal size supported by DIGRAM.
- The recursive structure runs from right to left.
- The definition of the recursive block is implicitly defined by the arrows and undirected edges, but the recursive blocks are not explicitly shown.
- The positions of the nodes are given by a set of coordinates that are saved together with other information on the project.GRF file.
- The graph shows all the variables of the model.

All of this and several other things can be changed.

Changing the way variables are shown

- Click on the "Toggle names/labels" button if you prefer to have nodes with variable names rather than project nodes.
- Pull the "Node size" and "Font size" sliders down, if you want the nodes and names to be larger.
- Click on the "Draw boxes" button if you want DIGRAM to create boxes for each recursive block of the model.
- Invoke the **FLIP** command if you want to change the recursive direction.
- Use the **COLOR** or the **NODECOLOR** command if you want to change the colour of some of the nodes. (These commands are described below)
- Use the **HORIZONTAL** and **VERTICAL** commands to align edges (These commands are also described below).
- Use the **DOT** and **SOLID** commands to replace solid lines with dotted lines or vice versa.

Figure 2 shows the same model as Figure 1, where all of the above features have been used.



Separation hypotheses:

2 Hypotheses:

HYPOTHESIS 1: A & B | F G
HYPOTHESIS 2: A & B | F M

Project graph:From GRF file.

The current graph contains
the following variables:

DC<-AB<-F<-G<-I<-JKLM

D: Income
C: SRH
A: ChronDis
B: Unempl
F: VocEduc
G: School
I: Intellig
J: Urbaniza
K: FamSES
L: FamEduc
M: Sex

Figure 2. A different view of the model in Figure 1 with recursive blocks, a flipped recursive direction, coloured nodes and a dotted edge between A and B

The commands that you may use to change the appearance of the graph are summarized below

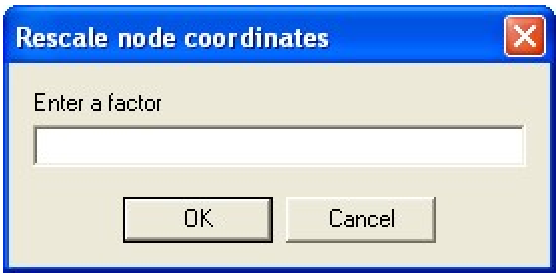
Command	Parameters	Effect
FLIP	None	Changes the recursive direction of the graph
COLOR	Variables	Shows a palette from which you may select new node colours
NODECOLOR	Variables colour code	Assign colours to nodes
HORIZONTAL	Variables	Puts all variables on a horizontal line
VERTICAL	Variables	Puts all variables on a vertical line
DOT	Variable pairs	Replaces solid lines with dotted lines
SOLID	Variable pairs	Replaces dotted lines with solid lines

Changing the position of nodes

To change the position of a node, you first click on the "Move" button, click on the node (the node colour changes to black), and then click on the new position. Note that you cannot pull the node to the new position.

