BUGS 0.5*Examples
Volume 2 (version *ii*)

David Spiegelhalter **Andrew Thomas** **Nicky Best**
Wally Gilks

MRC Biostatistics Unit, Institute of Public Health,
Robinson Way, Cambridge CB2 2SR

Tel: 44-1223-330300 *Fax: 44-1223-330388*
e-mail: bugs@mrc-bsu.cam.ac.uk *ftp: ftp.mrc-bsu.cam.ac.uk*

August 14, 1996

Introduction and Disclaimer

These worked examples illustrate the use of the BUGS language and sampler in a wide range of problems. They contain a number of useful "tricks", but are certainly not exhaustive of the models that may be analysed.

We emphasise that all the results for these examples have been derived in the most naive way: in general a burn-in of 500 iterations and a single long run of 1000 iterations. This is not recommended as a general technique: no tests of convergence have been carried out, and traces of the estimates have not even been plotted. However, comparisons with published results have been made where possible. Times have been measured on a 60 MHz superSPARC: a 60 MHz Pentium PC appears to be about 4 times slower, and a 30 MHz superSPARC about 2 times slower.

Users are warned to be extremely careful about assuming convergence, especially when using complex models including errors in variables, crossed random effects and intrinsic priors in undirected models.

 The support of the Economic and Social Research Council (UK) is gratefully acknowledged. The work was funded in part by ESRC (UK) Award Number H519 25 5023.

Warning

BUGS version 0.5

Release date: August 14, 1996

BUGS version 0.5 released on August 14, 1996 is a TEST version only.

If you encouter any errors in the program, please notify us by e-mailing bugs@mrc-bs.cam.ac.uk. In particular, users are warned that BUGS version 0.5 may crash during sampling with the error

```
Can not locate mode of sampling density
```

or

```
Allowed number of function evaluations exceeded for ARS.
```

Such errors typically occur when estimating models involving a log or logit function of parameters whose values are very close to zero. We are currently working to fix this bug, and will release a revised version 0.5 when this has been sorted out. Please note that the Cosmos example in BUGS Examples Volume 2 crashes with this error when running BUGS version 0.5, although the model can be run successfully using BUGS version 0.30.

Contents

1 Dugongs: a nonconjugate, nonlinear model **4**

2 Biops: discrete variable latent class models **6**

3 Eyes: normal mixture models **9**

4 Hearts: a mixture model for count data **11**

5 Air: covariate measurement error **13**

6 Cervix: case-control study with errors in covariates **15**

7 Jaw: repeated measures analysis of variance **18**

8 Birats: a bivariate Normal hierarchical model **21**

9 Schools: ranking school examination results using multivariate hierarchical models **25**

10 Ice: non-parametric smoothing in an age-cohort model **30**

10.1 Autoregressive smoothing of relative risks 30

10.2 An undirected model using an intrinsic prior for the random effects 32

10.3 An intrinsic prior with a hyperparameter 34

11 Lips: spatial smoothing of cancer rates **37**

11.1 Spatial smoothing using an intrinsic prior 38

11.2 Spatial model with intrinsic prior and hyperparameter. 40

12 Beetles: logistic, probit and extreme value (log-log) model comparison **43**

13 Pines: Bayes factors for selecting regression models **47**

14 Alli: multinomial-logistic models **51**

15 Endo: conditional inference in case-control studies **55**

16 Asia: a simple expert system **58**

16.1 Evidence propagation 58

16.2 Learning about parameters 59

17 Pigs: **genetic counselling and pedigree analysis** **62**

18 Cosmos: **flexible mean and variance relationships using Legendre polynomial basis functions** **66**

19 Marsbars: **order constraints in two-way ANOVA** **70**

20 Stagnant: **a changepoint problem** **72**

1 Dugongs: a nonconjugate, nonlinear model

Carlin and Gelfand (1991) present a nonconjugate Bayesian analysis of the following data set from Ratkowsky (1983):

Dugong	1	2	3	4	5	....	26	27
Age (X)	1.0	1.5	1.5	1.5	2.5	....	29.0	31.5
Length(Y)	1.80	1.85	1.87	1.77	2.02	....	2.72	2.57

The data are length and age measurements for 27 captured dugongs (sea cows). Carlin and Gelfand (1991) model this data using a nonlinear growth curve with no inflection point and an asymptote as X_i tends to infinity:

$$\begin{aligned} Y_i &\sim \text{Normal}(\mu_i, \tau), \qquad i = 1, ..., 27 \\ \mu_i &= \alpha - \beta\gamma^{X_i} \qquad \alpha, \beta > 1; 0 < \gamma < 1 \end{aligned}$$

Standard noninformative priors are adopted for α, β and τ, and a uniform prior on (0,1) is assumed for γ. However, this specification leads to a non conjugate full conditional distribution for γ which is also non log-concave. This problem may be handled within BUGS by discretizing γ, and specifying equal prior probabilities for each discrete value. The BUGS code is shown below, and the graph is given in Figure 1.

```
model dugongs;
const
   N = 27,  # number of observations
   M = 128; # number of bins for gamma
var
  x[N],Y[N],mu[N],alpha,beta,gamma,tau,sigma,p[M],iGamma,U1,U2,U3;
data x, Y in "dugongs.dat";
inits in "dugongs.in";
{
  for (i in 1:N) {
     mu[i] <- alpha - beta*pow(gamma,x[i]);
     Y[i] ~ dnorm(mu[i],tau)
  }
  alpha ~ dnorm(0.0,1.0E-4);
  beta ~ dnorm(0.0,1.0E-4);
  tau ~ dgamma(1.0E-3,1.0E-3); sigma <- 1.0/sqrt(tau);

  iGamma ~ dcat(p[]);  # discretize gamma
  gamma <- iGamma/M;   # normalize discretized gamma to range (0,1)
  for (j in 1:M) { p[j] <- 1/M }   # equal prior for all values of iGamma

# Transform alpha, beta and gamma to scale used by Carlin and Gelfand
  U1 <- log(alpha);
  U2 <- log(beta);
  U3 <- logit(gamma);
}
```

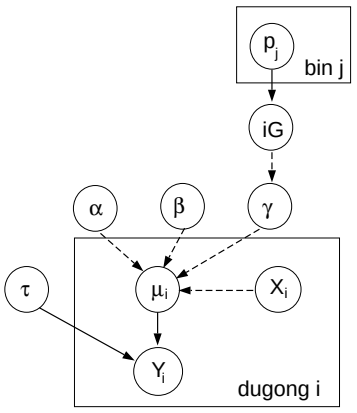

Figure 1: Graphical model for `dugongs` example.

Analysis

After a 500 iteration burn-in, 1000 iterations took 1 minutes 41 seconds (using 128 bins for discretizing γ). The results are shown below, together with those of Carlin and Gelfand, and Ratkowsky. Results are also given for 1000 iteration BUGS runs using 64, 32, 16, and 8 bins for γ, to illustrate the change in precision incurred by using a coarser categorization. Note that the speed of running this model in BUGS is approximately proportional to the number of bins, with the 8 bin model taking as little as 9 seconds for 1000 iterations.

	U1 (log α)	**U2 (log β)**	**U3 (logit γ)**	σ
C & G posterior mode	0.975	-0.014	1.902	–
Ratkowsky least squares estimate	0.981	-0.028	1.932	–
BUGS posterior mode (95% interval)				
128 bins	0.979 (0.933, 1.032)	-0.0291 (-0.180, 0.117)	1.896 (1.366, 2.364)	0.098 (0.074, 0.129)
64 bins	0.977 (0.934, 1.023)	-0.030 (-0.183, 0.109)	1.880 (1.366, 2.268)	0.098 (0.074, 0.128)
32 bins	0.984 (0.932, 1.034)	-0.024 (-0.167, 0.123)	1.941 (1.272, 2.268)	0.098 (0.075, 0.131)
16 bins	0.976 (0.933, 1.002)	-0.034 (-0.183, 0.116)	1.883 (1.466, 1.945)	0.097 (0.074, 0.127)
8 bins	0.981 (0.958, 1.003)	-0.036 (-0.182, 0.099)	1.945 (1.945, 1.945)	0.096 (0.073, 0.127)

We note that the BUGS estimates and 95% intervals for $\log \alpha$, $\log \beta$ and σ are virtually unaffected by the number of bins chosen for γ. However, the 95% interval estimate for logit γ itself is too precise for the 16 and 8 bin models because the bin width is too coarse and nealry all the sampled values for γ fall within the same interval. The models with 32 or more bins give more realistic interval estimates.

2 Biops: discrete variable latent class models

Spiegelhalter and Stovin (1983) presented data on repeated biopsies of transplanted hearts, in which a total of 414 biopsies had been taken at 157 sessions. Each biopsy was graded on evidence of rejection using a 4-category scale of none (O), minimal (M), mild (+) and moderate-severe (++). Part of the data is shown below.

Combination			Multinomial response	Session frequency
O	O		(2, 0, 0, 0)	12
M	M	O	(1, 2, 0, 0)	10
+	+	O	(1, 0, 2, 0)	17
++	++	++	(0, 0, 0, 3)	5

The sampling procedure may not detect the area of maximum rejection, which is considered the true underlying state at the time of the session and denoted t_i — the underlying probability distribution of the four true states is denoted by the vector p. It is then assumed that each of the observed biopsies are conditionally independent given this true state with the restriction that there are no 'false positives': i.e. one cannot observe a biopsy worse than the true state. We then have the sampling model

$$\begin{aligned} b_i &\sim \text{Multinomial}(e_{t_i}, n_i) \\ t_i &\sim \text{Categorical}(p) \end{aligned}$$

where b_i denotes the multinomial response at session i where n_i biopsies have been taken, and e_{jk} is the probability that a true state $t_i = j$ generates a biopsy in state k. The no-false-positive restriction means that $e_{12} = e_{13} = e_{14} = e_{23} = e_{24} = e_{34} = 0$. Spiegelhalter and Stovin (1983) estimated the parameters e_j and p using the EM algorithm, with some smoothing to avoid zero estimates.

The appropriate graph is shown in Figure 2, where the role of the true state t_i is simply to pick the appropriate row from the 4 x 4 error matrix e. Here the probability vectors e_j $(j = 1, ..., 4)$ and p are assumed to have uniform priors on the unit simplex, which correspond to Dirichlet priors with all parameters being 1.

The BUGS code for this model is given below. No initial value file is provided, since the forward sampling procedure will find a configuration of starting values that is compatible with the expressed constraints. It has been necessary, however, to introduce dummy arrays for the separate rows of the error matrix since they have different lengths. We also note the apparent "cycle" in the graph created by the expression `nbiops[i] <- sum(biopsies[i,])`. This will lead BUGS to generate the warning message `Possible directed cycle or undirected link in model` during compilation. Such "cycles" are permitted provided that they are only data transformation statements, since this does not affect the essential probability model.

Biops: **model specification in** BUGS

```
model biops;
const
    ns=157;    #  number of sessions
var
    biopsies[ns,4], #  grades observed in ith session (multinomial)
    nbiops[ns],     # total number of biopsies in ith session
    true[ns],       # true state in ith session
    error[4,4],     # error matrix in taking biopsies
    error2[2],      # non-zero elements of error[2,]
    error3[3],      # non-zero elements of error[3,]
    p[4],           # underlying   incidence of true  states
    prior2[2],      # prior parameters for error2
    prior3[3],      # prior parameters for error3
    prior4[4];      # prior on p and error[4,] (fixed in data-file)
data biopsies in "biops.dat";
{
# TRANSFORMATIONS

   for (i in 1:ns){
      nbiops[i]    <- sum(biopsies[i,]);
   }

# MODEL

   for (i in 1:ns){
      true[i]       ~ dcat(p[]);
      biopsies[i,]  ~ dmulti(error[true[i],],nbiops[i]); # multinomial
   }

#  force no false positives

   error[1,1] <- 1; error[1,2] <- 0; error[1,3] <- 0; error[1,4] <- 0;
   error[2,3] <- 0; error[2,4] <- 0; error[3,4] <- 0;

#  priors for parameters

   prior2[1] <- 1; prior2[2] <- 1;
   prior3[1] <- 1; prior3[2] <- 1; prior3[3] <- 1;
   prior4[1] <- 1; prior4[2] <- 1; prior4[3] <- 1; prior4[4] <- 1;

   error2[] ~ ddirch(prior2[]); for (j in 1:2) {error[2,j] <- error2[j]}
   error3[] ~ ddirch(prior3[]); for (j in 1:3) {error[3,j] <- error3[j]}
   error[4,] ~ ddirch(prior4[]);

   p[]       ~ ddirch(prior4[]);     # prior for p
}
```

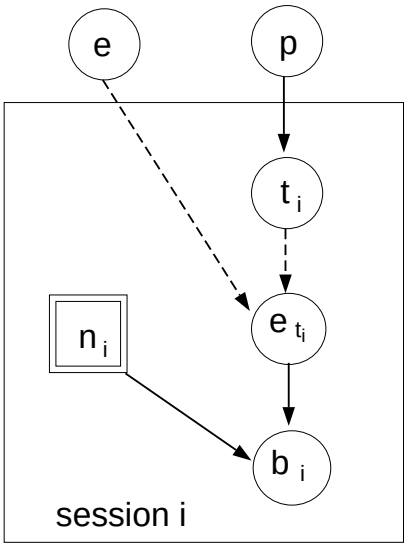

Figure 2: Graphical model for `biops` example

Analysis

A simple `BUGS` run took 21 seconds for 1000 iterations and gave the following results which are similar to those obtained by the EM algorithm.

parameter (%)	EM algorithm	`BUGS`
p_1	20 ± 4.5	16 ± 4.6
p_2	28 ± 4.5	31 ±5.1
p_3	35 ± 4.3	39 ±4.3
p_4	17 ± 3.4	15 ±3.0
e_{11}	100	100
e_{21}	53 ± 6.4	58 ± 6.6
e_{22}	47 ± 6.4	42 ± 6.6
e_{31}	33 ± 4.5	34 ± 4.4
e_{32}	4 ± 1.7	4 ± 1.8
e_{33}	63 ± 4.7	62 ± 4.7
e_{41}	11 ± 5.0	10 ± 4.4
e_{42}	1 ± –	2 ± 2.4
e_{43}	25 ± 7.4	20 ± 5.7
e_{44}	64 ± 8.5	67 ± 7.1

3 Eyes: normal mixture models

Bowmaker *et al.* (1985) analyse data on the peak sensitivity wavelengths for individual microspectrophotometric records on a small set of monkey's eyes. Data for one monkey ($S14$ in the paper) are given below (500 has been subtracted from each of the 48 measurements).

29.0	30.0	32.0	33.1	33.4	33.6	33.7	34.1	34.8	35.3
35.4	35.9	36.1	36.3	36.4	36.6	37.0	37.4	37.5	38.3
38.5	38.6	39.4	39.6	40.4	40.8	42.0	42.8	43.0	43.5
43.8	43.9	45.3	46.2	48.8	48.7	48.9	49.0	49.4	49.9
50.6	51.2	51.4	51.5	51.6	52.8	52.9	53.2		

Part of the analysis involves fitting a mixture of two normal distributions with common variance to this distribution, so that each observation y_i is assumed drawn from one of two groups. Let $T_i = 1, 2$ be the true group of the ith observation, where group j has a normal distribution with mean λ_j and precision τ. We assume an unknown fraction P of observations are in group 2, $1 - P$ in group 1. The model is thus

$$\begin{aligned} y_i &\sim \text{Normal}(\lambda_{T_i}, \tau) \\ T_i &\sim \text{Categorical}(P). \end{aligned}$$

We note that this formulation easily generalises to additional components to the mixture, although for identifiability an order constraint must be put onto the group means.

