On the drawing of certain diagrams useful in latent trait modelling

Michael E Dewey

July, 2009

1 Introduction

1.1 Purpose

This document describes a routine written in R (R Development Core Team, 2008) for displaying response patterns in a way useful in latent trait modelling.

1.2 Assumptions

I assume that you know about item response models. I also assume that you have installed the `dot` program (see page 3 for details).

1.3 Background

Rosenbaum (1987) considers a set of binary items (here coded 0 or 1) which form a latent scale of a certain class where item characteristic surfaces do not cross. This class includes Guttman, Rasch and the Mokken double monotone scales. He shows that the frequency of occurrence of the item response patterns will form a function decreasing in transposition. In the representation used here a function of the response patterns is decreasing in transposition if rearranging the 0s and 1s so that the 1s are further to the left reduces the value of the function. The sequences of patterns involved can be formed one from another (having the same number of 1s) by transposing a 1 with the 0 immediately to its left.

A convenient way of checking this assumption is by drawing the directed graph in which the nodes are the patterns and an edge exists between patterns connected by the transposition operation. This also provides a useful visual display of the response patterns

1.4 An example

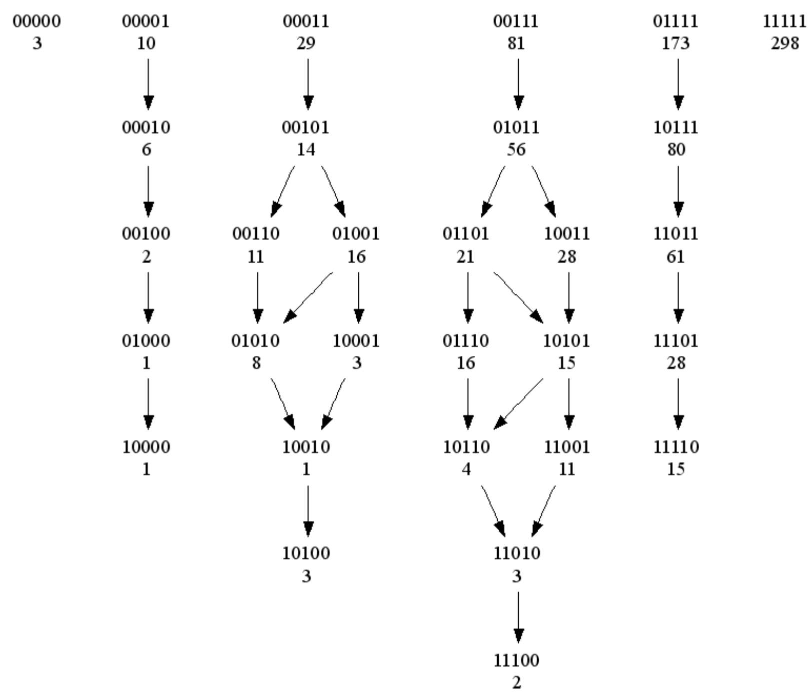

Figure 1: Patterns in the Lsat dataset

An example is shown in Figure 1. This uses the dataset `Lsat` contained in the `ltm` package. To display the patterns properly it is necessary to rearrange the items in order of increasing frequency of 1s and this has been done for Figure 1. The commands necessary are shown in Example 1

```
require(ltm)
draw.latent(LSAT, rootname = "lsat")
```

Example 1: Commands to generate Figure 1

In Figure 1 only those patterns which occur in the dataset are displayed and this is the default behaviour. The only two possible patterns which did not in fact occur are 01100 and 11000. We can see that most of the patterns do fit the desired behaviour with some small exceptions. For instance 00101 should be more prevalent than 01001 which is not true (14 versus 16) but the difference from expected is small.

Note that nothing is asserted about the frequencies of patterns with different numbers of 1s for instance 00110 and 10101. Similarly nothing is asserted about patterns not connected in the diagram for instance 00110 and 10001.

We can see that $3+10+29+81+173+298 = 594$ of the 1000 patterns form a Guttman scale.

2 Design decisions

The basic design decision underlying the function is to let the bulk of the detailed graph drawing be undertaken by the specialised graph drawing program `dot` (available from `http://www.graphviz.org/`). Users who wish to undertake fine tuning of their graph can do so by editing the output file of `dot` commands.

