Creating graphical models in LaTeX

Tom Lodewyckx

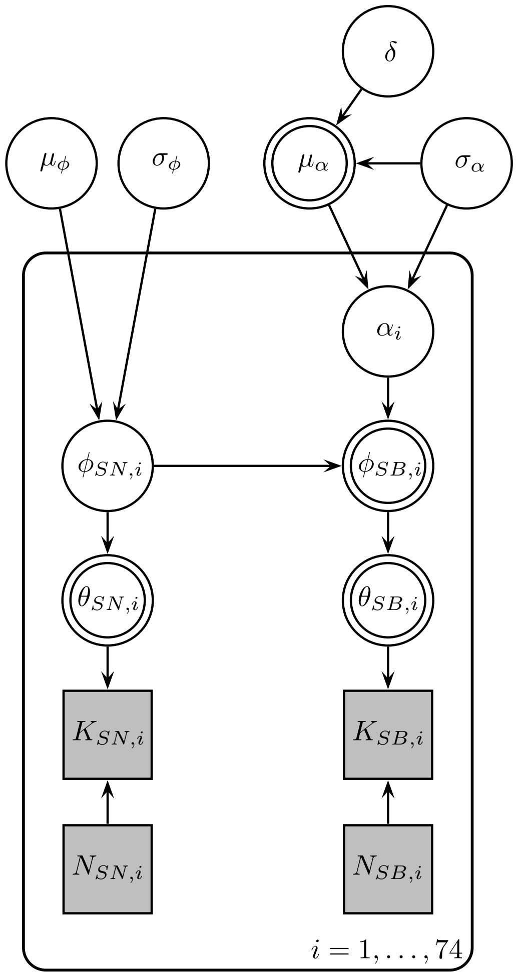
University of Leuven

Department of psychology, Tiensestraat 102, 3000 Leuven BE

☎ (+32) 16 326052

✉ tom.lodewyckx@psy.kuleuven.be

sites.google.com/site/tomlodewyckx



Contents

1 About graphical modeling

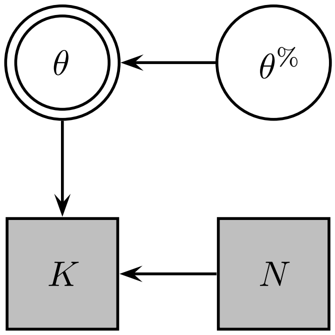
Graphical modeling was introduced to facilitate the interpretation of Bayesian models by means of a graphical representation of the model. Dependencies between and hierarchical structures of the parameters and observations are easily interpreted by inspection of the graphical model. In this tutorial, we will illustrate that it is not too difficult to construct a graphical model with LaTeX code yourself. This is a tutorial on how to make graphical models, not on graphical modeling itself. We will explain the basics of graphical modeling in this section. If you would like more information, there are some other sources available (Gilks, Thomas & Spiegelhalter, 1994; Lee & Wagenmakers, 2009; Spiegelhalter, 1998).

The structure of a graphical model is based on a bottom-up strategy. At the bottom of the graph, the observed variables are represented. Then latent parameters are "built" upon these variables. The physical distance of latent parameters to the observed variables in the graph symbolizes the level of abstraction of the inference. Quantities on the same horizontal level should have (more or less) the same level of abstraction, for instance: observed variables (bottom), individual level parameters, group level parameters and differences between group parameters (top).

The graph consists of nodes that can represent observed variables or latent parameters. In a statistical model, all nodes are interrelated through deterministic or stochastic dependencies. In a particular dependency relation, the child node depends on its parent node(s), visualized in the graph with an arrow pointing from parent to child. A node can be child and parent simultaneously, when it is involved in several dependency relations. The visual properties of a node indicate some of its characteristics:

- **Observed versus latent.** Observed variables (or known quantities that are typical for the design of the study, such as the number of trials) are represented by nodes shaded in gray, whereas latent variables are represented by non shaded nodes.
- **Continuous versus discrete.** Continuous quantities are given circular nodes, whereas discrete or categorical quantities are given square nodes.
- **Stochastic versus deterministic.** Nodes that are fully determined by the values of their parent nodes have a double border, whereas other nodes have only one border.

Example: Assume we roll a die N times and we find K successes, where a succes is defined as an outcome of six. A simple and realistic model for this type of data is presented in Figure 1. θ is the proportion of successes in the Binomial distribution and is multiplied by 100 to obtain $\theta^{\%}$, the percentage of successes. Both K and N are observed (shaded) and discrete (square) quantities, whereas θ and $\theta^{\%}$ are unknown (not shaded) and continuous (circular) quantities. The double border of θ indicates that its value is fully determined by the value of its parent $\theta^{\%}$. The only unknown quantity to estimate is $\theta^{\%}$, which is given a uniform prior distribution on the range of 0 to 100. Notice that θ is simultaneously a child of $\theta^{\%}$ and a parent of K. It is convenient to assign Greek symbols to parameter nodes and Latin symbols to data nodes.



$$K \sim \text{Binomial}\big(\theta, N\big)$$

$$\theta = \theta^{\%}/100$$

$$\theta^{\%} \sim \text{Uniform}(0, 100)$$

Figure 1: Graphical model for the die example.