Robert (1994) points out that when using this model, there is a danger that at some iteration, *all* the data will go into one component of the mixture, and this state will be difficult to escape from — this matches our experience. Robert suggests a re-parameterisation, a simplified version of which is to assume

$$\lambda_2 = \lambda_1 + \theta, \quad \theta > 0.$$

$\lambda_1, \theta, \tau, P$ are given independent "noninformative" priors, including a uniform prior for P on (0,1). The appropriate graph is shown below, and the BUGS code is given over the page.

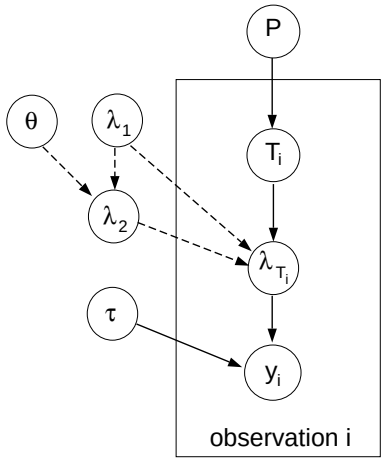

Figure 3: Graphical model for eyes example

Eyes: model specification in BUGS

```
model eyes;

const
     N=48;
var
    y[N],        # observations
    T[N],        # true groups (labelled 1,2)
    lambda[2],   # means of two groups
    theta,       # scaled positive shift between groups
    tau,         # sampling precision
    sigma,       # sampling standard deviation
    P[2],        # proportion in first group
    alpha[2];    # prior parameters for proportions

data y in "eyes.dat";
inits in "eyes.in";
{
   for (i in 1:N){
       y[i]  ~ dnorm(lambda[T[i]],tau);
       T[i]  ~ dcat(P[])
   }
   sigma     <- 1/sqrt(tau);
   tau        ~ dgamma(0.01,0.01);
   lambda[1]  ~ dnorm(0,1.0E-6);
   lambda[2] <- lambda[1]+theta;
   theta      ~ dnorm(0,1.0E-6) I(0,);
   P[]        ~ ddirch(alpha[]);    # prior for mixing proportion
   alpha[1]  <- 1;                  # uniform prior
   alpha[2]  <- 1;
}
```

Analysis

A BUGS run of 1000 iterations was extremely fast (4 seconds) and gave the following results which are compared with the maximum likelihood estimates of Bowmaker *et al.* (1985)

parameter	maximum likelihood	BUGS
λ_1	536.9 ± .7	536.7 ± 0.99
λ_2	549.0 ± 1.1	548.8 ± 1.26
σ	3.45 ± .39	3.80 ± .72
P_1	.62 ± .08	.60 ± .09

We note the appropriately wider intervals provided by the full Bayesian analysis. We also point out that even with this re-parameterization we have experienced problems with some mixture models, in that a component may contain no observations at some iteration. One solution is to force at least one pre-specified observation to be in each component.

4 Hearts: a mixture model for count data

The table below presents data given by Berry (1987) on the effect of a drug used to treat patients with frequent premature ventricular contractions (PVCs) of the heart.

	PVCs per minute		
number (i)	**Pre-drug** (x_i)	**Post-drug** (y_i)	**Decrease**
1	6	5	1
2	9	2	7
3	17	0	17
.	..	..	..
11	9	13	-4
12	51	0	51

Farewell and Sprott (1988) model this data as a mixture distribution of Poisson counts in which some patients are "cured" by the drug, whilst others experience varying levels of response but remain abnormal. A zero count for the post-drug PVC may indicate a "cure", or may represent a sampling zero from a patient with a mildly abnormal PVC count. The following model thus is assumed:

$$\begin{aligned} x_i &\sim \text{Poisson}(\lambda_i) && \text{for all patients} \\ y_i &\sim \text{Poisson}(\beta\lambda_i) && \text{for all } \textit{uncured} \text{ patients} \\ P(\text{cure}) &= \theta \end{aligned}$$

To eliminate the nuisance parameters λ_i, Farewell and Sprott use the conditional distribution of y_i given $t_i = x_i + y_i$. This is equivalent to a binomial likelihood for y_i with denominator t_i and probability $p = \frac{\beta}{1+\beta}$ (see Cox and Hinkley (1974) pp. 136–137 for further details of the conditional distribution for Poisson variables). Hence the final mixture model may be expressed as follows:

$$\begin{aligned} P(y_i = 0 \mid t_i) &= \theta + (1-\theta)(1-p)^{t_i} \\ P(y_i \mid t_i) &= (1-\theta)\binom{t_i}{y_i} p^{y_i}(1-p)^{(t_i - y_i)} \qquad y_i = 1, 2, ..., t_i \end{aligned}$$

The graph for the `hearts` model is shown in Figure 4 and the `BUGS` code follows.

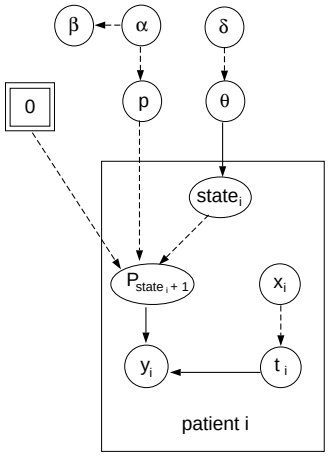

Figure 4: Graphical model for `hearts` example

```
model hearts;
const N = 12;
var
  x[N],y[N],t[N],     # pre-drug, post-drug and total PVC count
  state[N],state1[N], # binary indicator of whether patient is cured
  theta,              # probability of cure (prob of state = 1)
  p, beta,            # p = binomial probability = beta/(1+beta)
  P[2],               # 'pick' variable used to select appropriate
                      # value for the binomial probability depending
                      # on whether state1 = 1 or 2 (not cured or cured)
  alpha, delta;       # p and theta transformed to logit scale for normality

data x, y in "hearts.dat";
inits in "hearts.in";

{
# TRANSFORMATIONS
   for (i in 1:N) {
      t[i] <- x[i] + y[i];
   }
# MODEL
   for (i in 1:N) {
      y[i] ~ dbin(P[state1[i]], t[i]);
      state[i] ~ dbern(theta);
      state1[i] <- state[i]+1;   # state[i] takes values 0 or 1, so need to
                                 # add 1 to get values for use as index on P
   }
   P[1] <- p; P[2] <- 0;
   logit(p) <- alpha; alpha ~ dnorm(0,1.0E-4);
   beta <- exp(alpha);  # beta measures change in rate of PVCs after treatment
   logit(theta) <- delta; delta ~ dnorm(0,1.0E-4)
}
```

Analysis

10000 iterations took 32 seconds after a 1000 iteration burn-in. The posterior means (95% C.I.) are given below, together with Farewell and Sprott's maximum likelihood estimates.

Parameter	BUGS		MLE	
θ	0.572	(0.289, 0.823)	0.575	(0.30, 0.81)
β	0.646	(0.359, 1.055)	0.629	–
p	0.386	(0.264, 0.514)	0.386	(0.27, 0.52)

The BUGS results are in close agreement with those of Farewell and Sprott, and suggest that there is just over a 50% chance of a patient being "cured" ($\theta = 0.572$); if not, the number of PVCs per minute is likely to fall to about 65% ($\beta = 0.646$) of their pre-drug count.

5 Air: covariate 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 $\sigma^2 = 81.14$. 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

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

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 in Figure 5.

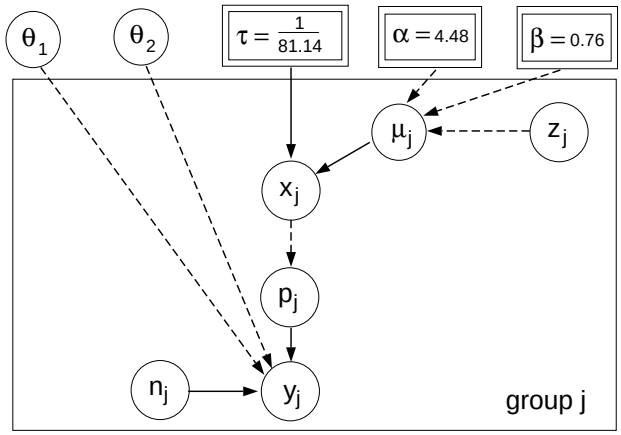

Figure 5: Graphical model for `air` example

Model specification for `air` example

```
model air;
const
   alpha = 4.48,        # intercept of measurement error model
   beta = 0.76,         # slope of measurement error model
   sigma2 = 81.14,      # error variance of measurement error model
   J = 3;               # number of exposure levels for covariate
var
   theta[2],X[J],Z[J],mu[J],p[J],y[J],n[J],tau;
data y, n, Z in "air.dat";
inits in "air.in";
{
   theta[1] ~ dnorm(0.0,1.0E-3);
   theta[2] ~ dnorm(0.0,1.0E-3);
   tau <- 1/sigma2;
   for (j in 1:J) {
      mu[j]       <- alpha + beta*Z[j];
      X[j]         ~ dnorm(mu[j],tau);
      logit(p[j]) <- theta[1] + theta[2]*X[j];
      y[j]         ~ dbin(p[j],n[j]);
   }
}
```

Analysis

2000 iterations took 8 seconds after a 500 iteration burn-in, and produced the following output

variable	estimate	95% interval
θ_1	-0.669	-2.127, 0.218
θ_2	0.038	0.002, 0.098
x_1 (low exposure)	11.6	-4.6, 26.2
x_2 (medium exposure)	27.3	11.7, 42.4
x_3 (high exposure)	41.8	25.3, 58.7

These results should be compared with the plots shown by Stephens and Dellaportas (1992). The posterior mean for $\{\theta_1, \theta_2\}$ is also similar to that obtained by Whittemore and Keller (1988), although their maximum likelihood analysis yielded considerably smaller standard errors. In addition, note that the posterior mean estimates for the elements of x_1 and x_2 (the "true average exposure" to NO_2 in the low and medium groups) are close to the "prior" values of 10 and 30 selected by Whittemore and Keller. However, the value of x_3 is somewhat lower than its "prior value" of 50, largely because the posterior estimate is "pulled in" by the need to fulfil the linear logistic model assumption.

6 Cervix: case-control study with errors in covariates

Carroll *et al.* (1993) consider the problem of estimating the odds ratio of a disease d in a case-control study where the binary exposure variable is measured with error. Their example concerns exposure to herpes simplex virus (HSV) in women with invasive cervical cancer ($d = 1$) and in controls ($d = 0$). Exposure to HSV is measured by a relatively inaccurate western blot procedure w for 1929 of the 2044 women, whilst for 115 women, it is also measured by a refined or "gold standard" method x. The data are given in the table below. They show a substantial amount of misclassification, as indicated by low sensitivity and specificity of w in the "complete" data, and Carroll *et al.* (1993) also found that the degree of misclassification was significantly higher for the controls than for the cases (p=0.049 by Fisher's exact test).

	d	x	w	Count
Complete data				
	1	0	0	13
	1	0	1	3
	1	1	0	5
	1	1	1	18
	0	0	0	33
	0	0	1	11
	0	1	0	16
	0	1	1	16
Incomplete data				
	1		0	318
	1		1	375
	1		0	701
	1		1	535

They fitted a prospective logistic model to the case-control data as follows

$$\begin{aligned} d_i &\sim \text{Bernoulli}(p_i) \qquad i = 1, ..., 2044 \\ \text{logit}(p_i) &= \beta_{0C} + \beta x_i \qquad i = 1, ..., 2044 \end{aligned}$$

where β is the log odds ratio of disease d. Since the relationship between d and x is only directly observable in the 115 women with "complete" data, and because there is evidence of differential measurement error, the following parameters are required in order to estimate the logistic model

$$\begin{aligned} \phi_{1,1} &= \text{P}(w = 1 \mid x = 0, d = 0) \\ \phi_{1,2} &= \text{P}(w = 1 \mid x = 0, d = 1) \\ \phi_{2,1} &= \text{P}(w = 1 \mid x = 1, d = 0) \\ \phi_{2,2} &= \text{P}(w = 1 \mid x = 1, d = 1) \\ q &= \text{P}(x = 1) \end{aligned}$$

The differential probability of being exposed to HSV ($x = 1$) for cases and controls is calculated as follows

$$
\begin{aligned}
\gamma_1 &= \mathrm{P}(x = 1 \mid d = 1) \\
&= \frac{\mathrm{P}(d = 1 \mid x = 1)\mathrm{P}(x = 1)}{\mathrm{P}(d = 1)} \\
&= \frac{1}{1 + \frac{1+e^{\beta_{0c}+\beta}}{1+e^{\beta_{0c}}}\frac{1-q}{q}} \\
\gamma_2 &= \mathrm{P}(x = 1 \mid d = 0) \\
&= \frac{\mathrm{P}(d = 0 \mid x = 1)\mathrm{P}(x = 1)}{\mathrm{P}(d = 0)} \\
&= \frac{1}{1 + \frac{1+e^{-\beta_{0c}-\beta}}{1+e^{-\beta_{0c}}}\frac{1-q}{q}}
\end{aligned}
$$

The graph for the above model is in Figure 6.

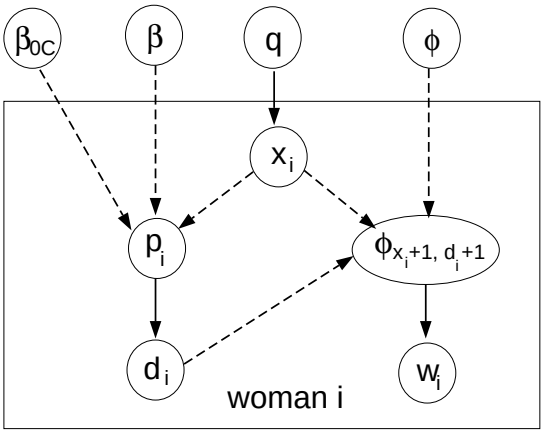

Figure 6: Graphical model for `cervix` example.

The role of the variables `x1` and `d1` is to pick the appropriate value of `phi` (the incidence of `w`) for any given true exposure status `x` and disease status `d`. Since `x` and `d` take the values 0 or 1, and the subscripts for `phi` take values 1 or 2, we must first add 1 to each `x[i]` and `d[i]` before using them as index values for `phi`. `BUGS` does not allow subscripts to be functions of variable quantities — hence the need to create `x1` and `d1` for use as subscripts. In addition, note that γ_1 and γ_2 were not simulated directly in `BUGS`, but were calculated as functions of other parameters. This is because the dependence of γ_1 and γ_2 on d would have led to a cycle in the graphical model which would no longer define a probability distribution.