Rescaling graphs

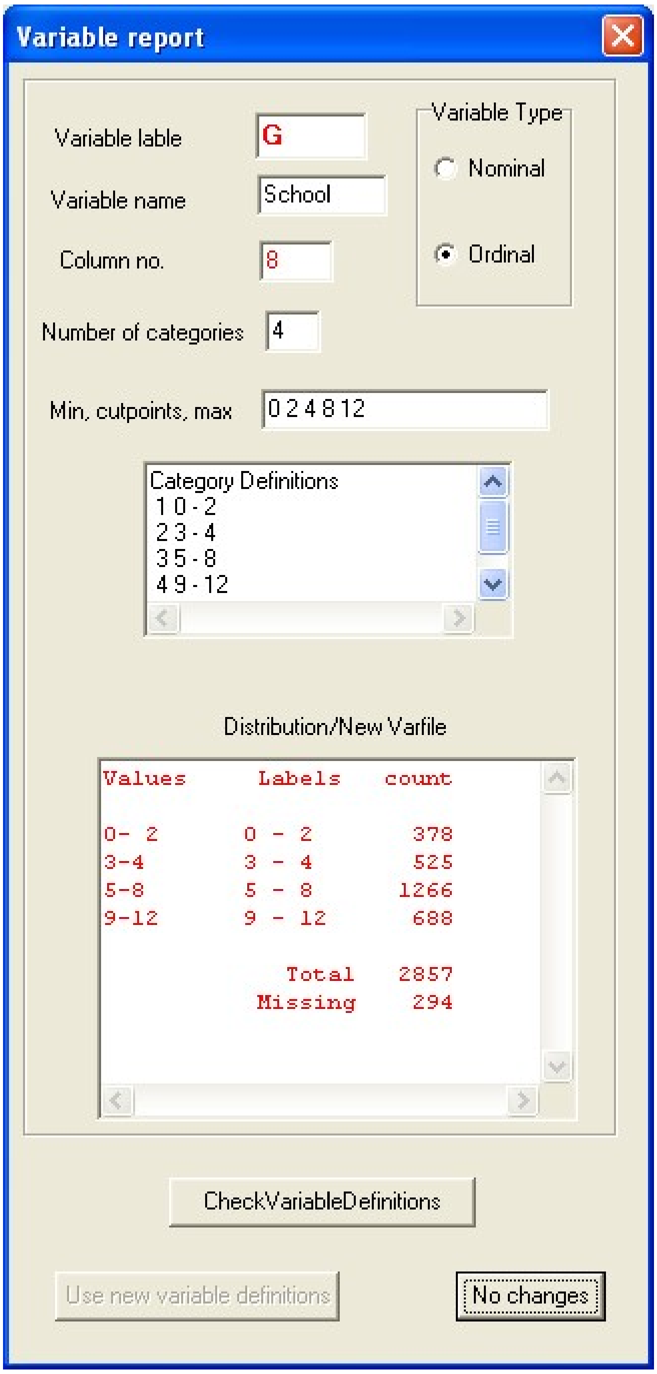
You can, of course, take your project with you if you want to work on a different computer, but the coordinates of the graph may not always fit the graph display. DIGRAM will shrink the graph if it too large, but will not do anything if it is too small. Rather that changing the position of all edges to fit the graph window on the new computer, you may rescale that graph, by a factor. Tod do this, select "Rescale graph" from the File menu and select a rescale factor.



If the factor rescales the graph beyond the size of the current window, DIGRAM reduces the factor to the largest factor leaving the complete graph inside the window.

Getting information and changing variables

Right-click on a node to get information on how the variable is defined and on the distribution of the variables as shown below



Variable report

Variable lable: G

Variable name: School

Column no.: 8

Number of categories: 4

Min, cutpoints, max: 0 2 4 8 12

Variable Type: ○ Nominal ● Ordinal

```
Category Definitions
 1 0 - 2
 2 3 - 4
 3 5 - 8
 4 9 - 12
```

Distribution/New Varfile

```
Values      Labels    count

0- 2       0 - 2       378
3-4        3 - 4       525
5-8        5 - 8      1266
9-12       9 - 12      688

             Total    2857
           Missing     294
```

You can change the definitions written in black, if you are dissatisfied with the way the variable has been defined and you can change the variable type. You cannot change the label and you cannot change the position of the variable in the original dataset. Click on “Check variable Definitions” when you are ready to see whether the new definitions are acceptable. If so, the “Use new variable definitions” button will be enabled you. Click on it to implement the changes, following which new VAR and CAT files will be created. If categories have been altered, DIGRAM will also generate new SYS and TAB files. This may take a little while.

Modifying the graph

Underlying the graph there is an edge status matrix defining the status of the connection between all pairs of variables. We distinguish between four different status values

Edge status	
0	No edge. Variables are assumed to be conditionally independent. Edges are prevented from being added to the model during the semi-automatic model search procedures
1	No edge. Variables are assumed to be conditionally independent.
2	Edge included. Variables are assumed to be conditionally dependent.
3	Variables are assumed to be conditionally independent. The edge is fixed in the model in the sense that it will not be removed during semi-automatic model search.

Click on the “Add edge” button or invoke the ADD command, if you want to add an edge to the graph with edge status equal to 2. Having clicked on the button, click first on one of the variables (the node colour turns green) and then on the other variable. You can continue to add edges without clicking on the button until you make an error clicking outside the nodes.

