ADVANCED BAYESIAN MODELING

HIERARCHICAL MODELING FUNDAMENTALS:

HIERARCHICAL MODEL REPRESENTATIONS

Recall a hierarchical model considered in rat tumor example:

$$
\begin{aligned}
y_j \mid \theta_j &\sim \mathrm{Bin}(n_j, \theta_j) \\
\theta_j \mid \alpha, \beta &\sim \mathrm{Beta}(\alpha, \beta) \\
\alpha, \beta &\sim \text{indep. Expon}(\lambda)
\end{aligned}
$$

(λ is specified; conditional independence is implicit.)

Is there a representation that makes hierarchical structure more obvious?

Graphical Models

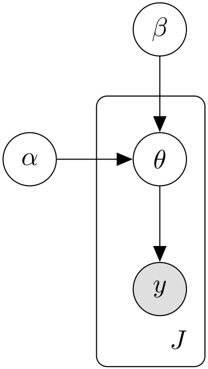
The figure on the right represents a **graphical model**.

Variables are **nodes** and connections are **edges**.

It is a **directed acyclic graph (DAG)**: All edges are arrows in one direction, and there are no cycles.

Each variable in a circled node is random. If the circle is shaded, the variable is observed (data).

The rounded rectangle is a **plate**: All nodes on the plate represent variables that are vectors of length J.

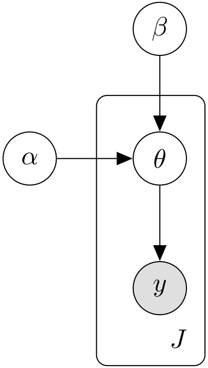


Each edge is from a **parent** to a **child**.

E.g., θ is a child of parents α and β.

In a hierarchical representation, a child's distribution is specified conditionally on its parents only.

Top-level variables (α and β) have distributions specified unconditionally (marginally). They are (marginally) independent if they occupy different nodes.

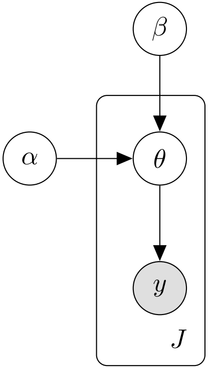


On a plate, different elements of the vector variables are assumed conditionally independent (given the parents).

E.g., the θ_js are conditionally independent given α and β.

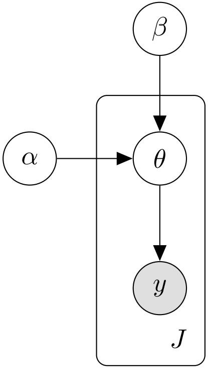
A parent-child dependence on a plate usually indicates that each child *element* conditionally depends only on the corresponding parent *element*.

E.g., y_j (conditionally) depends only on θ_j.



Compare:

$$
\begin{aligned}
y_j \mid \theta_j &\sim \text{Bin}(n_j, \theta_j) \qquad j = 1, \ldots, J \\
\theta_j \mid \alpha, \beta &\sim \text{Beta}(\alpha, \beta) \qquad j = 1, \ldots, J \\
\alpha, \beta &\sim \text{indep. Expon}(\lambda)
\end{aligned}
$$



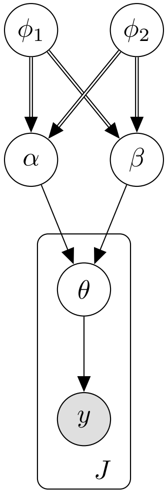
($n = (n_1, \ldots, n_J)$ could be included, but would be a *constant* node on the plate)

Alternative model (BDA3, Sec. 5.3):

$$
\begin{aligned}
y_j \mid \theta_j &\sim \mathrm{Bin}(n_j, \theta_j) \qquad j = 1, \ldots, J \\
\theta_j \mid \alpha, \beta &\sim \mathrm{Beta}(\alpha, \beta) \qquad j = 1, \ldots, J
\end{aligned}
$$

$$
\phi_1 = \frac{\alpha}{\alpha + \beta} \sim \mathrm{U}(0, 1)
$$

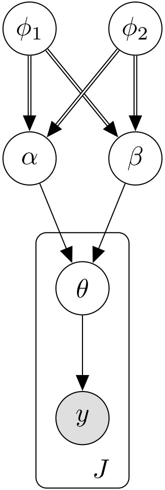
$$
\phi_2 = (\alpha + \beta)^{-1/2} \sim \text{flat on } (0, \infty)
$$



Double arrows represent deterministic relationships.

Nodes α and β are **deterministic**: Defined as an exact function of their parents.

All other nodes are **stochastic**: Defined in terms of a distribution. Their parents (if any) define parameters of the distribution.



DAG models are well-defined – joint distribution exists and is unique.

Advice: Always make sure your model is a DAG model (unless you really know what you are doing).

Bayesian Simulation Software

BUGS: Bayesian inference Using Gibbs Sampling – software project

- WinBUGS
- OpenBUGS – Windows, Linux, Mac (under Wine)
- JAGS: Just Another Gibbs Sampler

All attempt to automate posterior simulation, requiring only a DAG model to be specified.

BUGS Modeling Languages

BUGS project developed specialized language for model specification, based on DAG models.

JAGS uses a variant of this language.

We will use the JAGS variant, described in manual here:
`https://sourceforge.net/projects/mcmc-jags/files/`

Compare:

$$
\begin{aligned}
y_j \mid \theta_j &\sim \text{Bin}(n_j, \theta_j) \quad j = 1, \ldots, J \\
\theta_j \mid \alpha, \beta &\sim \text{Beta}(\alpha, \beta) \quad j = 1, \ldots, J \\
\alpha, \beta &\sim \text{indep. Expon}(0.001)
\end{aligned}
$$

```
model {

  for (j in 1:J) {
    y[j] ~ dbin(theta[j], n[j])
    theta[j] ~ dbeta(alpha, beta)
  }

  alpha ~ dexp(0.001)
  beta ~ dexp(0.001)

}
```

In JAGS, "~" defines a **stochastic relation**: Variable on the left-hand side is a stochastic node.

Note parameterization of `dbin`. Always check JAGS manual!

Consider this alternative model:

$$\phi_1 \;=\; \frac{\alpha}{\alpha+\beta} \;\sim\; \mathrm{U}(0,1)$$

$$\phi_2 \;=\; (\alpha+\beta)^{-1/2} \;\sim\; \mathrm{U}(0,1000)$$

$$\alpha \;=\; \phi_1/\phi_2^2$$

$$\beta \;=\; (1-\phi_1)/\phi_2^2$$

```
model {

  for (j in 1:J) {
    y[j] ~ dbin(theta[j], n[j])
    theta[j] ~ dbeta(alpha, beta)
  }

  alpha <- phi1 / phi2^2
  beta <- (1-phi1) / phi2^2

  phi1 ~ dunif(0,1)
  phi2 ~ dunif(0,1000)

}
```

In JAGS, “<-” defines a **deterministic relation**: Variable on left-hand side is a deterministic (or **logical**) node.

Notes about JAGS:

- Statements within a block may be listed in any order.
- Improper priors are not allowed.
- Data values are not allowed for deterministic nodes.
- See full manual for definitions of distributions.