Cervix: model specification in BUGS

```
model cervix;
const
   N = 2044;  # number of observations
var
     x[N], x1[N],  # 'true' HSV status (x[i] + 1)
     d[N], d1[N],  #  cancer status (d[i] + 1)
            p[N],  #  prob of case
               q,  #  incidence of HSV
  w[N], phi[2,2],  #  approx HSV status; rates for w being positive
    beta0C, beta,  #  intercept and log-odds ratio
  gamma1, gamma2;  #  prob HSV positive given control or case
data d, x, w in "cervix.dat";
inits in "cervix.in";
{
 for (i in 1:N) {
         x[i]   ~ dbern(q);           # incidence of HSV
   logit(p[i]) <- beta0C + beta*x[i]; # logistic model
          d[i]  ~ dbern(p[i]);        # incidence of cancer
         x1[i] <- x[i]+1; d1[i] <- d[i]+1;
          w[i]  ~ dbern(phi[x1[i],d1[i]]); # incidence of w
 }
 q      ~ dunif(0.0,1.0);           # prior distribution
 beta0C ~ dnorm(0.0,0.00001); beta   ~ dnorm(0.0,0.00001);
 for(j in 1:2) {
    for(k in 1:2){ phi[j,k] ~ dunif(0.0,1.0);  }
 }
# calculate gamma1 = P(x=1|d=0) and gamma2 = P(x=1|d=1)
  gamma1 <- 1/(1 + (1+exp(beta0C+beta))/(1+exp(beta0C)) * (1-q)/q);
  gamma2 <- 1/(1 + (1+exp(-beta0C-beta))/(1+exp(-beta0C)) * (1-q)/q);
}
```

Analysis

BUGS took 8 minutes to run for 1000 iterations, following a 500 iteration burn-in. The posterior means and standard errors are shown in the table below, and are compared to the pseudolikelihood (*PSL*) estimates obtained by Carroll *et al.* (1993).

		BUGS		*PSL*	
Parameter		mean	(S.E.)	estimate	(S.E.)
β_{0C}		-0.953	(0.240)	-0.981	(0.185)
β	(log odds ratio)	0.690	(0.416)	0.622	(0.355)
$\phi_{1,1}$	$\mathrm{P}(w=1 \mid x=0, d=0)$	0.307	(0.047)	0.317	(0.057)
$\phi_{1,2}$	$\mathrm{P}(w=1 \mid x=0, d=1)$	0.222	(0.084)	0.195	(0.089)
$\phi_{2,1}$	$\mathrm{P}(w=1 \mid x=1, d=0)$	0.586	(0.065)	0.577	(0.067)
$\phi_{2,2}$	$\mathrm{P}(w=1 \mid x=1, d=1)$	0.749	(0.067)	0.790	(0.067)
γ_1	$\mathrm{P}(x=1 \mid d=0)$	0.441	(0.053)	0.421	(0.057)
γ_2	$\mathrm{P}(x=1 \mid d=1)$	0.608	(0.076)	0.590	(0.079)

7 Jaw: repeated measures analysis of variance

Elston and Grizzle (1962) present repeated measurements of ramus (jaw) bone height on a cohort of 20 boys over an 18 month period:

	Age (in years)			
Subject	**8**	**8.5**	**9**	**9.5**
1	47.8	48.8	49.0	49.7
2	46.4	47.3	47.7	48.4
3	46.3	46.8	47.8	48.5
.	.	.	.	.
.	.	.	.	.
19	46.2	47.5	48.1	48.4
20	46.3	47.6	51.3	51.8
Mean	48.66	49.62	50.57	51.45
variance	6.35	6.45	6.92	7.45

Interest focuses on describing the average growth curve of the ramus bone. The 4 measurements $\underline{Y_i} = \{Y_{i1}, Y_{i2}, Y_{i3}, Y_{i4}\}$ for each child i are assumed to be correlated and follow a multivariate normal (MVN) distribution with unknown population mean vector $\underline{\mu}$ and precision matrix $\underline{\Omega}$. That is $\underline{Y_i} \sim \text{MVN}(\underline{\mu}, \underline{\Omega})$

The following location models for the population mean $\underline{\mu}$ were fitted in turn:

$$\begin{aligned} E(\mu_j) &= \beta_0 && \textit{Constant height} \\ E(\mu_j) &= \beta_0 + \beta_1 x_j && \textit{Linear growth curve} \\ E(\mu_j) &= \beta_0 + \beta_1 x_j + \beta_2 x_j^2 && \textit{Quadratic growth curve} \end{aligned}$$

where x_j = age at jth measurement. Non-informative independent normal priors were specified for the regression coefficients β_0, β_1 and β_2. The population precision matrix $\underline{\Omega}$ was assumed to follow a Wishart(R, ρ) distribution. To represent vague prior knowledge, we chose the the degrees of freedom ρ for this distribution to be as small as possible (i.e. 4, the rank of $\underline{\Omega}$). The scale matrix R was specified as $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$, which represents an assessment of the order of magnitude of the *covariance* matrix $\underline{\Omega}^{-1}$ for $\underline{Y_i}$ (see subsection on the use of the Wishart distribution in the "Multivariate normal nodes" section of the BUGS manual 0.50). Note that except for cases with very few individuals, the choice of R has little effect on the posterior estimate of $\underline{\Omega}^{-1}$ (Lindley, 1970).

Comparision of the fit of the 3 location models may be assessed by calculating the deviance. This is given by −twice the sum of the log-likelihood contributions for each boy i:

$$\text{llike}_i = -\frac{M}{2}\log 2\pi + \frac{1}{2}\log|\underline{\Omega}| - \frac{1}{2}(\underline{Y_i} - \underline{\mu})'\underline{\Omega}(\underline{Y_i} - \underline{\mu})$$

where M=4, the number of measurements per boy. The *change* in (minimum) deviance between the constant and linear models or the linear and quadratic models may be compared to a χ^2 distribution on 1 degree of freedom. In addition, we may compute the residual sum of squares $RSS = \sum_{ij}(Y_{ij} - \mu_j)^2$ for each model.

The graph of the quadratic model is shown in Figure 7, and the BUGS code is given below.

```
model jaw;
const
 M = 4,   # number of time points
 N = 20,  # number of boys
 PI = 3.141593;
var
 Y[N,M], age[M],               # jaw bone length in mm and age
 mu[M],Omega[M,M],Sigma2[M,M], # mean, precision & covariance matrix for Y
 beta0, beta1, beta2,          # regression coefficients for location models
 R[M,M],                       # prior guess at magnitude of Sigma2
 resid[N,M],resid2[N,M],RSS,   # residuals and residual sum of squares
 L1[N,M],llike[N],deviance;    # log-likelihood terms and deviance
data Y in "jawy.dat", age, R in "jawcov.dat";
inits in "jaw.in";
{
  for (i in 1:N) {
     Y[i,] ~ dmnorm(mu[], Omega[,]);  # The 4 measurements for each
  }                                   # boy are multivariate normal

  for(j in 1:M) {     # location model for mean bone length at each age
#    mu[j] <- beta0;                                          # constant
     mu[j] <- beta0 + beta1*age[j];                           # linear
#    mu[j] <- beta0 + beta1*age[j] + beta2*pow(age[j],2);     # quadratic
  }
  beta0 ~ dnorm(0.0, 0.001);
  beta1 ~ dnorm(0.0, 0.001);
  beta2 ~ dnorm(0.0, 0.001);
  Omega[,] ~ dwish(R[,], 4); # between-child variance in length at each age
  Sigma2[,] <- inverse(Omega[,]);

  for (i in 1:N) {
     for  (j in 1:M) {
        resid[i,j] <- Y[i,j] - mu[j];         # residuals
        resid2[i,j] <-pow(resid[i,j], 2);     # squared residuals
        L1[i,j] <- inprod(Omega[j,], resid[i,]);
     }
     llike[i] <- -0.5*(M*log(2*PI) - logdet(Omega[,])
                               + inprod(resid[i,], L1[i,]));
  }
  RSS <- sum(resid2[,]);                     # Residual Sum of Squares
  deviance <- -2 * sum(llike[]);             # Deviance
}
```

Note that at present, BUGS is unable to perform matrix multiplication. Hence to compute $(\underline{Y_i} - \underline{\mu})'\underline{\Omega}(\underline{Y_i} - \underline{\mu})$ we use the `inprod` function to first compute $L_1 = \underline{\Omega}(\underline{Y_i} - \underline{\mu})$, and then to compute $(\underline{Y_i} - \underline{\mu})'L_1$. Also note the use of the `inverse` function to compute `Sigma2`.

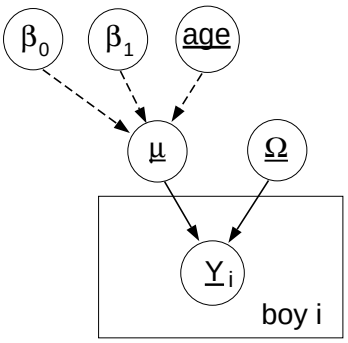

Figure 7: Graphical model for `jaw` example.

Analysis

After a 500 iteration burn-in, 1000 iterations took between 12 and 45 seconds for the 3 models and produced the following output

Variable	**Constant model**	**Linear model**		**Quadratic model**
	BUGS	BUGS	MLE*	BUGS
β_0	49.50 ± 1.05	33.68 ± 1.79	33.75	37.28 ± 2.70
β_1	–	1.87 ± 0.20	1.88	1.04 ± 0.55
β_2	–	–	–	0.05 ± 0.03
μ_1	49.50 ± 1.05	48.63 ± 0.53	48.65 ± 0.55	48.64 ± 0.54
μ_2	49.50 ± 1.05	49.57 ± 0.53	49.62 ± 0.55	49.55 ± 0.54
μ_3	49.50 ± 1.05	50.50 ± 0.54	50.57 ± 0.57	50.48 ± 0.55
μ_4	49.50 ± 1.05	51.43 ± 0.57	51.45 ± 0.60	51.44 ± 0.59
$\Sigma_{1,1}$	8.27 ± 3.69	6.68 ± 2.17	6.01 ± 1.90	6.69 ± 2.18
$\Sigma_{1,2}$	7.19 ± 3.02	6.46 ± 2.12	5.88 ± 1.89	6.46 ± 2.13
$\Sigma_{1,3}$	5.90 ± 2.89	6.01 ± 2.13	5.49 ± 1.87	6.02 ± 2.14
$\Sigma_{1,4}$	4.86 ± 3.20	5.77 ± 2.15	5.27 ± 1.88	5.77 ± 2.15
$\Sigma_{2,2}$	7.61 ± 2.95	6.75 ± 2.18	6.13 ± 1.94	6.76 ± 2.18
$\Sigma_{2,3}$	7.36 ± 3.29	6.39 ± 2.20	5.85 ± 1.93	6.40 ± 2.20
$\Sigma_{2,4}$	7.24 ± 3.93	6.16 ± 2.22	5.63 ± 1.94	6.17 ± 2.22
$\Sigma_{3,3}$	9.23 ± 4.15	7.20 ± 2.42	6.57 ± 2.08	7.22 ± 2.42
$\Sigma_{3,4}$	10.20 ± 5.03	7.19 ± 2.47	6.60 ± 2.12	7.20 ± 2.47
$\Sigma_{4,4}$	12.60 ± 6.19	7.80 ± 2.64	7.09 ± 2.24	7.81 ± 2.64
RSS (*minimum*)	603.1	516.1	516.1	516.1
RSS (*mean* ± *S.E.*)	718.2 ± 163.3	539.8 ± 327.6	–	540.9 ± 336.3
Deviance (*minimum*)	258.5	226.1	–	226.3
Deviance (*mean* ± *S.E.*)	267.5 ± 42.4	236.0 ± 44.0	–	236.5 ± 45.5

*MLE = Maximum likelihood estimates obtainied by Goldstein (1979) (p. 95), and Prosser *et al.* (1991*b*) (p. 106) using the ML3 software.

Examination of the RSS clearly indicates that the linear model is a superior fit to the constant model, but that a quadratic term is unecessary. This is confirmed by the change in (minimum) deviance between successive models: the linear *versus* constant model yields a log likelihood ratio statistic of 32.4 on 1 d.f. (p <0.000001). The quadratic *versus* linear model yield virtually identical deviances, giving a non-significant log likelihood ratio statistic.

8 Birats: a bivariate Normal hierarchical model

We return to the rats example in Volume 1, and illustrate the use of a multivariate Normal (MVN) population distribution for the regression coefficients of the growth curve for each rat. This is the model adopted by Gelfand *et al.* (1990) for this data, and assumes *a priori* that the intercept and slope parameters for each rat are correlated. For example, positive correlation would imply that initially heavy rats (high intercept) tend to gain weight more rapidly (steeper slope) than lighter rats. The model is as follows

$$\begin{aligned} Y_{ij} &\sim \text{Normal}(\mu_{ij}, \tau_c) \\ \mu_{ij} &= \beta_{1i} + \beta_{2i} x_j \\ \underline{\beta_i} &\sim \text{MVN}(\underline{\mu}_\beta, \underline{\Omega}_\beta) \end{aligned}$$

where Y_{ij} is the weight of the ith rat measured at age x_j, and $\underline{\beta_i}$ denotes the vector (β_{1i}, β_{2i}). We assume 'non-informative' independent univariate Normal priors for the separate components μ_{β_1} and μ_{β_2}. A Wishart(R, ρ) prior was specified for $\underline{\Omega}_\beta$, the population precision matrix of the regression coefficients. To represent vague prior knowledge, we chose the the degrees of freedom ρ for this distribution to be as small as possible (i.e. 2, the rank of $\underline{\Omega}_\beta$). The scale matrix $R = \begin{pmatrix} 200 & 0 \\ 0 & 0.2 \end{pmatrix}$. This represents our prior guess at the order of magnitude of the *covariance* matrix $\underline{\Omega}_\beta^{-1}$ for $\underline{\beta_i}$ (see BUGS manual section on Multivariate normal models), and is equivalent to the prior specification used by Gelfand *et al.* Finally, a non-informative Gamma(0.001, 0.001) prior was assumed for the measurement precision τ_c.

The appropriate graphical model is shown in Figure 8, and the BUGS code is given below. Note the use of the inverse function to compute Sigma2.beta, the population variance-covariance matrix for the regression coefficients. This matrix is then used to compute r, the correlation between the population mean intercept and slope parameters.

We note that in the original rats analysis we not only assumed β_{1i}, β_{2i} were *a priori* independent, but also centred the covariates around their mean, which ensured that the likelihood for each β_{1i}, β_{2i} pair factorised. By not centering the covariates and using a multivariate normal prior for β_{1i}, β_{2i}, we have therefore introduced two additional forms of dependency.

We can investigate the influence of allowing such prior and likelihood dependence by fitting the range of models listed below:

- Likelihood for β_{1i}, β_{2i} independent (centred covariates)
  - Prior independence of β_{1i}, β_{2i} assumed: this is the original rats analysis, and implies that μ_{β_1} and μ_{β_2} retain independence even conditional on the data.
  - Prior dependence of β_{1i}, β_{2i} allowed: this leads to dependence in the posterior of μ_{β_1} and μ_{β_2} to be introduced, essentially due to the estimates of β_{1i}'s and β_{2i}'s being correlated (in fact, fitting completely independent growth curves to the 30 rats leads to an empirical correlation of .50 between the estimated slopes and estimated intercepts, even though each specific pair of estimates $\hat{\beta}_{1i}, \hat{\beta}_{2i}$ are independent).

- Likelihood for β_{1i}, β_{2i} dependent (uncentred covariates)
  - Prior independence of β_{1i}, β_{2i} assumed: in this formulation dependence between β_{1i}, β_{2i} is only introduced through the likelihood: when learning about μ_{β_1}, say, only current values of β_{1i}'s are taken into account.
  - Prior dependence of β_{1i}, β_{2i} allowed: this is the full multivariate analysis described above.

The BUGS code for all these combinations is given below: relevant models may be fitted by changing the commented lines appropriately. We note that the definition of β_{1i}, μ_{β_1} and α_0 depends on whether the likelihood is independent (covariates centred) or not.