In Section 2, we will show how to use LaTeX code to create a graphical model. This will all be applied in Section 3, where a graphical model is created for a hierarchical Bayesian model for data from a priming experiment.

2 Creating a graphical model in LaTeX

Before you start the construction of the graph in LaTeX, it is evident that you know how your graphical model will look like. It helps to compare several pencil-and-paper drafts. Some important criteria are the following.

- **Clarity.** The model structure should be clear at inspection of the graph. Avoid arrows to touch other nodes than the child and parent node that are being connected.
- **Symmetry.** If possible, replicate substructures that are assumed to have the same set of dependencies (e.g., when data from two groups are analyzed similarly).
- **Horizontal levels.** All nodes on one level should have (more or less) the same degree of abstraction.

Once you have chosen the optimal design, visit my homepage[1] and download the ZIP archive `TutorialGMLTX.zip` from the downloads section. Unpack its files into a temporary folder and copy the LaTeX style files `com.braju.graphicalmodels.sty` and `com.braju.pstricks.sty`[2] and the template `GraphTemplate.tex` to a preferred working

[1] `http://sites.google.com/site/tomlodewyckx`

[2] These style files were originally developed by Henrik Bengtsson (`http://www1.maths.lth.se/matstat/staff/hb/`) and adapted by Michael Lee (`http://www.socsci.uci.edu/` $\sim$ `mdlee/`)

directory. Now open the template with your LaTeX editor[3] and set the output profile to [LaTeX⇒ `PS`] or [LaTeX⇒ `PS` ⇒ `PDF`]. Here you find five sections in the document that correspond to different elements of the graphical model:

1. The **grid** is the reference frame, consisting of an orthogonal coordinate system.
2. The **nodes** are representing data and parameters in the model.
3. The **arrows** indicate dependencies between the nodes.
4. The **plates** take hierarchical structures into account.
5. The **model equations** define the full statistical model with the exact transformations and distributional assumptions.

In the subsections 2.1 to 2.5, we will discuss each element in detail and illustrate how LaTeX code is written in order to create the desired graphical model.

2.1 Grid

The grid is the orthogonal coordinate system that functions as the reference frame for all elements in the graph. Distances in the grid are expressed in units, whereas one unit is set to 14 millimeters[4]. The grid is defined with two lines of LaTeX code.

```
\begin{pspicture}(minx,miny)(maxx,maxy)
\showgrid
```

- **Grid display.** The `\showgrid` command is used to display the grid. This is useful as long as you are designing the graphical model: it helps you to choose the right coordinates easily. Deactivate it once the graph is finished by commenting out the command with the % symbol.
- **Minimum and maximum grid values.** The sets of coordinates `(minx,miny)` and `(maxx,maxy)` specify respectively the minimum and maximum grid values on the x-axis and y-axis. It is convenient to set `(minx,miny)` equal to (0,0) such that they can be interpreted as the coordinates of the origin of the coordinate system. Subsequently, `(maxx,maxy)` are to be interpreted as the width and heighth of the grid. Once you have added nodes and plates to the graph (see Sections 2.2 and 2.3), check again whether your grid is still large enough: elements far outside the grid might fully or partly disappear from the graph.

[3]TeXnicCenter is popular freeware and is available at `http://www.texniccenter.org`

[4]The size of this basic unit is set in the `\ppset` command in the beginning of the template with `unit=14mm`

2.2 Nodes

All observed variables and stochastic parameters are visually represented with nodes. Several graphical properties of the nodes imply certain characteristics of the corresponding quantities. A node is specified with two lines of code: the first line defines the basic visual properties of the node, the second line defines the label that is associated to the node. Options for the node (`NodeOptions`) and for the label (`LabelOptions`) can be specified within the square brackets, seperated by commas, e.g., `[option1=a,option2=b,...]`.

```
\rput(locx,locy){\GM@node[NodeOptions]{name}}
                 \GM@label[LabelOptions]{name}{label}
```

General

- **Location.** The set of coordinates `(locx,locy)` defines the horizontal and vertical position of the node within the grid.
- **Identification.** `name` is substituted by a unique node name. When drawing arrows between nodes, these names are necessary to identify the right nodes. For clarity, one chooses the "textified" mathematical symbol, e.g., $\theta^{\%}$ is given the name `ThetaPerc`. Spaces, special symbols and LaTeX commands are not allowed in the node name.
- **Label.** `label` is substituted by the statistical symbol of the corresponding quantity (activate the math environment using two $ signs if needed). This label will appear inside the node.