Click on the “Delete edge” button or invoke the DELETE command if you want to delete an edge. (edge status will be equal to 1) Having clicked on the button, click first on one of the variables (the node colour turns red) and then on the other variables. Again, you can continue to add edges without clicking on the button until you make an error clicking outside the nodes.

The complete list of commands that you may use to change the graph is shown in the following table:

Command	Parameters	Effect
FIX	Variable pairs	Edge status = 3
ADD	Variable pairs	Edge status = 2
DELETE	Variable pairs	Edge status = 1
PREVENT	Variable pairs	Edge status = 0
NEW	New status	The new status applies to all edges

Saving the graph

Click on the "SAVE" button or invoke the **SAVE** command without additional parameters. The default file on which DIGRAM assumes that you will save the model is the GRF file of the project. You may, of course, save the graph on a file with a different name. The current project model is saved on the project.GRF file. If the GRF file already exists, you will be prompted for a response to a question of whether you really want to overwrite the file.

Not all information on the graph is saved on the GRF file. The graph file will include information on labels, names and locations of the nodes, on the recursive structure and on the edge status matrix. Node colours and the node sizes are not save. If you want to be able to return to the same set of node colours you should save NODECOLOUR commands on a CMD file. Information on whether or not edges should be solid or dotted are also not be saved in the current version of DIGRAM. It will eventually be implemented. Look at the list of DIGRAM news (**SHOW N**) to see whether it has happened.

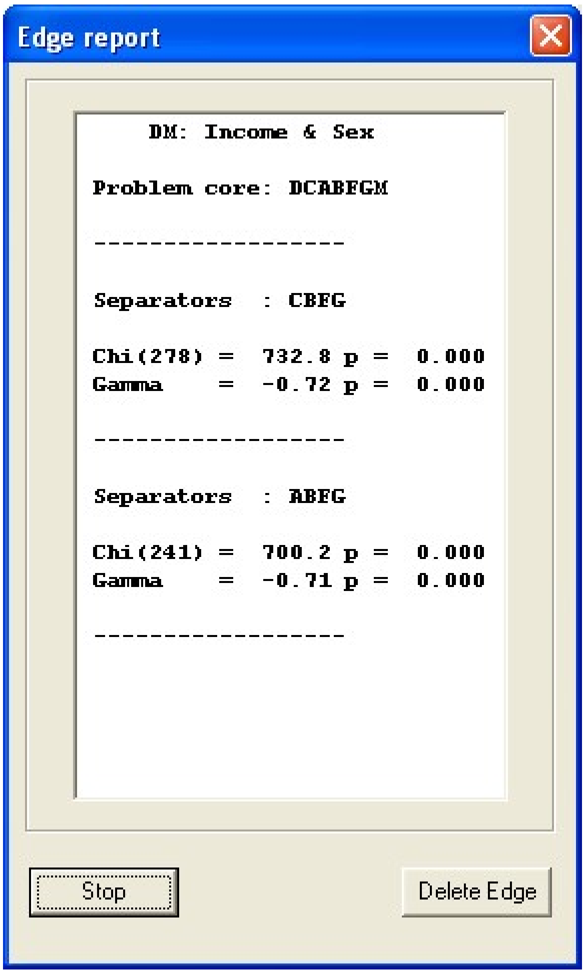
Reading graphs

Click on the "READ" button and select one of the GRF files to return to the model which is saved on the file. No command is available for this purpose at the moment.

Including information from the analysis of data in the graph

You have already seen how you can obtain information the distribution of a variable by right clicking the variable. If you right click on an edge DIGRAM will test all separation hypotheses of conditional independence of the two variables connected by the edge.