Birats: **model specification in** BUGS

```
model birats;

const
   N = 30,  # number of rats
   T = 5;   # number of time points

var
   x[T],mu[N,T],Y[N,T],beta[N,2],mu.beta[2],Omega.beta[2,2],
   Sigma2.beta[2,2],sigma.beta[2],tau.c,sigma,R[2,2],r,alpha0,
   tau.beta[2];

data Y in "biratsy.dat", x in "biratsx.dat";
inits in "birats.in";

{
    for (i in 1:N) {
      for (j in 1:T) {
        Y[i,j] ~ dnorm(mu[i,j],tau.c);  #
        mu[i,j] <- beta[i,1] + beta[i,2]*x[j];             # uncentred
#       mu[i,j] <- beta[i,1] + beta[i,2]*(x[j]-mean(x[])); # centred
     }
     beta[i,] ~ dmnorm(mu.beta[],Omega.beta[,]);  # bivariate Normal
#    beta[i,1] ~ dnorm(mu.beta[1],tau.beta[1]);  # independent Normals
#    beta[i,2] ~ dnorm(mu.beta[2],tau.beta[2]);  # independent Normals
     }

# intercept at zero for centred model
#  alpha0 <- mu.beta[1] - mu.beta[2]* mean(x[]);

# intercept at mean(x[]) for uncentred model
   alpha0 <- mu.beta[1] + mu.beta[2]* mean(x[]);
# prior for sampling precision
  tau.c  ~ dgamma(1.0E-3,1.0E-3);  sigma <- 1.0/sqrt(tau.c);
```

```
# parameters considered MVN

 Omega.beta[,]  ~ dwish(R[,],2);    # Wishart prior on precision matrix
  R[1,1] <- 200.0; R[1,2] <- 0.0;   # R = prior guess at order of covariance
  R[2,1] <- 0.0; R[2,2] <- 0.2;     # matrix for beta[i,]
  Sigma2.beta[,] <-inverse(Omega.beta[,]);
  sigma.beta[1]<-sqrt(Sigma2.beta[1,1]);
  sigma.beta[2]<-sqrt(Sigma2.beta[2,2]);
  r <- Sigma2.beta[1,2] / (sqrt(Sigma2.beta[1,1])
                          *sqrt(Sigma2.beta[2,2]));  # correlation

# parameters considered independent
# for (k in 1:2){
#  tau.beta[k]  ~ dgamma(1.0E-3,1.0E-3);
#  sigma.beta[k]<-1/sqrt(tau.beta[k]);
# }

  mu.beta[1] ~ dnorm(0,.00001);   # 'flat' univariate Normal prior on mean
  mu.beta[2] ~ dnorm(0,.00001);   # 'flat' univariate Normal prior on mean
}
```

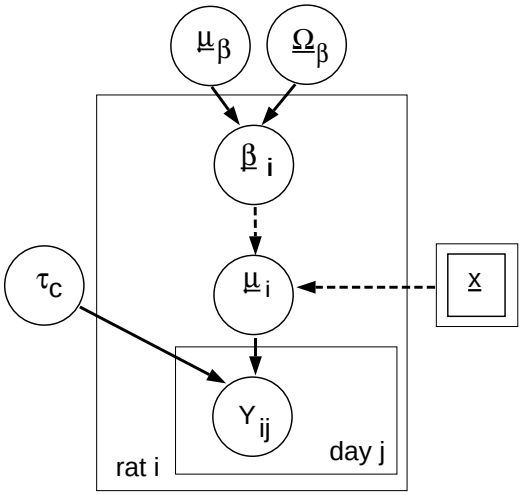

Figure 8: Graphical model for the `birats` example

Analysis

Each run used a 500 iteration burn-in and a further 5000 iterations, which took around 90 seconds and yielded the following results.

Variable		**Posterior mean &** *s.d.*			
	Likelihood for β_{1i}, β_{2i}:	independent		dependent	
	Prior for β_{1i}, β_{2i}:	indep.	dep.	indep.	dep.
Population intercept at $\overline{x}$					
μ_{β_1} (centred) or α_0 (uncentred)		242.4	242.7	242.6	242.6
		27.5	*27.3*	*28.6*	*27.7*
Population slope	μ_{β_2}	6.18	6.19	6.18	6.19
		.10	*.10*	*.10*	*.10*
Population s.d. in intercepts	$\Sigma_{\beta_{11}}$	14.7	14.6	10.4	10.7
		2.1	*2.0*	*1.9*	*2.0*
Population s.d. in slopes	$\Sigma_{\beta_{22}}$	.51	.51	.50	.51
		.09	*.09*	*.09*	*.09*
Correlation between	r	–	.64	–	-.07
slopes & intercepts)			*.13*		*.24*
Population intercept at 0:		106.3	106.4	106.5	106.5
α_0 (centred) or μ_{β_1} (uncentred)		*3.6*	*2.4*	*2.3*	*2.3*

We can make a number of observations from these results.

1. The estimated correlation r between slopes and intercepts is high for the centred data and almost zero for the uncentred, which suggests a 'fan' pattern in which the speed of growth is unrelated to initial weight at birth, but as the lines separate the intercept at $\overline{x}$ is higher for those with faster growth.

2. Independence assumptions make no difference to the main location parameter estimates, only their precision (and convergence properties).

3. For centred data, the population intercept at 0 is substantially less precise with the prior independence model than allowing dependence, since it is a function of μ_{β_1} and μ_{β_2} and the dependence introduced between them from the highly correlated β_{1i}, β_{2i}'s has been ignored.

4. For uncentred data, the population intercept at $\overline{x}$ is only slightly less precise with the prior independence model than allowing dependence, since it is a function of μ_{β_1} and μ_{β_2} and the very small dependence introduced between them from the correlated β_{1i}, β_{2i}'s has been ignored.

In general we would advise using the multivariate normal model for multiple random effects, particularly when interest lies in functions of the population coefficients, such as predictions.

It is also technically possible to assume a multivariate sampling model for the y_i's, in which correlated residuals around the growth curve are permitted. However, we have found these models have very poor convergence properties: initial poor estimates lead to the variance-covariance matrix of the observation vector having very large entries, and once this has occurred the sampler tends to 'stick' with these values since the fitted curves have little incentive to approach the data. If such models are attempted, extremely good starting values would be necessary, perhaps derived from the type of analysis above.

9 Schools: ranking school examination results using multivariate hierarchical models

Goldstein *et al.* (1993) present an analysis of examination results from inner London schools. They use hierarchical or multilevel models to study the between-school variation, and calculate school-level residuals in an attempt to differentiate between 'good' and 'bad' schools. Here we analyse a subset of this data and show how to calculate a rank ordering of schools and obtain credible intervals on each rank.

Data

Standardized mean examination scores (Y) were available for 1978 pupils from 38 different schools. The median number of pupils per school was 48, with a range of 1–198. Pupil-level covariates included gender plus a standardized London Reading Test (LRT) score and a verbal reasoning (VR) test category (1, 2 or 3, where 1 represents the highest ability group) measured when each child was aged 11. Each school was classified by gender intake (all girls, all boys or mixed) and denomination (Church of England, Roman Catholic, State school or other); these were used as categorical school-level covariates.

Model

We consider the following model, which essentially corresponds to Goldstein *et al.*'s model 1.

$$\begin{aligned} Y_{ij} &\sim \text{Normal}(\mu_{ij}, \tau_{ij}) \\ \mu_{ij} &= \alpha_{1j} + \alpha_{2j}\text{LRT}_{ij} + \alpha_{3j}\text{VR1}_{ij} + \beta_1\text{LRT}_{ij}^2 + \beta_2\text{VR2}_{ij} + \beta_3\text{Girl}_{ij} \\ &\quad +\beta_4\text{Girls' school}_j + \beta_5\text{Boys' school}_j + \beta_6\text{CE school}_j \\ &\quad +\beta_7\text{RC school}_j + \beta_8\text{other school}_j \\ \log\tau_{ij} &= \theta + \phi\text{LRT}_{ij} \end{aligned}$$

where i refers to pupil and j indexes school. We wish to specify a regression model for the variance components, and here we model the logarithm of τ_{ij} (the inverse of the between-pupil variance) as a linear function of each pupil's LRT score. This differs from Goldstein *et al.*'s model which allows the *variance* σ_{ij}^2 to depend linearly on LRT. However, such a parameterization may lead to negative estimates of σ_{ij}^2.

Prior distributions

The fixed effects β_k $(k = 1, ..., 8)$, θ and ϕ were assumed to follow vague independent Normal distributions with zero mean and low precision = 0.0001. The random school-level coefficients α_{jk} $(k = 1, 2, 3)$ were assumed to arise from a multivariate normal population distribution with unknown mean $\underline{\gamma}$ and covariance matrix Σ. A non-informative multivariate normal prior was then specified for the population mean $\underline{\gamma}$, whilst the inverse covariance matrix $T = \Sigma^{-1}$ was assumed to follow a Wishart distribution. To represent vague prior knowledge, we chose the the degrees of freedom for this distribution to be as small as possible (i.e. 3, the rank of T). The scale matrix R was specified as $\begin{pmatrix} 0.1 & 0.005 & 0.005 \\ 0.005 & 0.01 & 0.005 \\ 0.005 & 0.005 & 0.01 \end{pmatrix}$, which represents our prior guess at the order of magnitude of Σ.

Estimating the ranks

The school-specific intercept α_{j1} measures the 'residual effect' for school j after adjusting for pupil- and school-level covariates. This might represent an appropriate quantity by which to rank schools' performance. We compute the ranks in `BUGS` using the `step()` function where `step(x)` $= 1$ if `x` $\geq$ 0 and 0 otherwise. The jth row of the array `greater.than[]` (see `BUGS` code below) thus contains a 1 in columns corresponding to schools with an equal or higher intercept than school j, and zeros elsewhere. Summing this row yields the total number of schools who perform 'better' than school j, and thus corresponds to that school's rank. Since `rank[]` is a function of the stochastic node `alpha[,1]`, its value will change at every iteration. Hence we may obtain a posterior distribution for `rank[]` which may be summarized to give a point estimate and 95% credible interval for the rank of each school.

`BUGS` code for the `schools` example

A graphical representation of the model is shown in Figure 9 and the essentials of the `BUGS` code are given below. Note that the data are entered as a rectangular array with 1978 rows indexed by pupil. Column 2 of the data array is a school indicator taking value 1 for all pupils in school 1, 2 for all pupils in school 2 and so on. For computational convenience, `Y`, `mu` and `tau` are indexed over a single dimension `p` = 1,...,1978 rather than as pupil i within school j as used in equations on the previous page and in the graphical model. The appropriate school-level coefficients for pupil `p` are then selected using the school indicator in row `p` of the data array — for example `alpha[school[p],1]`.

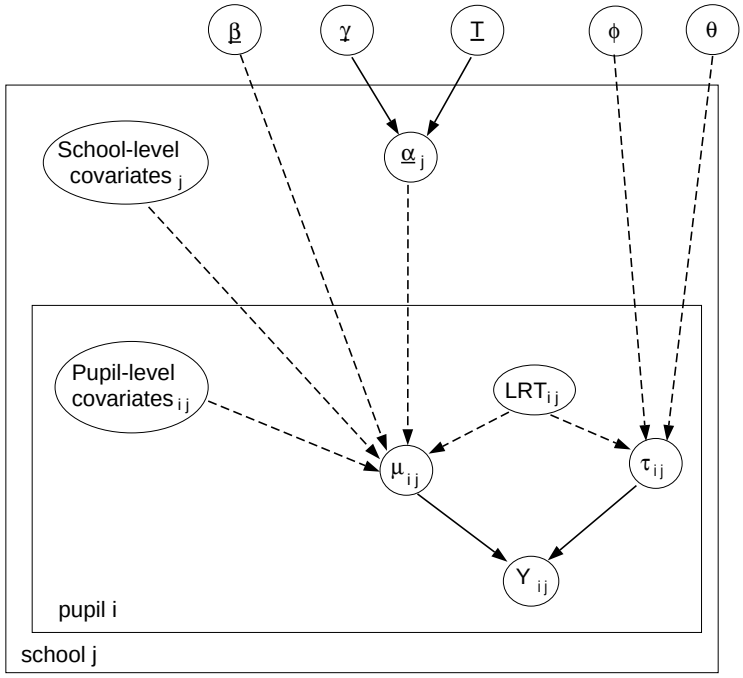

Figure 9: Graphical model for `schools` example

```
model schools;
const
   N = 1978,# number of pupils
   M = 38;      # number of schools

var
   Y[N], pupil[N], school[N], LRT[N], LRT2[N], VR[N,2], Gender[N],
   School.gender[N,2], School.denom[N,3], alpha[M,3], beta[8], mu[N],
   tau[N], sigma2[N], theta, phi, min.var, max.var, gamma[3], T[3,3],
   Sigma[3,3], mn[3], prec[3,3], R[3,3], rank[M], greater.than[M,M];

data school, pupil, LRT, School.gender, School.denom, Gender,
   VR, Y in "schools.dat";
inits in "schools.in";

{
   for(p in 1:N) {
      Y[p] ~ dnorm(mu[p], tau[p]);
      mu[p] <- alpha[school[p],1] + alpha[school[p],2]*LRT[p]
               + alpha[school[p],3]*VR[p,1] + beta[1]*LRT2[p]
               + beta[2]*VR[p,2] + beta[3]*Gender[p]
               + beta[4]*School.gender[p,1] + beta[5]*School.gender[p,2]
               + beta[6]*School.denom[p,1] + beta[7]*School.denom[p,2]
               + beta[8]*School.denom[p,3];
      log(tau[p]) <- theta + phi*LRT[p];
      sigma2[p] <- 1/tau[p];
      LRT2[p] <- pow(LRT[p],2);
   }
   min.var <- exp(-(theta + phi * (-34.6193))); # lowest LRT score = -34.6193
   max.var <- exp(-(theta + phi * (37.3807)));  # highest LRT score = 37.3807

   # Priors for fixed effects:
   for (k in 1:8) {  beta[k] ~ dnorm(0.0, 0.0001);   }
   theta ~ dnorm(0.0, 0.0001); phi ~ dnorm(0.0, 0.0001);

   # Priors for random coefficients:
   for (j in 1:M) {
      alpha[j,] ~ dmnorm(gamma[], T[,]);
   }

   # Hyper-priors:
   gamma[] ~ dmnorm(mn[], prec[,]);
   # Vague prior mean and precision for gamma
   for(k in 1:3) {
      mn[k] <- 0.0; prec[k,k] <- 0.0001;
      for(l in (k+1):3) { prec[l,k] <- 0.0; prec[k,l] <- 0.0; }
   }
   T[,] ~ dwish(R[,],3);
```

```
    # Prior guess at order of magnitude of covariance matrix for gamma
    R[1,1] <- 0.1; R[2,2] <- 0.01; R[3,3] <- 0.01;
    for(k in 1:3) {
       for(l in (k+1):3) {  R[k,l] <- 0.005; R[l,k] <- 0.005; }
    }
    Sigma[,] <- inverse(T[,]);

    # Compute ranks:
    for (j in 1:M) {
       for (k in 1:M) {
          greater.than[j,k] <- step(alpha[k,1] - alpha[j,1]);
       }
       rank[j] <- sum(greater.than[j,]);  # rank of school j
    }
}
```

Results

A 5000 iteration burn-in + 5000 further iterations took approximately 4 hours to complete. The posterior mean and standard error for each regression coefficient and the between-schools covariance matrix are shown in the table on the next page. Maximum likelihood estimates obtained using the ML3 software (Prosser *et al.*, 1991*a*) are also shown for comparison. Figure 10a shows the posterior mean and 95% credible intervals for the 'school effect' α_{j1}, and Figure 10b shows the corresponding posterior mean and 95% credible intervals for each school's rank.

