BUGS

Air: Berkson measurement error

Whittemore and Keller (1988) use an approximate maximum likelihood approach to analyse the data shown below on reported respiratory illness versus exposure to nitrogen dioxide (NO_2) in 103 children. Stephens and Dellaportas (1992) later use Bayesian methods to analyse the same data.

	Bedroom NO_2 level in ppb (z)			
Respiratory illness (y)	<20	20–40	40+	Total
Yes	21	20	15	56
No	27	14	6	47
Total	48	34	21	103

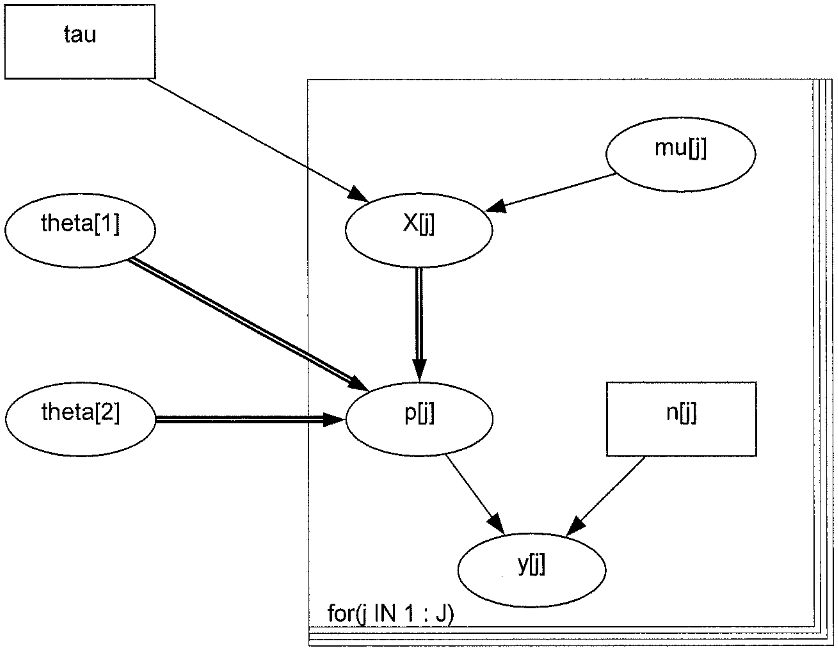
A discrete covariate z_j ($j = 1,2,3$) representing NO_2 concentration in the child's bedroom classified into 3 categories is used as a surrogate for true exposure. The nature of the measurement error relationship associated with this covariate is known precisely via a calibration study, and is given by

$$x_j = \alpha + \beta z_j + \varepsilon_j$$

where $\alpha = 4.48$, $\beta = 0.76$ and ε_j is a random element having normal distribution with zero mean and variance σ^2 $(= 1/\tau) = 81.14$. Note that this is a Berkson (1950) model of measurement error, in which the true values of the covariate are expressed as a function of the observed values. Hence the measurement error is independent of the latter, but is correlated with the true underlying covariate values. In the present example, the observed covariate z_j takes values 10, 30 or 50 for j = 1, 2, or 3 respectively (i.e. the mid-point of each category), whilst x_j is interpreted as the "true average value" of NO_2 in group j. The response variable is binary, reflecting presence/absence of respiratory illness, and a logistic regression model is assumed. That is

$$y_j \sim \text{Binomial}(p_j, n_j)$$
$$\text{logit}(p_j) = \theta_1 + \theta_2 x_j$$

where p_j is the probability of respiratory illness for children in the jth exposure group. The regression coefficients θ_1 and θ_2 are given vague independent normal priors. The graphical model is shown below:




```
model
{
  for(j in 1 : J) {
    y[j] ~ dbin(p[j], n[j])
    logit(p[j]) <- theta[1] + theta[2] * X[j]
    X[j] ~ dnorm(mu[j], tau)
    mu[j] <- alpha + beta * Z[j]
  }
  theta[1] ~ dnorm(0.0, 0.001)
  theta[2] ~ dnorm(0.0, 0.001)
}
```

Data

list(J = 3, y = c(21, 20, 15), n = c(48, 34, 21), Z = c(10, 30, 50), tau = 0.01234, alpha = 4.48, beta = 0.76)

Inits

list(theta = c(0.0, 0.0), X = c(0.0, 0.0, 0.0))

Results

A 1000 update burn in followed by a further 10000 updates gave the parameter estimates

a) Without over-relaxation.

node	mean	sd	MC error	2.5%	median	97.5%
X[1]	12.57	7.979	0.204	-4.034	12.83	27.17
X[2]	27.17	7.494	0.1085	12.66	27.08	42.2
X[3]	41.33	8.4	0.1784	25.43	41.28	57.97
theta[1]	-0.8276	0.7515	0.03839	-2.812	-0.6843	0.2217
theta[2]	0.04355	0.02906	0.001501	0.003022	0.03805	0.1201

b) With over-relaxation.

node	mean	sd	MC error	2.5%	median	97.5%
X[1]	12.87	8.185	0.1593	-3.643	13.08	27.74
X[2]	27.42	7.432	0.05592	12.81	27.44	42.42
X[3]	41.43	8.472	0.1318	25.31	41.26	58.49
theta[1]	-0.893	0.898	0.03741	-3.445	-0.6819	0.2371
theta[2]	0.04524	0.03292	0.001379	0.00294	0.03772	0.1404

Re-parameterised model with centred covariates:

```
model
    {
     for( j in 1 : J ) {
       y[j] ~ dbin(p[j],n[j])
       logit(p[j]) <- theta0+ theta[2] * (X[j] - mean(mu[]))
       X[j] ~ dnorm(mu[j],tau)
       mu[j] <- alpha + beta * Z[j]
     }
     theta0 ~ dnorm(0.0,0.001)
     theta[2] ~ dnorm(0.0,0.001)
     theta[1] <- theta0 - theta[2] * mean(mu[])
    }
```

Data

```
list(J = 3, y = c(21, 20, 15), n = c(48, 34, 21), Z = c(10, 30, 50), tau = 0.01234,
alpha = 4.48, beta = 0.76)
```

Inits

```
list(theta = c(NA, 0.0), theta0 = 0.0, X = c(0.0, 0.0, 0.0))
```

Results

A 1000 update burn in followed by a further 10000 updates gave the parameter estimates, with over-relaxation.

node	mean	sd	MC error	2.5%	median	97.5%
X[1]	13.49	8.595	0.144	-3.65	13.65	29.75
X[2]	27.37	7.4	0.06966	13.04	27.31	42.18
X[3]	40.8	8.612	0.1284	24.35	40.69	57.81
theta[1]	-1.027	1.842	0.06678	-5.001	-0.7077	0.3557
theta[2]	0.05012	0.0671	0.002496	-0.003214	0.03884	0.1966