Node options

- **Node size.** The `nodeSize` option defines the diameter of the node and is generally set to 11 millimeters (`nodeSize=11mm`).
- **Shading.** The `observed` option is set to true (`observed=true`) when the quantity is observed, such that this node is shaded. The `observedColor` option defines the shading color and is generally set to light gray (`observedColor=lightgray`), which is a bit lighter than the default color gray[5]. For unobserved nodes, these options are not mentioned within the node options environment.
- **Geometric form.** For discrete or categorical quantities, the `query` option is set to true (`query=true`) such that they are displayed as a square node. For continuous quantities, this option is not specified and they are displayed as circular nodes.

[5] Colors from the LaTeX palette are listed at `http://en.wikibooks.org/wiki/LaTeX/Colors`

Label options

- **Correction of label position.** It often happens that the label is not presented exactly in the middle of the node, especially when that label contains indices. A subtle correction of the position of the label in millimeters can be made with the `offset` option (`offset=?mm`). The corrections are parallel to the x-axis, whereas positive values move the label to the right and negative values move it to the left. If needed, the direction of the movement can be changed using the `angle` option. The rotation of the correction axis is expressed in degrees in counterclockwise direction relative to the x-axis (e.g., `angle`=45 moves the label to the upper right corner for positive correction values or to the lower left corner for negative correction values).

Fully determined child nodes

A somewhat more complex issue is how to create a double border for child nodes that are fully determined by the value(s) of their parent node(s). Two nodes are created at the same location: first the outer border as a node with a normal size of 11mm and then the inner border as a node with a smaller size of 9 mm. In the basic structure of the code below, the label is linked to the inner node because they have the same value for `name` (it could as well be linked to the outer node without any problem).

```
\rput(locx,locy){\GM@node[nodeSize=11mm,...]{nameouter}}
\rput(locx,locy){\GM@node[nodeSize=9mm,...]{nameinner}}
                  \GM@label[...]{nameinner}{label}
```

2.3 Arrows

One of the goals of graphical modeling is to reveal dependencies in the model by looking at the graph. An arrow from parent node to child node indicates that the value of the child node depends on that parent node. An arrow is constructed with one line of code.

```
\ncline[arrows=->]{nameparent}{namechild}
```

- **Identification.** `nameparent` and `namechild` are substituted by the unique names of the parent and child node, as specified when these nodes were created.

- **Direction.** The direction of the arrow is defined in the `arrows` option. To let the arrow begin in the parent node and end in the child node (as is convenient in graphical modeling) it is set equal to an arrow to the right (`arrows=->`). However, you can always change the direction: from child to parent (`arrows=<-`) or bidirectional (`arrows=<->`).

- **Fully determined node.** In case that either parent node or child node is a deterministic node, be sure to choose the name of the outer node (to avoid that the arrow will be drawn partly inside the node).

- **Arrow thickness.** The arrowscale option within the \ppset command (in the beginning of the template) specifies the thickness of the arrows. The default value is set to 1.5 units (arrowscale=1.5) but can be easily changed when preferred.

2.4 Plates

Plates can visualize hierarchical structures in the data, where various independent, exchangeable quantities are assumed to be drawn from the same distribution. The exchangeable nodes can be either data or parameters. A plate is defined with one line of LATEX code:

```
\rput(orix,oriy){\GM@plate[options]{sizex}{sizey}{label}}
```

- **Origin.** The set of coordinates (orix,oriy) defines the location of the origin (the lower left corner) of the plate.
- **Size.** The values of width and heighth define respectively the width and heighth of the plate, implying that the upper right corner of the plate has coordinates (orix+width,oriy+heighth)[6].
- **Label.** label is substituted by a loop expression, (e.g., $i=1,\ldots,20$ in case there are 20 exchangeable nodes). Each node label within the plate should contain the index that is specified in this label.
- **Position label.** The exact location inside the plate is defined with the plateLabelPos option. In graphical modeling, it is convenient to place the label in the bottom right corner (plateLabelPos=br). The alternatives are the bottom left (bl), top right (tr) and top left (tl) corner.

2.5 Model equations

The graphical model is not a model on itself but rather complementary to the exact model equations. Therefore it is useful to add these equations to the graph. Each equation is added with a single line of code.

```
\rput(locx,locy){\pnode{name}}\GM@label[options]{name}{label}
```

- **Location.** The set of coordinates (locx,locy) specify the location of a specific equation line. For a nice layout of all equations, use the same value of locx for all equations and vary the value of locy such that the equations have the same vertical spacing between each other.

[6]This specification is different from the coordinate sets (minx,miny) and (maxx,maxy) for the grid, where (maxx,maxy) can only be interpreted as width and heigth of the grid if (minx,miny) = (0,0)

- **Identification.** We explained earlier that it is important to identify the right nodes such that arrows can be drawn appropriatly between these nodes. However, equation identification is not that important. For each equation, `name` can be substituted with the same word (e.g., *equation*).
- **Label.** `label` is substituted by the actual equation that will be written in the graph. Use the math environment for mathematical symbols.
- **Correction of label position.** For subtle corrections of the location of the equation, the `offset` and `angle` options can be used in exactly the same way as explained in the subsection on creating nodes.