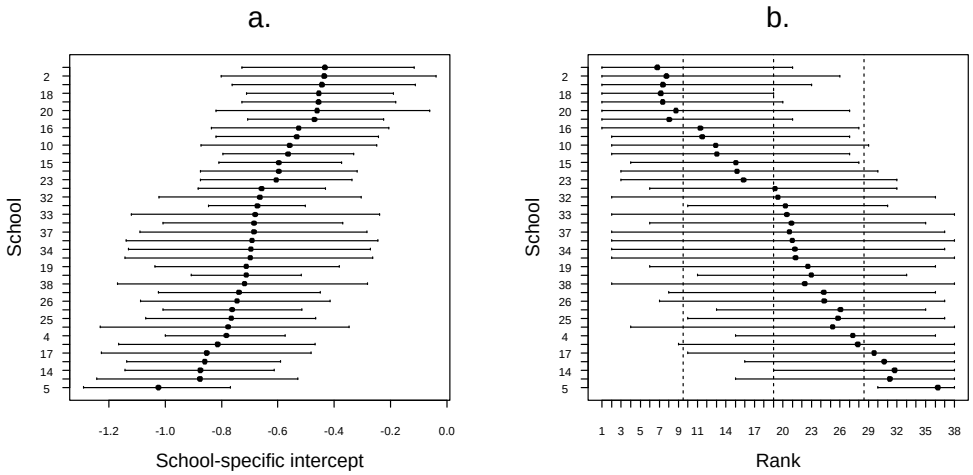

Figure 10: BUGS posterior means (•) and 95% credible intervals (⟝⟞) for (a) the school effect α_{j1} and (b) the rank for each school

BUGS posterior means and standard errors (SE), plus ML3 maximum likelihood estimates (MLE) for the `schools` example

Variable	BUGS **mean (SE)**	ML3 **MLE (SE)**
γ_1 (Population intercept)	−0.673 (0.090)	−0.703 (0.086)
γ_2 (LRT)	0.031 (0.005)	0.031 (0.002)
γ_3 (VR1)	0.962 (0.069)	0.939 (0.075)
β_1 (LRT^2)	0.00027 (0.00009)	0.00023 (0.00009)
β_2 (VR2)	0.419 (0.055)	0.438 (0.061)
β_3 (Girl)	0.168 (0.047)	0.173 (0.047)
β_4 (Girls' school)	0.123 (0.132)	0.139 (0.112)
β_5 (Boys' school)	0.072 (0.102)	0.086 (0.088)
β_6 (CE school)	−0.280 (0.184)	−0.220 (0.152)
β_7 (RC school)	0.145 (0.101)	0.143 (0.091)
β_8 (Other school)	−0.171 (0.175)	−0.135 (0.147)
Σ_{11} (Variance of $\underline{\alpha}_1$)	0.048 (0.017)	0.031 (0.011)
Σ_{22} (Variance of $\underline{\alpha}_2$)	0.0005 (0.0001)	0.00001 (0.00002)
Σ_{33} (Variance of $\underline{\alpha}_3$)	0.006 (0.005)	0.005 (0.014)
Σ_{12} (Covariance of $\underline{\alpha}_1$ & $\underline{\alpha}_2$)	0.0006 (0.001)	0.0001 (0.0004)
Σ_{13} (Covariance of $\underline{\alpha}_1$ & $\underline{\alpha}_3$)	0.002 (0.008)	0.004 (0.010)
Σ_{23} (Covariance of $\underline{\alpha}_2$ & $\underline{\alpha}_3$)	0.0002 (0.0004)	0.0008 (0.0004)
θ (log precision for pupils with average LRT)	0.580 (0.033)	– –
ϕ (change in log precision per unit increase in LRT)	-0.0026 (0.0028)	– –

The results are very similar to those obtained using ML3. The interval estimates illustrate the large degree of uncertainty associated with 'league tables'; there is only one school (number 5) whose 95% interval excludes the median rank. We also note that mean rank order is not identical to the rank order of the mean intercepts.

10 Ice: non-parametric smoothing in an age-cohort model

Breslow and Clayton (1993) analyse breast cancer rates in Iceland by year of birth ($K = 11$ cohorts from 1840-1849 to 1940-1949) and by age ($J = 13$ groups from 20-24 to 80-84 years). Due to the number of empty cells we consider a single indexing over $I = 77$ observed number of cases, giving data of the following form.

i	age_i	year_i	cases_i	person-years_i
1	1	6	2	41380
2	1	7	0	43650
.....				
77	13	5	31	13600

In order to pull in the extreme risks associated with small birth cohorts, Breslow and Clayton (1993) first consider the exchangeable model

$$\begin{aligned} \text{cases}_i &\sim \text{Poisson}(\mu_i) \\ \log \mu_i &= \log \text{person-years}_i + \alpha_{\text{age}_i} + \beta_{\text{year}_i} \\ \beta_k &\sim \text{Normal}(0, \tau). \end{aligned}$$

Running this model in BUGS gave an estimated σ of $.76 \pm .23$ compared to the estimate in Breslow and Clayton (1993) of $.69 \pm .17$.

10.1 Autoregressive smoothing of relative risks

They then consider the alternative approach of smoothing the rates for the cohorts by assuming an auto-regressive model on the β's, assuming the second differences are independent normal variates. This is equivalent to a model and prior distribution

$$\begin{aligned} \text{cases}_i &\sim \text{Poisson}(\mu_i) \\ \log \mu_i &= \log \text{person-years}_i + \alpha_{\text{age}_i} + \beta_{\text{year}_i} \\ \beta_1 &\sim \text{Normal}(0, 0.000001 \times \tau) \\ \beta_2 | \beta_1 &\sim \text{Normal}(0, 0.000001 \times \tau) \\ \beta_k | \beta_{1,\ldots,k-1} &\sim \text{Normal}(2\beta_{k-1} - \beta_{k-2}, \tau) \;\; k > 2. \end{aligned}$$

We note that β_1 and β_2 are given "non-informative" priors, but retain a τ term in order to provide the appropriate likelihood for τ.

For computational reasons Breslow and Clayton (1993) impose constraints on their random effects β_i in order that their mean and linear trend are zero, and counter these constraints by introducing a linear term $b \times \text{year}_i$ and allowing unrestrained estimation of α_j. Since we allow free movement of the β's we dispense with the linear term, and impose a "corner" constraint $\alpha_1 = 0$. The graph is shown in Figure 11.

Model specification for auto-regressive smoothing (`iceAR.bug`)

```
model iceAR;
const
   I = 77, Nage=13, K=11;
var
   age[I],  year[I],   cases[I], pyr[I], mu[I], alpha[Nage],
   beta[K], betamean[K],  betaprec[K], logRR[K], tau, sigma;
data age, year, cases, pyr in "ice.dat";
inits in "ice.in";
{
   for (i in 1:I) {
      cases[i]        ~ dpois(mu[i]);
      log(mu[i])     <- log(pyr[i]) + alpha[age[i]] + beta[year[i]]
   }

   betamean[1]    <- 0.0;
   betaprec[1]    <- tau*1.0E-6;
   betamean[2]    <- 0.0;
   betaprec[2]    <- tau*1.0E-6;
   for (k in 3:K){
      betamean[k]    <- 2*beta[k-1] - beta[k-2];
      betaprec[k]    <- tau
   }

   for (k in 1:K){
      beta[k]        ~ dnorm(betamean[k],betaprec[k]);
      logRR[k]      <- beta[k] - beta[5]
   }

   alpha[1]      <- 0.0;
   for (j in 2:Nage){
      alpha[j]       ~ dnorm(0,1.0E-6)
   }
   tau            ~ dgamma(1.0E-3,1.0E-3);
   sigma         <- 1/sqrt(tau);
}
```

We note that log(RR) are calculated relative to the 5th cohort, as in Breslow and Clayton (1993).

Initial data file

Imposing the constraint $\alpha_1 = 0$ defines α_1 to be a deterministic node. Consequently, we must ensure that α_1 is *not* given an initial value. This is achieved by inserting an `NA` in the appropriate position of the vector `alpha` in the inital values file as follows:

```
list(tau=1, alpha=c(NA,0,0,0,0,0,0,0,0,0,0,0,0), beta=c(0,0,0,0,0,0,0,0,0,0,0))
```

Analysis

The results for a run of 1000 iterations (20 seconds) are shown later.

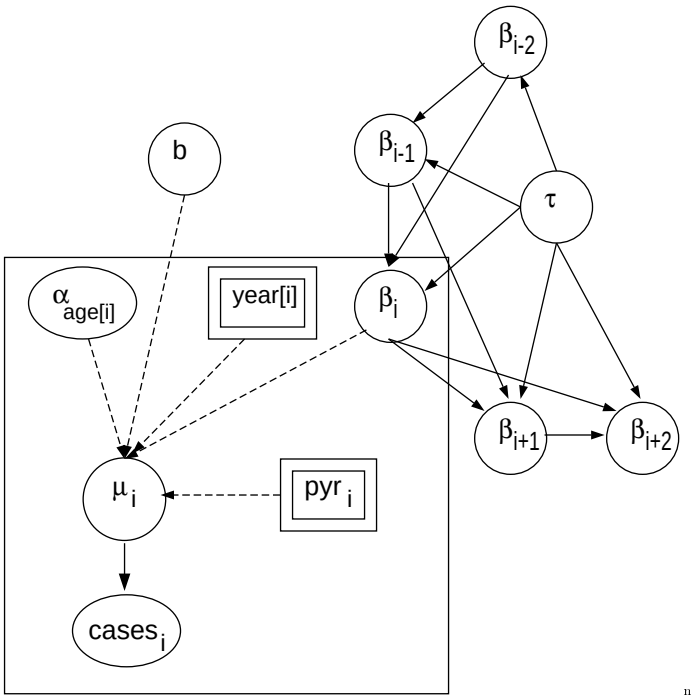

Figure 11: Graphical model for `ice` example, using the directed autoregressive representation

10.2 An undirected model using an intrinsic prior for the random effects

Breslow and Clayton (1993) point out that the joint prior for the β's defined by the autoregressive process can alternatively be represented in an undirected form, giving the model

$$
\begin{aligned}
\text{cases}_i &\sim \text{Poisson}(\mu_i) \\
\log \mu_i &= \log \text{person-years}_i + \alpha_{\text{age}_i} + \sigma\beta_{\text{year}_i} \\
\beta_i|\beta_j, j \neq i &\sim \text{Normal}(\overline{\beta_i}, n_i) \\
\text{where} & \\
\overline{\beta}_1 &= 2\beta_2 - \beta_3 \\
\overline{\beta}_2 &= 2\beta_1 + 4\beta_3 - \beta_4 \\
\overline{\beta}_k &= 4\beta_{k-1} + 4\beta_{k+1} - \beta_{k-2} - \beta_{k+2}, \quad 2 < k < K-1 \\
\overline{\beta}_{K-1} &= 2\beta_K + 4\beta_{K-2} - \beta_{K-3} \\
\overline{\beta}_K &= 2\beta_{K-1} - \beta_{K-2} \\
\text{and} & \\
n_1, n_K &= 1 \\
n_2, n_{K-1} &= 5 \\
n_k, \;\; k \neq 1, 2, K-1, K &= 6
\end{aligned}
$$

We note the use of σ as a multiplier for the random effects (see Section 9.5 of the manual and the `seeds` example): as mentioned previously a log-concave prior must be chosen for σ and we use $p(\sigma) = e^{-\sigma}$. Figure 12 shows the graph of this undirected smoothing model, and the associated `BUGS` code is shown below.

Model specification for undirected smoothing model in `iceCARsg.bug`

```
model iceCAR;
const
   I = 77, Nage=13, K=11;
var
   age[I],  year[I],   cases[I], pyr[I], mu[I], alpha[Nage],
   beta[K], betamean[K],  betaprec[K], logRR[K],   sigma;
data age, year, cases, pyr in "ice.dat";
inits in "ice.in";
{
   for (i in 1:I) {
      cases[i]        ~ dpois(mu[i]);
      log(mu[i])     <- log(pyr[i]) + alpha[age[i]]
                                    + sigma * beta[year[i]]
   }

   betamean[1]    <- 2*beta[2] - beta[3];
   betaprec[1]    <- 1;
   betamean[2]    <- (2*beta[1] + 4*beta[3] - beta[4])/5;
   betaprec[2]    <- 5;
   for (k in 3:(K-2)) {
      betamean[k]    <- (4*beta[k-1] + 4*beta[k+1]
                          - beta[k-2] - beta[k+2])/6;
      betaprec[k]    <- 6
   }
   betamean[K-1] <- (2*beta[K] + 4*beta[K-2] - beta[K-3])/5;
   betaprec[K-1] <- 5 ;
   betamean[K]   <- 2*beta[K-1] - beta[K-2];
   betaprec[K]   <- 1;

   for (k in 1:K) {
      beta[k]         ~ dnorm(betamean[k],betaprec[k]);
      logRR[k]       <- beta[k] - beta[5]
   }
   alpha[1]       <- 0.0;
   for (j in 2:Nage) {
      alpha[j]        ~ dnorm(0,1.0E-6)
   }
   sigma             ~ dgamma(1.00001,1.0);
 }
```

This illustrates the capacity of BUGS to deal with undirected graphs, using the precedence rule discussed in Section 9.2 of the manual. We note that σ represents an deviation from 0, and yet there is no constraint to make the β's be distributed around 0. We might expect, therefore, a strong induced dependence between σ and the parameters measuring location, and that the convergence might suffer as a consequence. 1000 iterations took 20 seconds and the results are shown at the end of the example.

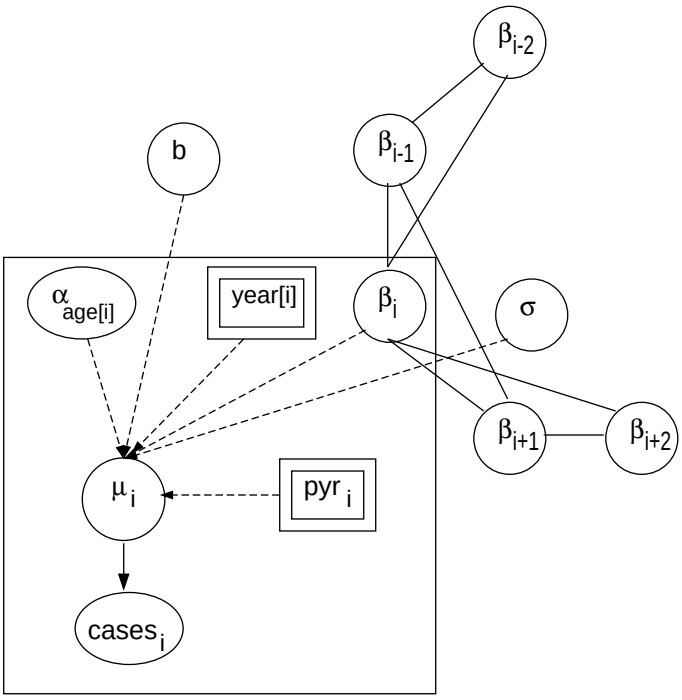

Figure 12: Graphical model for `ice` example, using the undirected representation (the intrinsic prior) for the random effects, and the variability of the β's acting directly within the linear predictor.

10.3 An intrinsic prior with a hyperparameter

An alternative, and perhaps more efficient, parameterisation is to consider the precision of the random effects as a hyperparameter. The essential changes to the previous model area as follows.

$$\begin{aligned} \log \mu_i &= \log \text{person-years}_i + \alpha_{\text{age}_i} + \beta_{\text{year}_i} \\ \beta_i | \beta_j, j \neq i &\sim \text{Normal}(\overline{\beta_i}, n_i \tau) \end{aligned}$$

The graph is shown in Figure 13.