Right-clicking on the arrow from Sex (M) to Income (D) results in the following



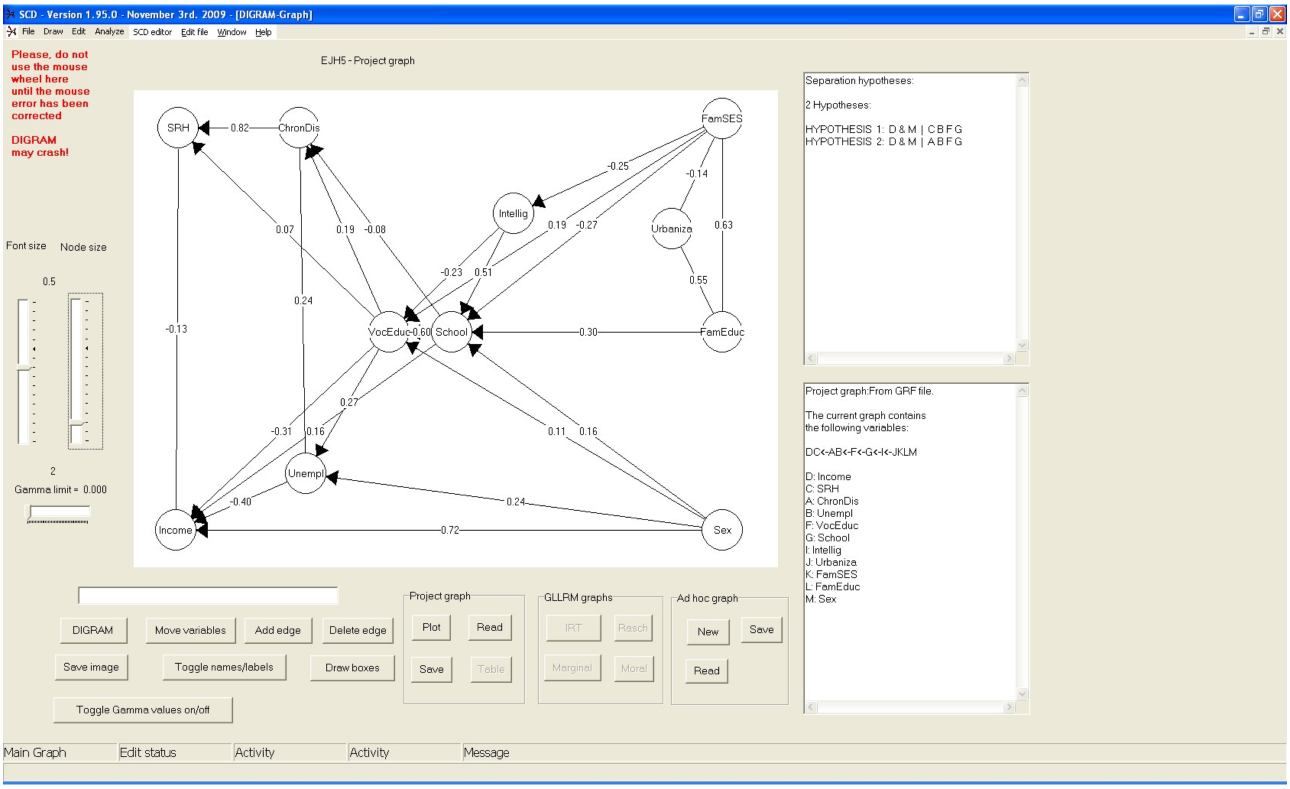
```
    DM: Income & Sex

Problem core: DCABFGM

------------------

Separators  : CBFG

Chi(278) =   732.8 p =   0.000
Gamma    =   -0.72 p =   0.000

------------------

Separators  : ABFG

Chi(241) =   700.2 p =   0.000
Gamma    =   -0.71 p =   0.000

------------------
```