Commenting out the `\showgrid` command makes the grid disappear, so you have your final graph. You have it available in postscript (PS) format, which can be transformed into an encapsulated postscript format (EPS)[7]. EPS figures can be easily added to LaTeXdocuments.

3 Application

In this section, we introduce a subliminal priming study by Zeelenberg, Wagenmakers & Raaijmakers (2002) and discuss how these data can be analysed with a Bayesian hierarchical model. The corresponding graphical model is constructed step by step with parts of LaTeX code. If you are not interested in the theoretical background and the meaning of the model and its parameters, just start with the subsection "Construction of the graphical model".

3.1 Experimental design and data

Zeelenberg, Wagenmakers & Raaijmakers (2002) set up several studies to test whether previous exposure to a stimulus would facilitate visual discriminability of that stimulus, relative to a foil stimulus. To distinguish this effect from a pure bias effect as a result of the prior study of the stimulus, the foil stimulus was also studied on forehand. We focus on the third experiment that is reported in their paper.

74 subjects participated and 42 pairs of similar looking pictures were used as the visual stimulus material (see an example in Figure 2). The critical within-subject manipulation was realized in the study block. Participants were able to study half of the pairs and these pairs belonged to the Study Both (SB) condition, whereas the other non-exposed pairs belonged to the Study Neither (SN) condition. Then the test block followed, consisting of 42 trials with each trial corresponding to one of the 42 picture pairs. A trial started with a short presentation of one picture of the pair for 40 ms, the "target". The other picture from that pair, the "foil", was not presented. Then a two-alternative forced-choice task followed: target and foil were presented together and the participant had to identify the

[7] This can be done using GSview, available for free at `http://pages.cs.wisc.edu/~ ghost/gsview/`

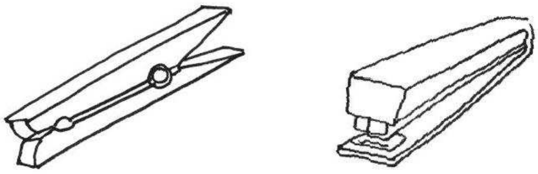


Figure 2: Example of a picture pair from the visual stimulus material.

target. For each subject, the test block resulted in a count of correct identifications for the SB condition (K_{SB}) and the SN condition (K_{SN}) out of respectively N_{SB} and N_{SN} trials. There were no missing data, so $N_{SB} = N_{SN} = 21$. In Figure 3, the proportions of successes in the Study Both condition are plotted against the proportions of successes in the Study Neither condition for all participants.

3.2 Hierarchical Bayesian model

The authors performed a paired t-test and found a significant condition effect: the percentage of correct trials was higher in the SB condition (74.7%) than in the SN condition (71.5%), $t(73) = 2.19, p < .05$. This result was taken to support the hypothesis that prior study leads to an improved visual discriminability. A new Bayesian hierarchical modeling approach for this data was proposed (Lodewyckx, Lee & Wagenmakers, 2009). We will discuss this model shortly and then build up the graphical model step by step. If you want more background, you can find it in our manuscript.

The graphical model of interest is presented in Figure 4. The counts of correct identifications $K_{SB,i}$ and $K_{SN,i}$ for each participant i are assumed to be Binomial distributed. The Binomial proportion parameters $\theta_{SB,i}$ and $\theta_{SN,i}$ (ranging from 0 to 1) were probit transformed into $\phi_{SB,i}$ and $\phi_{SN,i}$ (ranging from -∞ to +∞)[8]. We are interested whether participants score better in the SB condition than in the SN condition. In statistical terms, we want to know whether $\phi_{SB,i} > \phi_{SN,i}$. Instead of modeling $\phi_{SB,i}$ as a free parameter, we fix it to $\phi_{SB,i} = \phi_{SN,i} + \alpha_i$ and model the difference parameter $\alpha_i = \phi_{SB,i} - \phi_{SN,i}$ as a free parameter. Now it is assumed that the parameters $\phi_{SN,i}$ and α_i are Gaussian distributed with hyperparameters μ_ϕ, σ^2_ϕ, μ_α and σ^2_α. The effect size for the parameter μ_α can be derived as $\delta = \mu_\alpha/\sigma_\alpha$. The key parameter is δ, which is expected to be positive in case of a subliminal learning effect without bias. The prior distributions are chosen to be uninformative. The prior for μ_ϕ is truncated to the positive value domain because negative values correspond to (unrealistic) performances under chance level.