As introduced in Section 9.5 of the manual, we need to be careful in deriving the full conditional sampling distribution for τ. If we were to leave the construction of this distribution to `BUGS`, a likelihood term would be included for each β_i: however, the likelihood for τ is **not** the product of these terms. Therefore we have to calculate algebraically the full conditional distribution for τ and put it in the `BUGS` model specification: as in the sampling distributions for the β's the precedence rule in `BUGS` then ensures that the β terms that now appear in the apparent prior for τ are not included as likelihood terms.

One can show that the conditional autoregressive model shown above is equivalent to the improper prior

$$p(b_1, ..., b_K | \tau) \propto \tau^{K/2} e^{-\frac{\tau}{2} \sum n_i b_i (b_i - \overline{b}_i)}$$

which provides the correct likelihood term for τ.

The essentials of the model specification (in `iceCAR.bug`) are shown below.

```
{
   for (i in 1:I) {
      cases[i]        ~ dpois(mu[i]);
      log(mu[i])     <- log(pyr[i]) + alpha[age[i]] + beta[year[i]]
   }
   for (k in 1:K){
      betaprec[k]    <- Nneighs[k] * tau;
   }
   d              <- 0.0001 + sum(tau.like[])/2;
   r              <- 0.0001 + K/2;
   tau             ~ dgamma(r,d);
 }
```

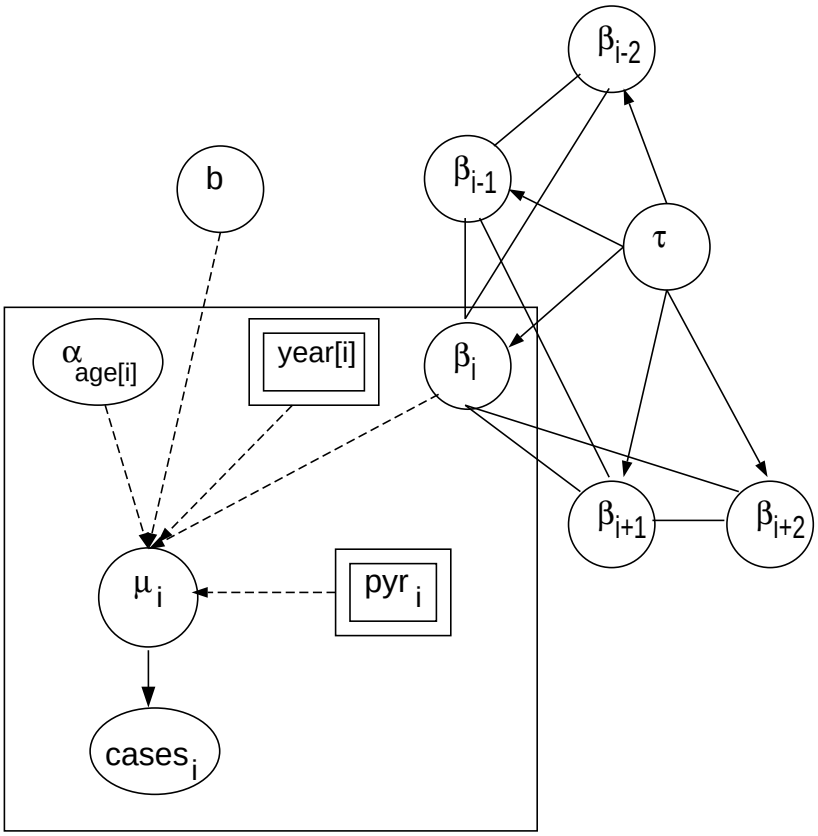

Figure 13: Graphical model for `ice` example, using the undirected representation (the intrinsic prior) for the random effects and a hyperparameter τ for their precision.

A simple BUGS run of 1000 iterations (following a 500 iteration burn-in) took 17 seconds. The results for all 3 models are shown below. We note essentially the same results coming from the autoregressive and the hyperparameter model, whereas the intrinsic prior model with σ acting in the linear predictor is substantially different. After a further 9000 iterations, this model still gave rather different results. We therefore do not recommend this approach, and suggest putting the full conditional distribution for the precision parameter. Breslow and Clayton (1993) estimated σ to be .12 (SD .06).

variable	Auto-regressive coeff ± SE	Intrinsic prior (σ) coeff ± SE	Intrinsic prior (τ) coeff ± SE
(σ)	.09 ± .05	1.59 ± 0.03	.06 ± .05
log RR[1]	-1.26 ± .22	-.89 ± .20	-1.16 ± .23
log RR[2]	-.89 ± .14	-.67 ± .12	-.83 ± .14
log RR[3]	-.52 ± .08	-.37 ± .08	-.50 ± .08
log RR[4]	-.20 ± .06	-.07 ± .06	-.21 ± .04
log RR[5]	0	0	0
log RR[6]	.13± .06	.04 ± .06	.16 ± .06
log RR[7]	.28± .07	.15 ± .06	.32 ± .09
log RR[8]	.43 ± .09	.29 ± .07	.49 ± .11
log RR[9]	.57 ± .12	.28 ± .09	.65 ± .15
log RR[10]	.77 ± .15	.55 ± .14	.83 ± .19
log RR[11]	.87 ± .24	.89 ± .29	1.02 ± .26

11 Lips: spatial smoothing of cancer rates

The rates of lip cancer in 56 counties in Scotland have been analysed by Clayton and Kaldor (1987) and Breslow and Clayton (1993). The form of the data includes the observed and expected cases (expected numbers based on the population and its age and sex distribution in the county), a covariate measuring the percentage of the population engaged in agriculture, fishing, or forestry, and the "position" of each county expressed as a list of adjacent counties.

County	Observed cases	Expected cases	x (% in agric..)	Observed SMR (100 O/E)	Adjacent counties
1	9	1.4	16	652.2	5,9,11,19
2	39	8.7	16	450.3	7,10
.....					
56	0	1.8	10	.0	18,24,30,33,45,55

We note that the extreme SMRs (Standardised Mortality Ratios) are based on very few cases. Breslow and Clayton (1993) initially consider a random-effects Poisson model allowing for over-dispersion, where O_i, E_i are the observed and expected cancer incidence in the ith county.

$$\begin{aligned} O_i &\sim \text{Poisson}(\mu_i) \\ \log \mu_i &= \log E_i + \alpha_1 x_i/10 + b_i \\ b_i &\sim \text{Normal}(\alpha_0, \tau) \\ \hat{SMR}_i &= 100\mu_i/E_i. \end{aligned}$$

α_0, α_1 and τ are given independent "noninformative" priors. We note that the prior distribution for the b's can be easily shown to be equivalent to a model with an "intrinsic" prior

$$b_i | b_j, j \neq i \sim \text{Normal}(\overline{b}_{\setminus i}, \frac{N-1}{N}\tau)$$

where N is the number of counties, and $\overline{b}_{\setminus i} = \frac{1}{N-1}\sum_{j \neq i} b_j$ is the average in all counties except i.

The graph is shown in Figure 14.

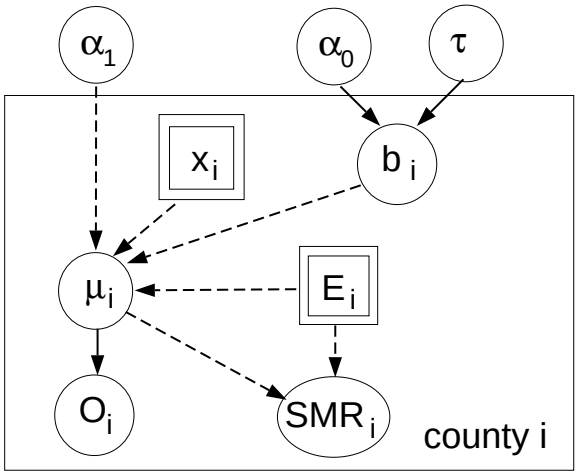

Figure 14: Graphical model for `lips` example, assuming exchangeable relative risks.

Model specification for exchangeable model (`lipsEX.bug`).

```
model lips;
const
   regions = 56, neighbours = 264;
var
   O[regions], b[regions], mu[regions],
   E[regions], x[regions], SMRhat[regions],
   alpha0, alpha1, tau, sigma;
data in "lips.dat";
inits in "lips.in";
{
   for (i in 1:regions) {
      O[i]        ~ dpois(mu[i]);
      log(mu[i]) <- log(E[i])  + alpha1* x[i]/10 +  b[i];
      b[i]        ~ dnorm(alpha0,tau);
      SMRhat[i]  <- 100*mu[i]/E[i];
   }
   alpha0   ~ dnorm(0.0,1.0E-5);
   alpha1   ~ dnorm(0.0,1.0E-5);
   tau      ~ dgamma(1.0E-3,1.0E-3);
   sigma   <- 1/sqrt(tau);
}
```

Results from this model, both with and without the covariate, are shown at the end of this example.

11.1 Spatial smoothing using an intrinsic prior

Breslow and Clayton (1993) consider a random-effects Poisson model allowing for over-dispersion and spatial correlation, using the conditional autoregressive (CAR) model of Besag (1974), which may be written

$$\begin{aligned}
O_i &\sim \text{Poisson}(\mu_i) \\
\log \mu_i &= \log E_i + \alpha_1 x_i/10 + \sigma b_i \\
b_i &\sim \text{Normal}(\overline{b}_i, n_i) \\
n_i &= \text{Number of neighbours of } i \\
\overline{b}_i &= \frac{1}{n_i} \sum_{j \epsilon \text{neighbours}(i)} b_j \\
\hat{SMR}_i &= 100\mu_i/E_i.
\end{aligned}$$

The graph for this model is shown in Figure 15. As with the exchangeable model, introducing the intrinsic prior means that a level term α_0 is not necessary in this model, although Breslow and Clayton (1993) retain this term due to their imposition of the constraint that $\sum_i b_i = 0$. As in the `seeds` and `ice` examples, the standard noninformative prior for σ cannot be used.

Specification of spatial model with intrinsic prior (`lipsSig.bug`).

```
model lipsSig;
const
   regions = 56, neighbours = 264;
var
   O[regions], b[regions], b.bar[regions], SMR[regions],
   mu[regions], E[regions], off[regions+1],Nneighs[regions],
   x[regions], SMRhat[regions],
   map[neighbours], b.neigh[neighbours], alpha1, sigma;
data in "lips.dat";
inits in "lips.in";
{
   for (i in 1:regions) {

      O[i]        ~ dpois(mu[i]);
      log(mu[i]) <- log(E[i]) + alpha1*x[i]/10 + sigma * b[i];
      b[i]        ~ dnorm(b.bar[i],Nneighs[i]);

      b.bar[i]   <- mean(b.neigh[ off[i]+1 : off[i+1] ]);
      SMRhat[i]  <- 100*mu[i]/E[i];
      Nneighs[i] <- off[i+1] - off[i];
    }

   for (i in 1:neighbours) {
      b.neigh[i] <- b[map[i]];
   }
   alpha1      ~ dnorm(0.0,1.0E-5);
   sigma       ~ dgamma(1.00001,1.0)
 }
```

This model shows how one can handle variable length attributes relating to each county. The data file needs to contain the "map" of adjacent counties. The relevant part of `lips.dat` is as follows:

```
      map = c( 5, 9,11,19,
               7,10,
               6,12,
               .....
               18,24,30,33,45,55),
     off = c(  0,  4,  6,  8, ... ,258,264))
```

This shows the neighbouring counties in a single long list `map`, with an additional list `off` of offset counts, indicating that the list of neighbours of county i starts at entry `off[i]+1` and ends at entry `off[i+1]` in `map`. This enables the calculation of the number of neighbours (`Nneighs`) within the `.bug` program, and also to identify the current value of b_j (`b.neigh`) for each neighbour of i. This in turn allows the calculation of the mean `b.bar[i]` of the neighbours of county i.

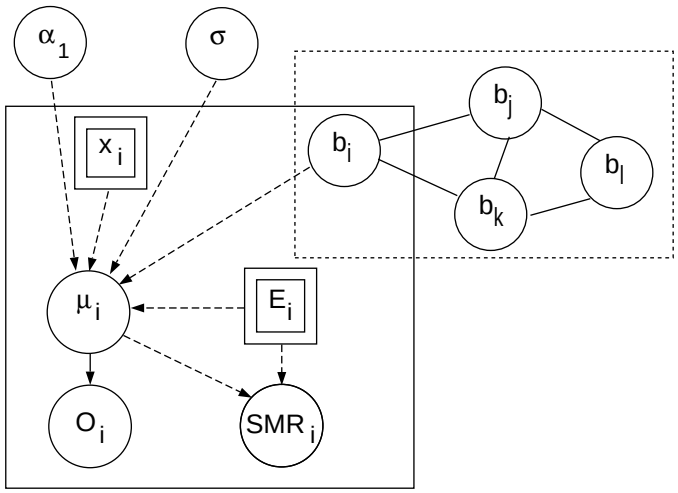

Figure 15: Graphical model for `lips` example, assuming spatial smoothing of relative risks using an intrinsic prior.

The precedence condition in `BUGS` ensures that the sampling for each b does not use the same variables in the prior and the likelihood. 1000 iterations for this model took 47 seconds, and the results are shown at the end of this example.

11.2 Spatial model with intrinsic prior and hyperparameter.

As in the `ice` example, we can introduce a precision parameter as a hyperparameter for the random effects, which will speed up the computation but requires the external calculation of the appropriate full conditional sampling distribution.

$$\begin{aligned}
O_i &\sim \text{Poisson}(\mu_i) \\
\log \mu_i &= \log E_i + \alpha_1 x_i/10 + b_i \\
b_i &\sim \text{Normal}(\overline{b}_i, \tau_i) \\
n_i &= \text{Number of neighbours of } i \\
\overline{b}_i &= \frac{1}{n_i} \sum_{\text{neighbours}(i)} b_i \\
\tau_i &= n_i \tau \\
\hat{SMR}_i &= 100\mu_i/E_i.
\end{aligned}$$

The graph of this model is shown in Figure 16. It can be shown that this is equivalent to the improper prior

$$p(b_1, ..., b_I|\tau) \propto \tau^{I/2} e^{-\frac{\tau}{2}\sum n_i b_i (b_i - \overline{b}_i)}$$

which provides the correct likelihood term for τ. Breslow and Clayton (1993) mention that this prior can also be expressed as

$$p(b_1, ..., b_I|\tau) \propto \tau^{I/2} e^{-\frac{\tau}{4}\sum_{i\sim j}(b_i - b_j)^2}$$

where $\sim$ here represents "is a neighbour of".

The likelihood for τ is derived above, and the term contributed by each county i (`tau.like[i]`) must be calculated. A proper prior $\Gamma(r^*, d^*) = \Gamma(1, 1)$ is assumed. The precedence condition within BUGS (see Section 9.5 of the Manual) then ensures terms are not included in both prior and likelihood when sampling.