The information included above tells you both that the there is a strong effect of sex on income (γ coefficients equal to 0.71 and 0.72 is indicative of *very* strong association) and that the effect is highly significant. If you want to include the information on the strength of the associations in the graphical display of the graph you must first return to the DIGRAM window, invoke the **GAMMA** command without parameters and then return to the Graph window. The γ coefficients are shown below. Note that they are equal to zero when the model assumes that variables are conditionally independent (no edges in the graph).

```
Partial Gamma coefficients calculated under the current model

             D      C      A      B      F      G      I      J      K      L

D:  Income   .    -0.129   0    -0.400 -0.314  0.164   0      0      0      0
C:     SRH -0.129   .     0.816   0     0.071   0      0      0      0      0
A:ChronDis   0     0.816   .     0.242  0.191 -0.081   0      0      0      0
B:  Unempl -0.400   0     0.242   .     0.273   0      0      0      0      0
F: VocEduc -0.314  0.071  0.191  0.273   .    -0.603 -0.231   0     0.192   0
G:  School  0.164   0    -0.081   0    -0.603   .     0.508   0    -0.274 -0.296
I:Intellig   0      0      0      0    -0.231  0.508   .      0    -0.253   0
J:Urbaniza   0      0      0      0      0      0      0      .    -0.142  0.549
K:  FamSES   0      0      0      0     0.192 -0.274 -0.253 -0.142   .     0.628
L: FamEduc   0      0      0      0      0    -0.296   0     0.549  0.628   .
M:     Sex -0.716   0      0     0.239  0.112  0.158   0      0      0      0

             M

D:  Income -0.716
C:     SRH   0
A:ChronDis   0
B:  Unempl  0.239
F: VocEduc  0.112
G:  School  0.158
I:Intellig   0
J:Urbaniza   0
K:  FamSES   0
L: FamEduc   0
M:     Sex   .
```

The γ coefficients are estimated by the tests for separation hypotheses. If more than one hypothesis exists the γ coefficients reorted above are weighted means of all the γ coefficients.

Having calculated the γ coefficients you must return to the GRAPH window, where you will see that the “Toggle gamma values on/off” has been enabled. Click on this button to add γ coefficients to the edges of the graph as shown below.



Separation hypotheses:

2 Hypotheses:

HYPOTHESIS 1: D & M | C B F G
HYPOTHESIS 2: D & M | A B F G

Project graph:From GRF file.

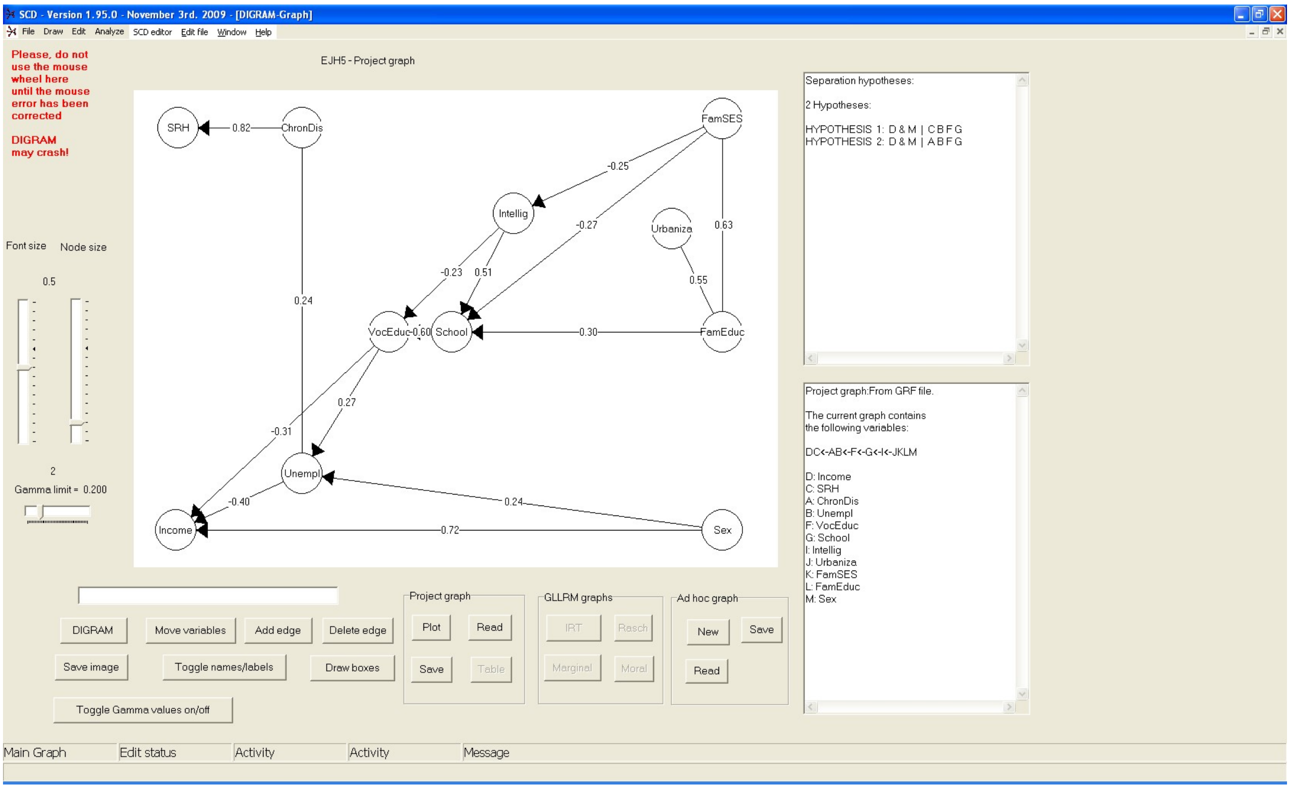
The current graph contains
the following variables:

DC<-AB<-F<-G<-I<-JKLM

D: Income
C: SRH
A: ChronDis
B: Unempl
F: VocEduc
G: School
I: Intellig
J: Urbaniza
K: FamSES
L: FamEduc
M: Sex

In the display above, note the “Gamma limit” ruler that has appeared to the left of the graph. Slide the ruler to the right to change the graph view to remove edges with γ coefficients below the gamma limit. In the display below, DIGRAM only shows the edges where |γ| ≥ 0.20.

Note that the weak edges in the model still remain in the project model. They have just been excluded from the visual graph in order to make it easier to see the main structure of the model. If you for some reason should want to consider models restricted by the γ coefficients you have to invoke the **SAVE G command** (remember to include the **G** parameter) following which DIGRAM will create a series of GRF files including edges limited by 0.1, 0.2, 0.3, 0.4 and 0.5.



Please, do not use the mouse wheel here until the mouse error has been corrected

DIGRAM may crash!

```
Separation hypotheses:

2 Hypotheses:

HYPOTHESIS 1: D & M | C B F G
HYPOTHESIS 2: D & M | A B F G
```

```
Project graph:From GRF file.

The current graph contains
the following variables:

DC<-AB<-F<-G<-I<-JKLM

D: Income
C: SRH
A: ChronDis
B: Unempl
F: VocEduc
G: School
I: Intellig
J: Urbaniza
K: FamSES
L: FamEduc
M: Sex
```

Displaying subgraphs

Use the **PLOT** command if you want to see subgraphs from containing subsets of the complete set of variables.

PLOT variable	plots all variables that are directly connected to the variables
PLOT variables	plots the subgraph with the nodes corresponding to the variable list

Click on the "PLOT" button or invoke a **PLOT *** command to return to the complete graph.

Analysis of Markov graphs

Analysis of data by graphical models depends on graph theoretical analysis of the Markov graphs. This kind of analysis is invoked automatically during the statistical analysis, but you can also do it yourself in both GRAPH and DIGRAM mode using the following commands. Output appears in the small output field at the top right side of the GRAPH window. Select "Copy output to DIGRAM" from the File menu to transfer the output to DIGRAM's larger output filed..

SEPARATE variable pairs
REDUCE variable pairs
GMP variable pairs
PATH variable pairs
CAUSALPATHS variable pairs
MORALIZE <level>
RELEVANCE two pairs of variables

Most of these commands are described in the guided tour through DIGRAMS facilities for analysis by graphical models. The only command which only functions in GRAPH mode is MORALIZE.

MORALIZE level creates and displays the moralized truncated graph containing all variables in the specified recursive block and all the variables in prior block according to the recursive structure.

Printing graphs

Select "Print Graph" on the File menu if you want to print the graph or save the graph image on a bitmap file using the "Save Image" button and insert the file as a picture in a Word document.

Different graph types

In addition to the Markov graph of the graphical project model DIGRAM also constructs Markov graphs for graphical loglinear Rasch models and let you create your own ad hoc graphs if you need such graphs. All the facilities for working with graphs also apply to these graphs.