[8]The probit transformation is the inverse cumulative Normal distribution function

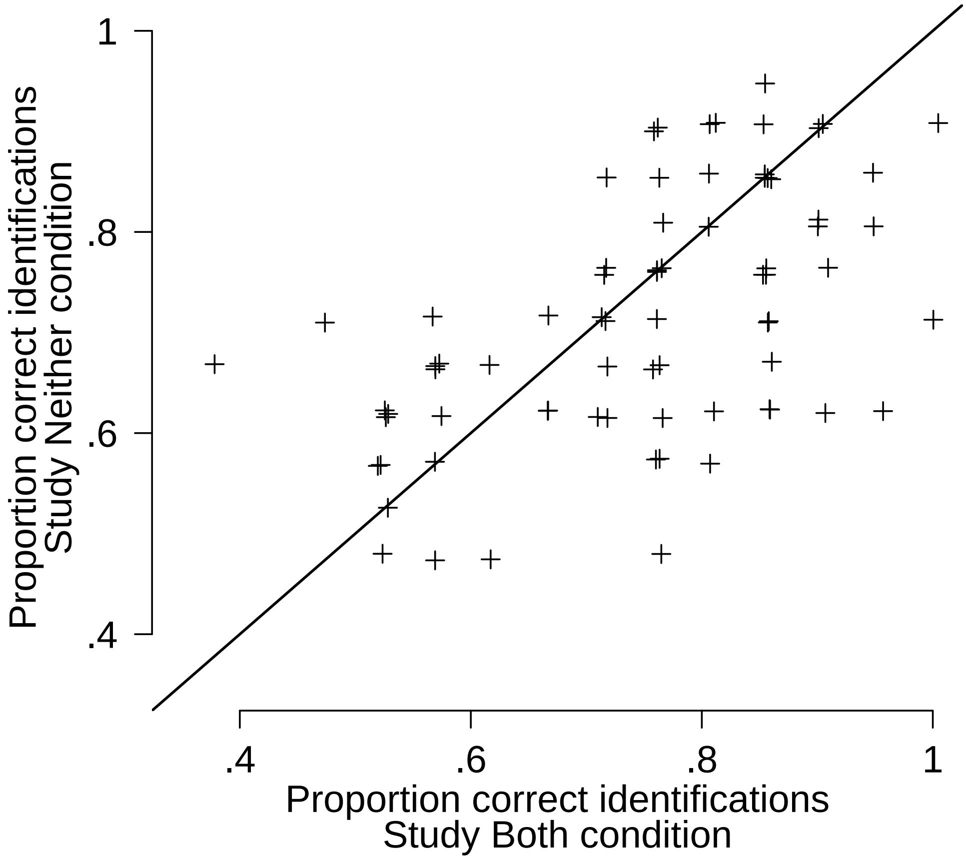
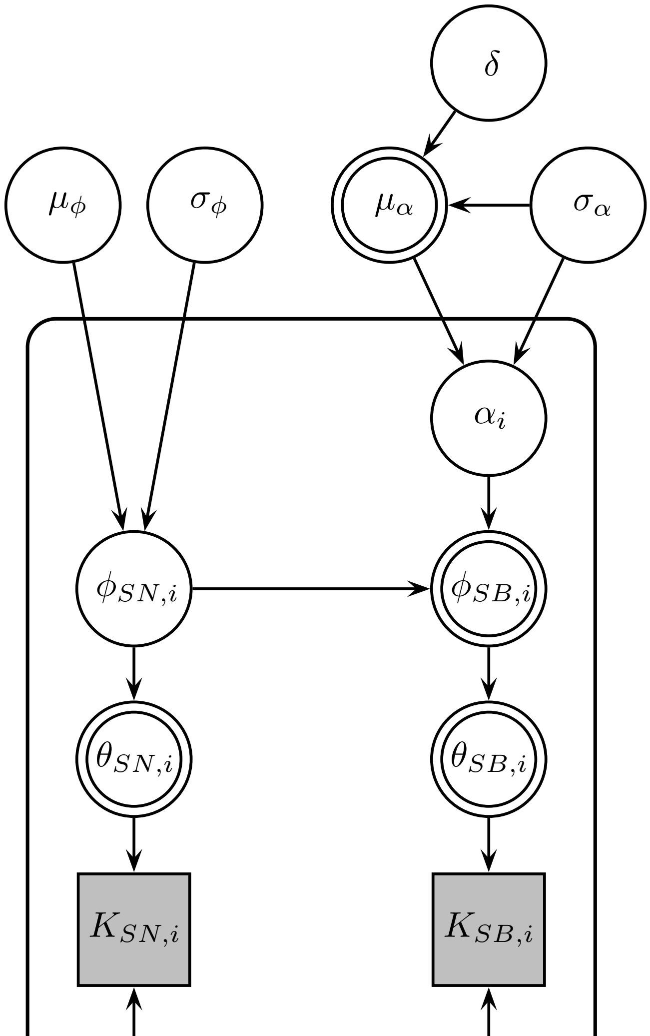


Figure 3: Proportions of correct identifications of the 74 subjects in the Study Both and Study Neither condition (Zeelenberg et al., 2002). A small amount of jitter is added to disentangle overlapping observations.