Essentials of specification of spatial model with intrinsic prior and hyperparameter (`lipsCAR.bug`)

```
{
   for (i in 1:regions) {

      O[i]        ~ dpois(mu[i]);
      log(mu[i]) <- log(E[i]) + alpha1*x[i]/10 + b[i];
      b[i]        ~ dnorm(b.bar[i],tau.i[i]);

      b.bar[i]   <- mean(b.neigh[ off[i]+1 : off[i+1] ]);
      tau.i[i]   <- tau * Nneighs[i];
      SMRhat[i]  <- 100*mu[i]/E[i];
      Nneighs[i] <- off[i+1] - off[i];
      tau.like[i]<- Nneighs[i] * b[i] * (b[i]-b.bar[i]);
   }

   d          <- dstar + sum(tau.like[])/2;
   r          <- rstar + regions/2;
   tau         ~ dgamma(r,d);
   sigma      <- 1/sqrt(tau);
}
```

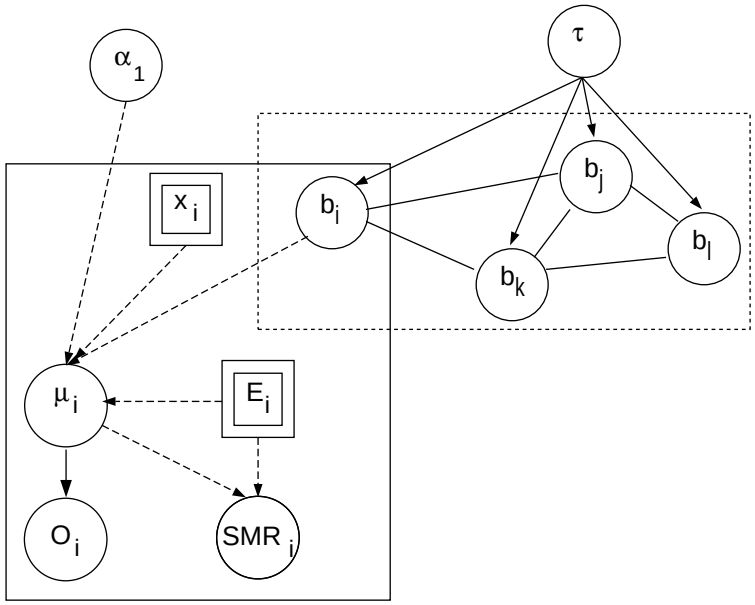

Figure 16: Graphical model for `lips` example, assuming spatial smoothing of relative risks governed by a hyperparameter τ.

Analysis

We may compare the penalized quasi-likelihood (PQL) (Breslow and Clayton, 1993) and `BUGS` results, using a burn-in of 500 iterations and estimation based on 1000 samples.

	constant (α_0)	$x/10$ (α_1)	σ
Exchangeable model			
PQL	-.44 ± .16	.68 ± .14	.60 ± .08
`BUGS`	-.52 ± .14	.71 ± .12	.61 ± .09
Spatial model			
PQL	-.18 ± .12	.35 ± .12	.73 ± .13
`BUGS` (using σ)	—	.37 ± .11	.69 ± .12
`BUGS` (using τ)	—	.36 ± .12	.76 ± .13

Unlike the previous `ice` example, the two parameterisations of the precision of the random effects arrive at the same answer, probably because in this example the random effects are distributed around 0.

Removing the covariate from the model provides the following estimates for the SMR's, which may be compared to the results of Breslow and Clayton (1993) using PQL. The `BUGS` results for the spatial model are using the hyperparameter τ.

Observed SMR	**Exchangeable model**		**Spatial model**	
(100 × O/E)	PQL	`BUGS`	PQL	`BUGS`
652.2	473.9	471.7 ± 166.0	446.3	477.0 ± 138.1
450.3	424.2	419.0 ± 65.4	438.3	432.4 ± 64.4
361.8	305.9	289.3 ± 83.6	352.1	326.7 ± 93.3
.....				
0.0	61.5	80.4 ± 42.6	68.3	74.8 ± 23.4

We note the substantial drop in the error in the low-risk areas through the spatial model.

12 Beetles: logistic, probit and extreme value (log-log) model comparison

Dobson (1983) analyses binary dose-response data published by Bliss (1935), in which the numbers of beetles killed after 5 hour exposure to carbon disulphide at N=8 different concentrations are recorded:

Concentration (x_i)	Number of beetles (n_i)	Number killed (r_i)
1.6907	59	6
1.7242	60	13
1.7552	62	18
1.7842	56	28
1.8113	63	52
1.8369	59	52
1.8610	62	61
1.8839	60	60

We assume that the observed number of deaths r_i at each concentration x_i is binomial with sample size n_i and true rate p_i. Plausible models for p_i include the logistic, probit and extreme value (complimentary log-log) models, as follows

$$\begin{aligned} p_i &= \frac{\exp(\alpha+\beta x_i)}{1+\exp(\alpha+\beta x_i)} \\ p_i &= \Phi(\alpha+\beta x_i) \\ p_i &= 1-\exp(-\exp(\alpha+\beta x_i)) \end{aligned}$$

The corresponding graph is shown in Figure 17.

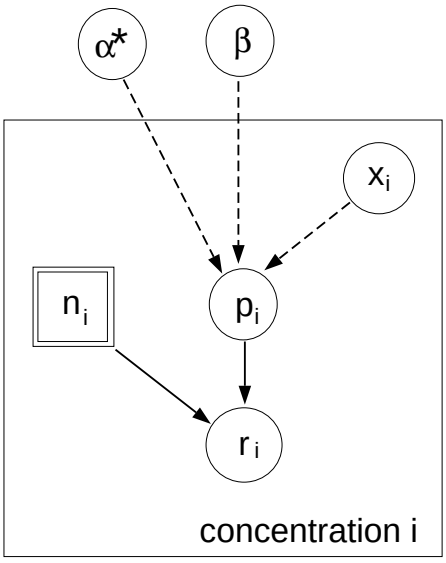

Figure 17: Graphical model for `beetles` example

The fit of each model may be assessed by calculating the deviance D as follows (see also the `seeds` example). The log-likelihood for an observation r_i arising from a binomial model with denominator n_i and success probability p_i is

$$\text{llike}_i \quad = \quad r_i \log(p_i) + (n_i - r_i) \log(1 - p_i)$$

The saturated log likelihood for the binomial model is

$$\text{llike.sat}_i \quad = \quad r_i \log(\frac{r_i}{n_i}) + (n_i - r_i) \log(1 - \frac{r_i}{n_i})$$

The deviance is calculated by computing a node $D = 2(\sum_i \text{llike.sat}_i - \sum_i \text{llike}_i)$ within `BUGS`. This will yield a posterior distribution for D, the *minimum* value of which corresponds to the classical deviance obtained using maximum likelihood estimation. This may be compared with a χ^2_{N-2} distribution to assess model fit.

```
model beetles;
const
 N = 8;    # number of doses
var
 r[N],p[N],x[N],n[N],alpha,alpha.star,beta,r.hat[N],llike[N],llike.sat[N],D;
data x, n, r in "beetles.dat";
inits in "beetles.in";
{
 for (i in 1:N) {
   r[i] ~ dbin(p[i], n[i]);
   logit(p[i])   <- alpha.star + beta*(x[i]-mean(x[]));
# alternative link functions:
#  probit(p[i])  <- alpha.star + beta*(x[i]-mean(x[]));
#  cloglog(p[i]) <- alpha.star + beta*(x[i]-mean(x[]));
# log likelihood for sample i & saturated log-likelihood:
   llike[i]     <- r[i]*log(p[i]) + (n[i]-r[i])*log(1-p[i]);
   llike.sat[i] <- r[i]*log(r[i]/n[i]) + (n[i]-r[i])*log(1-r[i]/n[i]);

   r.hat[i] <- p[i]*n[i];  # fitted values
 }
 alpha.star ~ dnorm(0.0, 1.0E-3);
 beta       ~ dnorm(0.0, 1.0E-3);
 alpha      <- alpha.star - beta*mean(x[]);

 D <- 2 * (sum(llike.sat[]) - sum(llike[]));
}
```

Note that we have standardized each dose x_i about the mean: this gives approximately uncorrelated regression coefficients, and greatly improves convergence. Figure 18 shows the sample traces (ploted using `CODA`) for `alpha` and `beta` after a 10000 iteration `BUGS` run. The first run (`chain:beetles1`) used the centered parameterization, whilst the second run (`chain:beetles2`) used the uncentered parameterization. The chains from the latter run have still not converged, and exhibit very high autocorrelations (see Figure 19), whilst those from the former run converge almost immediately, and exhibit rapid mixing rather than a slow, 'snaking' trace; this is reflected by the much lower autocorrelation.

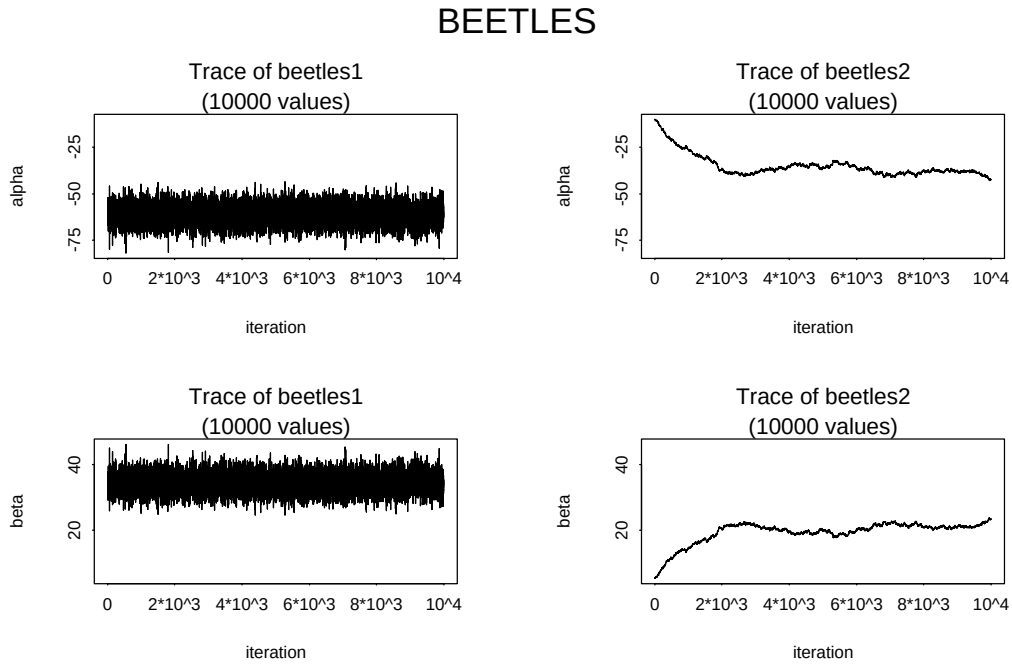

Figure 18: Traces for the models with centered covariates (`beetles1`) and uncentered covariates (`beetles2`)

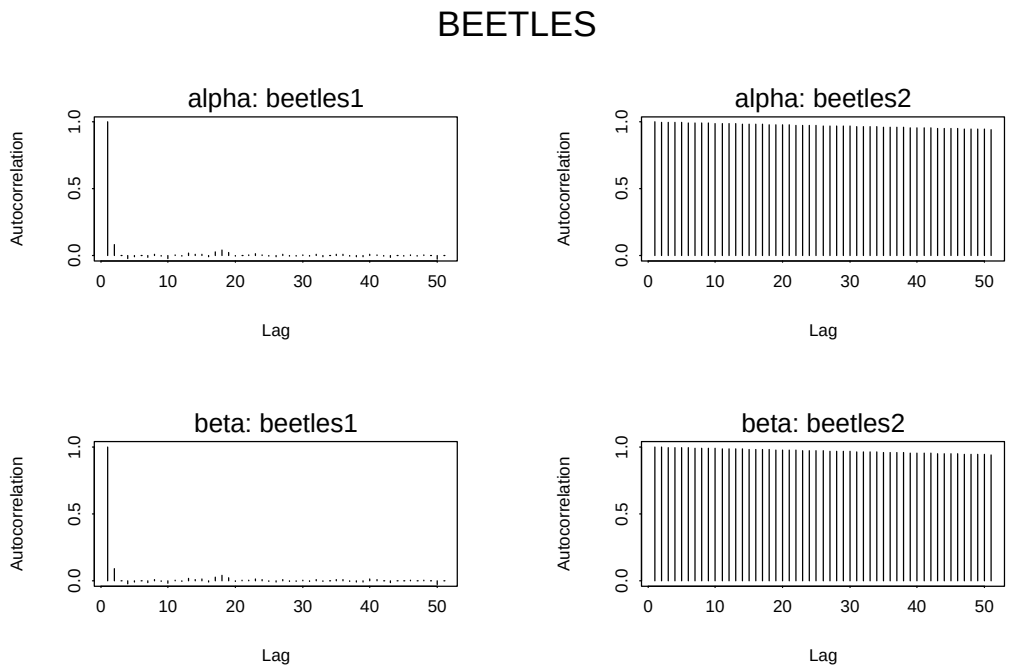

Figure 19: Plot of the within-chain autocorrelations for the models with centered covariates (`beetles1`) and uncentered covariates (`beetles2`)

Analysis

1000 iterations took 2 seconds after a 500 iteration burn-in. The `BUGS` posterior means and standard errors for the regression coefficients and fitted values, plus the minimum deviance for each model are given below, as are Dobson's maximum likelihood estimates (MLE). Note that the equivalent of MLE's may be obtained from the `BUGS` output by taking the values of `alpha`, `beta` and `r.hat` sampled at the iteration yielding the minimum value for the deviance. These are also given in the table below.

	Logistic			**Probit**			**Extreme value**		
	MLE	`BUGS` "MLE"	`BUGS` mean±SE	MLE	`BUGS` "MLE"	`BUGS` mean±SE	MLE	`BUGS` "MLE"	`BUGS` mean±SE
α	−60.72±5.18	−60.82	−60.76±5.13	–	−34.91	−35.09±2.58	–	−39.67	−39.89±3.24
β	34.27±2.91	34.33	34.30±2.88	–	19.71	19.82±1.45	–	22.10	22.22±1.80
$\hat{r_1}$	3.46	3.45	3.60±0.98	3.36	3.38	3.46±1.01	5.59	5.57	5.60±1.12
$\hat{r_2}$	9.84	9.84	1.00±1.72	10.72	10.77	10.81±1.71	11.28	11.27	11.25±1.60
$\hat{r_3}$	22.45	22.47	22.62±2.15	23.48	23.54	23.58±1.94	20.95	20.96	20.90±1.92
$\hat{r_4}$	33.90	33.93	34.00±1.78	33.82	33.86	33.93±1.61	30.37	30.41	30.35±1.69
$\hat{r_5}$	50.10	50.14	50.12±1.63	49.62	49.65	49.71±1.59	47.78	47.85	47.81±1.74
$\hat{r_6}$	53.29	53.32	53.25±1.10	53.32	53.33	53.34±1.11	54.14	54.20	54.14±1.21
$\hat{r_7}$	59.22	59.24	59.16±0.72	59.66	59.67	59.64±0.71	61.11	61.14	61.05±0.52
$\hat{r_8}$	58.74	58.75	58.69±0.42	59.23	59.23	59.19±0.34	59.95	59.95	59.92±0.09
D (min)	11.23	11.23	11.23	10.12	10.12	10.12	3.45	3.45	3.45

Comparison of the minimum deviances indicates that the extreme value model fits the data considerably better than do the logistic or probit models. This appears to be due to a smaller discrepancy between observed (r_i) and fitted ($\hat{r_i}$) values at the lower concentrations for the extreme value model.

13 Pines: Bayes factors for selecting regression models

General Formulation

Carlin and Chib (1995) consider the general problem of having K models with parameters $\theta_1, ..., \theta_K$, and wanting to obtain the posterior probability of each model. If the model indicator M is specified as a variable and hence as a node in the graph, M can then be sampled in a Gibbs run, and hence $\hat{p}(M = j|y)$ is obtained as a frequency of $M = j$ in the sample. However, we need to specify a full probability model in order to satisfy MCMC conditions for convergence.