3 Getting a useful display

By default all patterns which occur are drawn. If there are a large number of items this can lead to an unwieldy display. There are two issues here

1. Some patterns do not lead directly from or to any another pattern (because the nodes which link them have not been printed as they do not occur in the dataset).
2. The diagram becomes very wide

The first of these can be addressed by forcing all patterns to appear even if they occur with frequency zero. The second point can be addressed by selectively printing only those patterns with a certain number of items positive. By doing this successively the whole diagram can be created spead over several pages.

Figure 2 shows an example of a scale with ten items. This is unpublished data on the ownership of various assets. In this case only those patterns with five positive responses are drawn. Example 2 shows the commands which rely on the dataset being in data frame `dat`.

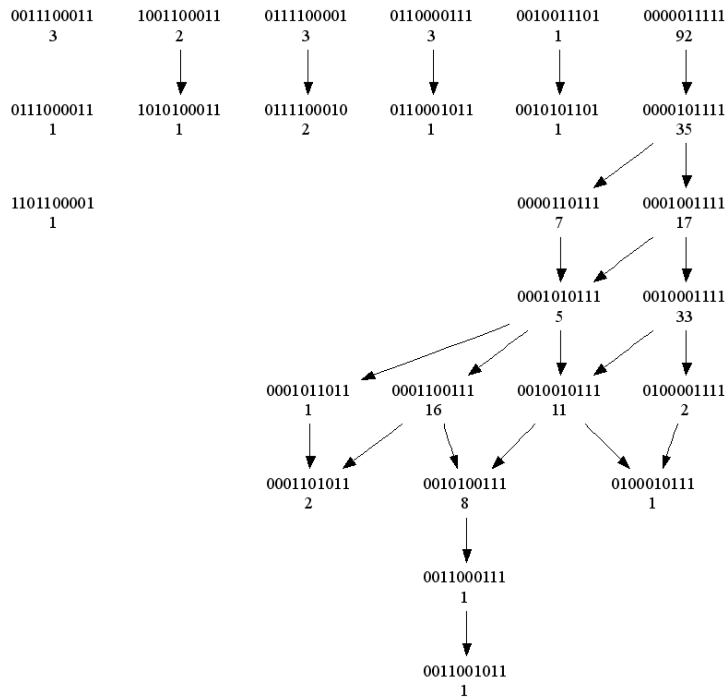

Figure 2: Ownership of various assets: (from left to right) tv, business, bed, valuables, radio, small animals, large animals, crop, land, house.

```
draw.latent(dat, rootname = "asset", which.npos = 5)
```

Example 2: Commands to generate Figure 2

We can see that only 26 of the possible 252 patterns actually occur and so the display is much simpler than it would be if we attempted to draw all the patterns. Bear in mind that the patterns which appear to have no ancestor in the diagram in fact do (except for 0000011111) it just occurred with frequency zero. We can see a number of problems in the diagram, for instance 0010001111 occurs with frequency 35 while its ancestor has frequency 17. Based on theoretical considerations, previous work, and fitting a model with two latent traits using `ltm` a final model was achieved with separate scales for rural and urban assets, but as expected from the diagram a single trait using the Rasch model was not possible for the whole set of items.

4 Using dot to draw the final diagram

If you are happy with the defaults you can just call the `plot` method on the object returned by `draw.latent`. If you want finer control then assuming you output the `dot` commands to `lsat.dt` as shown in Example 1 and you want Portable Network Graphic output then after possibly editing the commands you simply get a command prompt and type the commands shown in Example 3.

```
dot -Tpng lsat.dt -o lsat.png
```

Example 3: Commands to process output from Example 1

This gives you a file in the desired format in `lsat.png`. Many other output formats are possible, see the `dot` documentation for details. Note that I use the extension `.dt` for `dot` commands as the more natural `.dot` is used by Windows for another purpose. Unix users may prefer `.dot`.

5 Version history

May 2006 — first version uploaded.

February 2009 — changed to not depend on `ltm` and `plot` method added to call `dot`.

6 To do

Label for whole diagram and for sub–diagrams.

Write generics for objects from `ltm`.

7 Acknowledgements

I would not have thought of using R to generate these diagrams without the example of John Fox's `sem` with its `path.diagram` nor would I have used R for item response theory models without Dimitris Rizopoulos' `ltm`. I intend to add generics for the objects returned in `ltm` in a later version of the package.

References

R Development Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. URL `http://www.R-project.org`. ISBN 3-900051-07-0.

P R Rosenbaum. Probability inqualities for latent scales. *British Journal of Mathematical and Statistical Psychology*, 40:157–168, 1987.