$$K_{SN,i} \sim \text{Binomial}\big(\theta_{SN,i}, N_{SN,i}\big)$$

$$\theta_{SN,i} = \Phi\big(\phi_{SN,i}\big)$$

$$\phi_{SN,i} \sim \text{Normal}\big(\mu_\phi, \sigma_\phi^2\big)$$

$$K_{SB,i} \sim \text{Binomial}\big(\theta_{SB,i}, N_{SB,i}\big)$$

$$\theta_{SB,i} = \Phi\big(\phi_{SB,i}\big)$$

$$\phi_{SB,i} = \phi_{SN,i} + \alpha_i$$

$$\alpha_i \sim \text{Normal}\big(\mu_\alpha, \sigma_\alpha^2\big)$$

$$\mu_\phi \sim \text{Normal}_{(0,+\infty)}\big(0, 1\big)$$

$$\sigma_\phi \sim \text{Uniform}\big(0, 10\big)$$

$$\mu_\alpha = \delta \times \sigma_\alpha$$

$$\sigma_\alpha \sim \text{Uniform}\big(0, 10\big)$$

$$\delta \sim \text{Normal}\big(0, 1\big)$$

Figure 4: Graphical model for the Bayesian hierarchical model for the subliminal learning data.

3.3 Construction of the graphical model

In this section, the graphical model in Figure 4 will be constructed. Step by step, lines of LaTeX code are discussed for the different elements of the graphical model and the effect of the code is illustrated visually in Figure 5. If you want to have a look at the code yourself, open the file GraphZeelenberg.tex that you find in the ZIP file TutorialGMLTX.zip[9].

0. Begin document

These lines of code remain unchanged, unless you would like to change the size of the basic unit or the thickness of the arrows (within the `\ppset` command). However, it is recommended to work with these default settings.

```
\documentclass{article}
\usepackage{pst-all}
\usepackage{com.braju.graphicalmodels}
\catcode`\@=11%
\pagestyle{empty}
\begin{document}
\TeXtoEPS
\psset{unit=14mm,arrowscale=1.5}
\SpecialCoor
```

1. Grid

As the graphical model contains many nodes and it has approximatly the same width and heighth, we choose a grid of 10×10 units (see Figure 5.A).

```
\begin{pspicture}(0,0)(10,10)
\showgrid
```

2. Nodes

There are several nodes which are fully determined by the values of their parents: θ_{SB}, θ_{SN}, ϕ_{SB} and μ_α. We start with defining their outer nodes in the first paragraph of code. In the second paragraph, the nodes for the hierarchical parameters at the top of the graph outside the plate are created. In the third and fourth paragraph, the nodes within the plate for respectively the SN condition and the SB condition are defined. Notice that for every node there is a correction of the label position with the `offset` option. The result is shown in Figure 5.B.

[9] `http://sites.google.com/site/tomlodewyckx`

```
\rput(4.0,4.6){\GM@node[nodeSize=11mm]{phiSB_outer}}
\rput(4.0,3.4){\GM@node[nodeSize=11mm]{thetaSB_outer}}
\rput(1.5,3.4){\GM@node[nodeSize=11mm]{thetaSN_outer}}
\rput(3.3,7.3){\GM@node[nodeSize=11mm]{mualpha_outer}}

\rput(1.0,7.3){\GM@node[nodeSize=11mm]{muphiSN}}
               \GM@label[offset=-1.5mm]{muphiSN}{$\mu_{\phi}$}
\rput(2.0,7.3){\GM@node[nodeSize=11mm]{sdphiSN}}
               \GM@label[offset=-1.5mm]{sdphiSN}{$\sigma_{\phi}$}
\rput(4.0,8.3){\GM@node[nodeSize=11mm]{delta}}
               \GM@label[offset=-0.5mm]{delta}{$\delta$}
\rput(3.3,7.3){\GM@node[nodeSize=9mm]{mualpha}}
               \GM@label[offset=-1.5mm]{mualpha}{$\mu_\alpha$}
\rput(4.7,7.3){\GM@node[nodeSize=11mm]{sdalpha}}
               \GM@label[offset=-1.5mm]{sdalpha}{$\sigma_\alpha$}

\rput(1.5,4.6){\GM@node[nodeSize=11mm]{phiSN}}
               \GM@label[offset=-3.8mm]{phiSN}{$\phi_{SN,i}$}
\rput(1.5,3.4){\GM@node[nodeSize=9mm]{thetaSN}}
               \GM@label[offset=-3.8mm]{thetaSN}{$\theta_{SN,i}$}
\rput(1.5,2.2){\GM@node[nodeSize=11mm,observed=true,
                         observedColor=lightgray,query=true]{KSN}}
               \GM@label[offset=-4.5mm]{KSN}{$K_{SN,i}$}
\rput(1.5,1.0){\GM@node[nodeSize=11mm,observed=true,
                         observedColor=lightgray,query=true]{NSN}}
               \GM@label[offset=-4.5mm]{NSN}{$N_{SN,i}$}

\rput(4.0,5.8){\GM@node[nodeSize=11mm]{alpha}}
               \GM@label[offset=-1.5mm]{alpha}{$\alpha_i$}
\rput(4.0,4.6){\GM@node[nodeSize=9mm]{phiSB}}
               \GM@label[offset=-3.8mm]{phiSB}{$\phi_{SB,i}$}
\rput(4.0,3.4){\GM@node[nodeSize=9mm]{thetaSB}}
               \GM@label[offset=-3.8mm]{thetaSB}{$\theta_{SB,i}$}
\rput(4.0,2.2){\GM@node[nodeSize=11mm,observed=true,
                         observedColor=lightgray,query=true]{KSB}}
               \GM@label[offset=-4.5mm]{KSB}{$K_{SB,i}$}
\rput(4.0,1.0){\GM@node[nodeSize=11mm,observed=true,
                         observedColor=lightgray,query=true]{NSB}}
               \GM@label[offset=-4.5mm]{NSB}{$N_{SB,i}$}
```

3. Arrows

The third step consists of defining the arrows with one line of code each, connecting parent nodes to child nodes. In case at least one of the nodes is a deterministic nodes (consisting of an inner and outer node), the outer node is specified. The resulting graphical model after adding the code below is presented in Figure 5.C.

```
\ncline[arrows=->]{phiSN}{thetaSN_outer}
\ncline[arrows=->]{thetaSN_outer}{KSN}
\ncline[arrows=->]{NSN}{KSN}
\ncline[arrows=->]{phiSB_outer}{thetaSB_outer}
\ncline[arrows=->]{thetaSB_outer}{KSB}
\ncline[arrows=->]{NSB}{KSB}
\ncline[arrows=->]{phiSN}{phiSB_outer}
\ncline[arrows=->]{alpha}{phiSB_outer}
\ncline[arrows=->]{muphiSN}{phiSN}
\ncline[arrows=->]{sdphiSN}{phiSN}
\ncline[arrows=->]{mualpha_outer}{alpha}
\ncline[arrows=->]{sdalpha}{alpha}
\ncline[arrows=->]{delta}{mualpha_outer}
\ncline[arrows=->]{sdalpha}{mualpha_outer}
```

4. Plates

To account for the hierarchical structure (the 74 participants), a plate is added. The plate should comprise all quantities with an index i: $K_{SN,i}$, $N_{SN,i}$, $\theta_{SN,i}$, $\phi_{SN,i}$, $K_{SB,i}$, $N_{SB,i}$, $\theta_{SB,i}$, $\phi_{SB,i}$ and α_i. The label is placed in the bottom right corner and the index i covers the 74 exchangeable replicates (see Figure 5.D for the result).

```
\rput(.75,0.1){\GM@plate[plateLabelPos=br]{4.0}{6.4}{$i=1,\ldots,74$}}
```

5. Model equations

Finally, the full model equations should clarify the exact relations between parent and child nodes. Notice that `name` has been set to the word *equation* for each equation, as the identification of the equation lines is not important. All equations are written in the math environment. After adding the equation lines, the graphical model looks like Figure 5.E.

```
\rput(5.5,7.6){\pnode{equation}}\GM@label{equation}
  {$K_{SN,i} \sim \mathrm{Binomial} \bigl( \theta_{SN,i},N_{SN,i} \bigr)$}
\rput(5.5,6.8){\pnode{equation}}\GM@label{equation}
  {$\theta_{SN,i} = \Phi \bigl( \phi_{SN,i} \bigr)$}
\rput(5.5,6.2){\pnode{equation}}\GM@label{equation}
  {$\phi_{SN,i} \sim \mathrm{Normal} \bigl(\mu_\phi,\sigma_\phi^2 \bigr)$}
\rput(5.5,5.6){\pnode{equation}}\GM@label{equation}
  {$K_{SB,i} \sim \mathrm{Binomial} \bigl( \theta_{SB,i},N_{SB,i} \bigr)$}
\rput(5.5,5.0){\pnode{equation}}\GM@label{equation}
  {$\theta_{SB,i} = \Phi \bigl( \phi_{SB,i} \bigr)$}
```

```
\rput(5.5,4.4){\pnode{equation}}\GM@label{equation}
  {$\phi_{SB,i} = \phi_{SN,i} + \alpha_i$}
\rput(5.5,3.8){\pnode{equation}}\GM@label{equation}
  {$\alpha_i \sim \mathrm{Normal} \bigl(\mu_\alpha, \sigma_\alpha^2 \bigr)$}
\rput(5.5,3.2){\pnode{equation}}\GM@label{equation}
  {$\mu_\phi \sim \mathrm{Normal}_{(0,+ \infty)} \bigl( 0,1 \bigr)$}
\rput(5.5,2.6){\pnode{equation}}\GM@label{equation}
  {$\sigma_\phi \sim \mathrm{Uniform} \bigl( 0,10 \bigr)$}
\rput(5.5,2.0){\pnode{equation}}\GM@label{equation}
  {$\mu_\alpha = \delta \times \sigma_\alpha$}
\rput(5.5,1.4){\pnode{equation}}\GM@label{equation}
  {$\sigma_\alpha \sim \mathrm{Uniform} \bigl( 0,10 \bigr)$}
\rput(5.5,0.8){\pnode{equation}}\GM@label{equation}
  {$\delta \sim \mathrm{Normal} \bigl( 0,1 \bigr)$}
```

6. End document

The final three lines of code are closing the document. Now you can also comment out the `\showgrid` command in the grid section. The graphical model is ready and looks like the one in Figure 5.F.

```
\end{pspicture}
\endTeXtoEPS
\end{document}
```

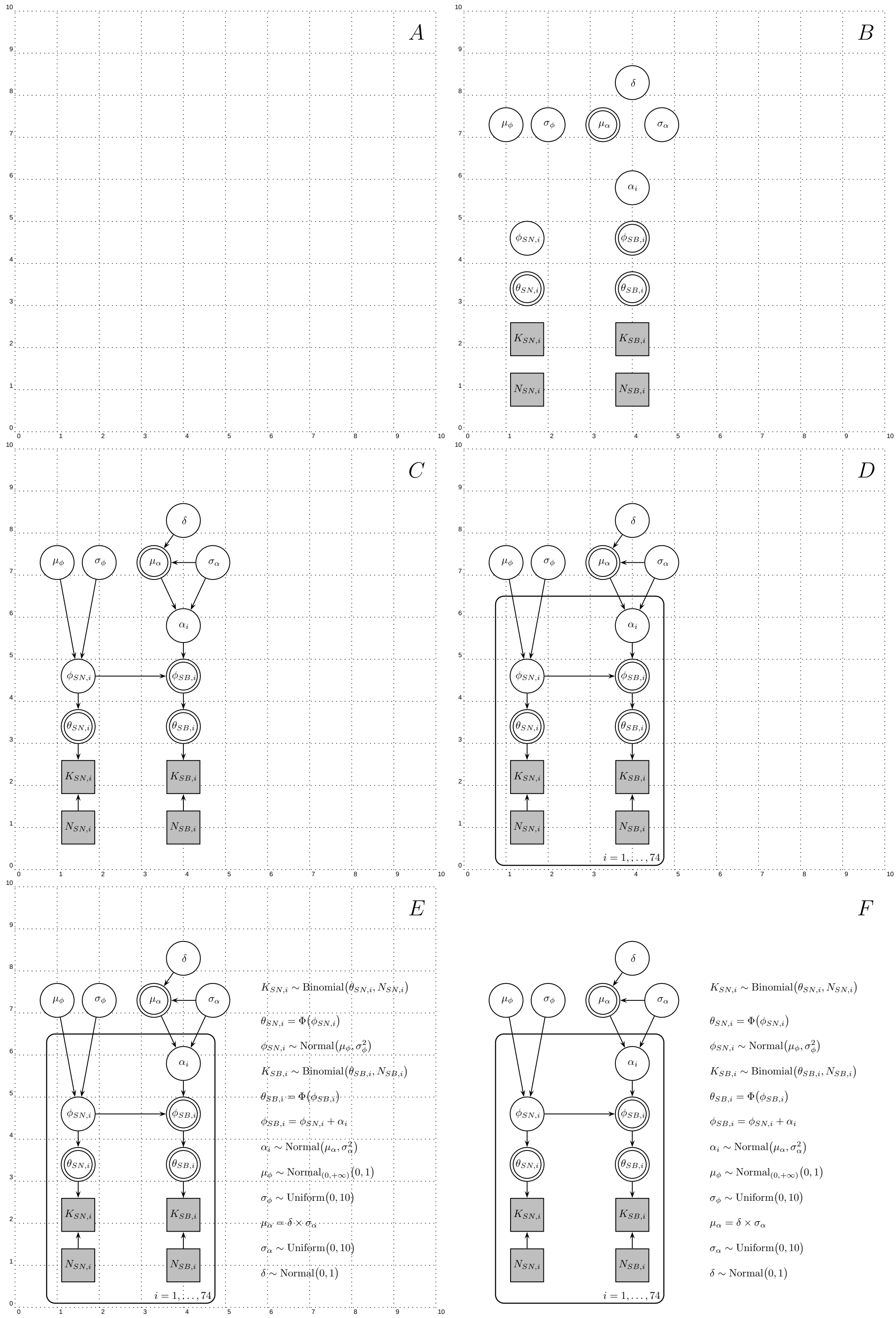


Figure 5: Different stages in the construction of the graphical model.

4 References

Gilks, W.R., Thomas, A., & Spiegelhalter, D. J. (1994). A language and program for complex Bayesian modelling. *The Statistician, 43*, 169-177.

Lee, M. D., & Wagenmakers, E.-J. (2009). A course in Bayesian graphical modeling for cognitive science. *Course notes available at* `http://www.socsci.uci.edu/~mdlee/bgm.html`.

Lodewyckx, T., Lee, M. D., & Wagenmakers, E.-J. (2009). Bayesian hypothesis testing using product space transdimensional MCMC. *Manuscript in preparation.*

Spiegelhalter, D. J. (1998). Bayesian graphical modelling: a case-study in monitoring health outcomes. *Applied Statistics, 47*, 115-133.