Their approach is to make the following assumptions:

- y is independent of $\theta_{k\neq j}$ given that $M = j$; *i.e.* M picks which parameters are relevant to y.
- $\theta_1, ..., \theta_K$ are independent given the model indicator M.

These imply an overall joint distribution

$$\begin{aligned} p(y, \underline{\theta}, M = j) &= p(y|\underline{\theta}, M = j)\, p(\underline{\theta}|M = j)\, p(M = j) \\ &= p(y|\theta_j, M = j) \times \prod_k p(\theta_k|M = j)\, p(M = j) \end{aligned}$$

When it comes to Gibbs sampling, the full conditional distributions are

$$\begin{aligned} p(M = j|\underline{\theta}, y) &\propto p(y, \underline{\theta}, M = j) \\ &= p(y|\theta_j, M = j) \times \\ &\quad \prod_k p(\theta_k|M = j)\, p(M = j) \\ p(\theta_j|\theta_{\neq j}, y, M = j) &\propto p(y|\theta_j, M = j)\, p(\theta_j|M = j) \\ p(\theta_j|\theta_{\neq j}, y, M = k) &\propto p(\theta_j|M \neq j) \end{aligned}$$

$p(\theta_{k=j}|M \neq j)$ are known as *pseudo-priors*, and although their form is theoretically arbitrary, it is convenient to have them close to $p(\theta_j|M = j, y)$ so that plausible values are generated even when the model is being assumed false.

Carlin and Chib recommend a two-stage approach to estimation and model choice:

- Run each model separately using 'estimation priors'.
- Use an approximation of the resulting posterior distributions as pseudo-priors for other models.
- Run sampler for all models together, monitoring M.
- Adjust the prior for M to ensure frequent visitation to all models.
- Re-adjust estimate of $p(M|y)$ to allow for the choice of prior on the model.

One of the examples of Carlin and Chib (1995) concerns data of Williams (1959) on 42 specimens of radiata pine. For each specimen the maximum compressive strength y_i was measured, with its density x_i and its density adjusted for resin content z_i. Part of the data is shown below.

Specimen	strength y_i	density x_i	adjusted z_i
1	3040	29.2	25.4
2	2470	24.7	22.2
3	3610	32.3	32.2
4	3480	31.3	31.0
....			
41	3030	33.2	29.4
42	3030	28.2	28.2

Two alternative models are being considered:

$$\begin{aligned} \text{Model 1: } y_i &\sim \text{Normal}(\alpha + \beta x_i, \tau_1) \\ \text{Model 2: } y_i &\sim \text{Normal}(\gamma + \delta z_i, \tau_2) \end{aligned}$$

The graph for the joint model is shown in Figure 20.

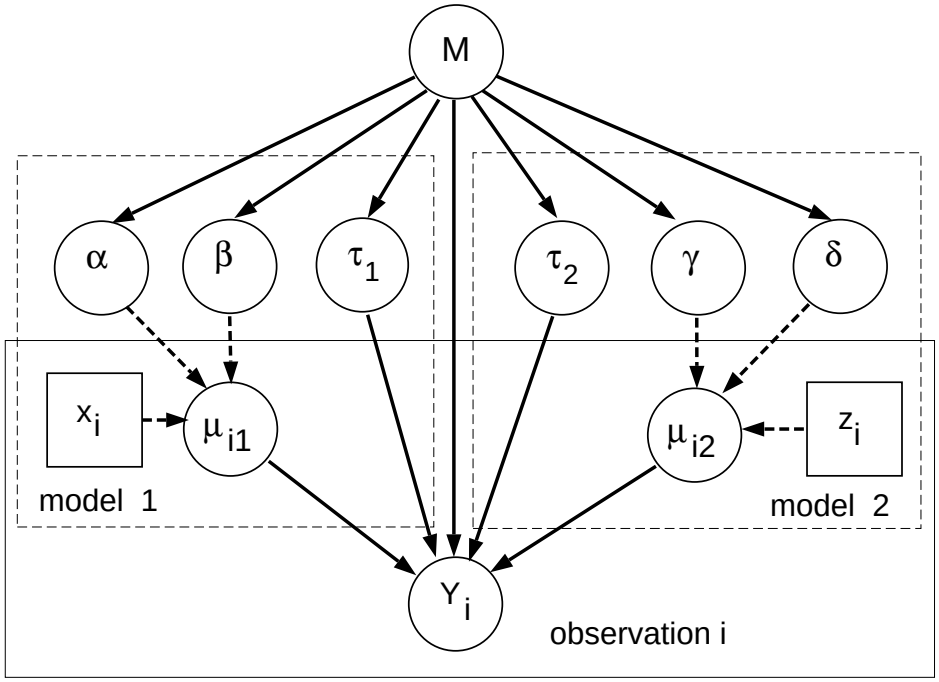

Figure 20: Graphical model for `pines` example showing the two models being simultaneously handled within a unified framework.

The following `BUGS` code shows that all variables were standardised to have mean 0 and variance 1 before analysis.

pines : model specification in BUGS

```
model pines;

const
   N = 42,  # number of data points
   M = 2;   # number of models
var
   Y[N],  Ys[N], # raw and standardised data
   x[N],  xs[N],
   z[N],  zs[N],
   mu[M,N],      # means for each model
   tau[M],       # precisions for each model
 alpha, mu.alpha[M], tau.alpha[M],  # priors for parameters
  beta, mu.beta[M] , tau.beta[M] ,
 gamma, mu.gamma[M], tau.gamma[M],
 delta, mu.delta[M], tau.delta[M],
 p[M],                             # prior for model
 pM2,                              # probability of model 2
  j,                               # true model
   r1[M], l1[M],                   # priors for tau[1]
   r2[M], l2[M];                   # priors for tau[2]

data in "pines.dat";
inits in "pines.in";

{
# standardise data
   for(i in 1:N){
       Ys[i] <- (Y[i] - mean(Y[]))/sd(Y[]);
       xs[i] <- (x[i] - mean(x[]))/sd(x[]);
       zs[i] <- (z[i] - mean(z[]))/sd(z[]);
    }

# model node
       j  ~ dcat(p[]);
    p[1] <- 0.9995; p[2] <- 0.0005; # use for joint modelling
#   p[1] <- 1; p[2] <- 0 ; # include for estimating Model 1
#   p[1] <- 0 ; p[2] <-1;  # include for estimating Model 2
     pM2 <- step(j - 1.5);

# model structure
   for(i in 1:N){
       mu[1,i] <- alpha + beta *xs[i];
       mu[2,i] <- gamma + delta*zs[i];
       Ys[i]    ~ dnorm(mu[j,i],tau[j]);
   }
```

```
# Model 1
   alpha  ~ dnorm(mu.alpha[j],tau.alpha[j]);
   beta   ~ dnorm(mu.beta[j],tau.beta[j]);
   tau[1] ~ dgamma(r1[j],l1[j]);
# estimation priors
   mu.alpha[1]<- 0; tau.alpha[1] <- 1.0E-6;
   mu.beta[1] <- 0; tau.beta[1]  <- 1.0E-4;
   r1[1]      <- 0.0001;   l1[1] <- 0.0001;
# pseudo-priors
   mu.gamma[1] <- 0;  tau.gamma[1] <- 400;
   mu.delta[1] <- 1;  tau.delta[1] <- 400;
   r2[1]       <- 46     ;   l2[1] <- 4.5;

# Model 2
   gamma  ~ dnorm(mu.gamma[j],tau.gamma[j]);
   delta  ~ dnorm(mu.delta[j],tau.delta[j]);
   tau[2] ~ dgamma(r2[j],l2[j]);
# estimation priors
   mu.gamma[2] <- 0; tau.gamma[2] <- 1.0E-6;
   mu.delta[2] <- 0; tau.delta[2] <- 1.0E-4;
   r2[2]       <- 0.0001;   l2[2] <- 0.0001
# pseudo-priors
   mu.alpha[2]<- 0; tau.alpha[2] <- 256;
   mu.beta[2] <- 1; tau.beta[2]  <- 256;
   r1[2]      <- 30    ;   l1[2] <- 4.5;
}
```

Running each of the models separately gave the following within-model parameter estimates (posterior means and standard deviations).

	Model 1 (x)	Model 2 (z)
intercept	-.0001 ± .06	-.0002± .05
gradient	.93 ± .06	.95 ± .05
$\tau = \sigma^{-2}$	6.8 ± 1.5	10.2± 2.2

Approximations to these results are then used as the pseudo-priors for the 'wrong' model shown in the BUGS code above: for Model 1 we set priors $\gamma \sim \text{Norm}(0, 400)$, $\delta \sim \text{Norm}(1, 400)$, $\tau \sim \text{Gamma}(46, 4.5)$, while under Model 2 we set priors $\alpha \sim \text{Norm}(0, 256)$, $\beta \sim \text{Norm}(1, 256)$,$\tau \sim \text{Gamma}(30, 4.5)$. The prior on the second model has to be adjusted to $p(M = 2) = .0005$ to ensure $M = 1$ is visited frequently.

A BUGS run of 500 burn-in and 10000 iterations took 1 minute and gave $\hat{p}(M = 2|y) = .629$. Hence the Bayes factor is $\frac{.629}{1-.629} \times \frac{.9995}{.0005} = 3389$, compared with Carlin and Chib's estimate of $\hat{p}(M = 2|y) = .689$ and their Bayes factor of 4420. The differences in these results could be due to the different estimation priors used in our analysis.

14 Alli: multinomial-logistic models

Agresti (1990) analyses a set of data on the feeding choice of 221 alligators, where the response measure for each alligator is one of 5 categories: fish, invertebrate, reptile, bird, other. Possible explanatory factors are the length of alligator (two categories: ≤ 2.3 metres and > 2.3 metres), and the lake (4 catgeories: Hancock, Oklawaha, Trafford, George). The full data is shown below.

		Primary Food Choice				
Lake	Size	Fish	Invertebrate	Reptile	Bird	Other
Hancock	≤ 2.3	23	4	2	2	8
	> 2.3	7	0	1	3	5
Oklawaha	≤ 2.3	5	11	1	0	3
	> 2.3	13	8	6	1	0
Trafford	≤ 2.3	5	11	2	1	5
	> 2.3	8	7	6	3	5
George	≤ 2.3	16	19	1	2	3
	> 2.3	17	1	0	1	3

Each combination of explanatory factors is assumed to give rise to a multinomial response with a logistic link, so that for lake i, size j, the observed vector of counts $X_{ij.} = X_{ij1}, ..., X_{ij5}$ has distribution

$$\begin{aligned} X_{ij.} &\sim \text{Multinomial}(p_{ij.}, n_{ij}) \\ p_{ijk} &= \phi_{ijk} / \sum_{k=1}^{5} \phi_{ijk} \\ \phi_{ijk} &= e^{\alpha_k + \beta_{ik} + \gamma_{jk}}; \end{aligned}$$

where $n_{ij} = \sum_{k=1}^{5} X_{ijk}$, and $\alpha_1, \beta_{i1}, \beta_{1k}, \gamma_{j1}, \gamma_{1k} = 0$ for identifiability. This model is discussed in detail in the BUGS manual section *Multinomial-logistic models*. All unknown α's, β's , γ's are initially given independent "noninformative" priors. The graph for the multinomial-logistic model is shown in Figure 21.

The BUGS manual discusses two ways of fitting this model: directly in the form given above or by using the multinomial-Poisson transformation which will be somewhat more efficient. Both techniques are illustrated in the code given on the next page.

We also illustrate how to transform the paramters for each food choice from the 'corner' parameter constraint used in the sampling to the Agresti constraint of making the parameters add to zero. Finally we calculate the standard goodness-of-fit (deviance) statistic

$$G^2 = 2 \sum_{ijk} X_{ijk} \log \frac{X_{ijk}}{n_{ij} p_{ijk}}.$$

Alligator: model specification in BUGS

```
model alli;
const
   I   = 4, # number of lakes
   J   = 2, # number of sizes
   K   = 5; # number of foods
var
 X[I,J,K],      # observations
 n[I,J],        # total for each covariate pattern
 E[I,J,K],      # fitted values
 OlogOE[I,J,K], # O log O/E
 G2,            # goodness-of-fit statistic
 mu[I,J,K],     # Poisson means
 phi[I,J,K],    # exp (beta[k] ' x[i,j])
 p[I,J,K],      # fitted probabilities
 lambda[I,J],   # baseline rates in each covariate strata
 alpha[K],      # factor for food = 2,3,4,5
 beta[I,K],     # factor for lakes = 2,3,4, for each food
 b[I,K],        # factor for lakes = 2,3,4, relative to food 1, centred
 gamma[J,K],    # factor for size = 2,  for each food
 g[J,K];        # factor for size = 2,  relative to food 1, centred
data X in "alli.dat";
inits in "alli.in";
{
#  TRANSFORMATIONS
  for (i in 1:I) {     # loop around lakes
     for (j in 1:J) {     # loop around sizes
        n[i,j] <- sum(X[i,j,]);
     }
   }

#  PRIORS

  alpha[1] <- 0;       # zero contrast for baseline food
  for (k in 2:K){ alpha[k] ~ dnorm(0,0.00001)} # vague priors
  # Loop around lakes:
  for (k in 1:K){  beta[1,k] <- 0 } # corner-point contrast with first lake
  for (i in 2:I) {
         beta[i,1] <- 0 ;  # zero contrast for baseline food
         for (k in 2:K){  beta[i,k] ~ dnorm(0,0.00001)} # vague priors
  }
  # Loop around sizes:
  for (k in 1:K){  gamma[1,k] <- 0}  # corner-point contrast with first size
  for (j in 2:J) {
         gamma[j,1] <- 0 ;  # zero contrast for baseline food
         for ( k in 2:K){ gamma[j,k] ~ dnorm(0,0.00001)} # vague priors
  }
```

```
# LIKELIHOOD

  for (i in 1:I) {     # loop around lakes
    for (j in 1:J) {     # loop around sizes

# Multinomial response
         X[i,j,] ~ dmulti( p[i,j,] , n[i,j]  );
         for (k in 1:K) {     # loop around foods
           p[i,j,k]        <- phi[i,j,k] / sum(phi[i,j,]);
           log(phi[i,j,k]) <- alpha[k] + beta[i,k]  + gamma[j,k];
         }
# Fit standard Poisson regressions relative to baseline
#        lambda[i,j] ~ dnorm(0,0.00001); # vague priors
#        for (k in 1:K) {     # loop around foods
#          X[i,j,k] ~ dpois(mu[i,j,k]);
#          log(mu[i,j,k]) <- lambda[i,j] + alpha[k] + beta[i,k]  + gamma[j,k];
#       }

    }
  }

# TRANSFORM OUTPUT TO ENABLE COMPARISON WITH AGRESTI'S RESULTS

for (k in 1:K) {     # loop around foods
  for (i in 1:I) {     # loop around lakes
     b[i,k] <- beta[i,k] - mean(beta[,k]);   # sum to zero constraint
  }
  for (j in 1:J) {     # loop around sizes
     g[j,k] <- gamma[j,k] - mean(gamma[,k]); # sum to zero constraint
  }
}

# FITTED VALUES
  for (i in 1:I) {     # loop around lakes
    for (j in 1:J) {     # loop around sizes
       for (k in 1:K) {     # loop around foods
#       p[i,j,k] <- mu[i,j,k]/sum(mu[i,j,]);  # fitted probabilities
        E[i,j,k] <- p[i,j,k] * n[i,j];
        OlogOE[i,j,k] <- X[i,j,k] * log( X[i,j,k] / E[i,j,k] );
      }
    }
  }
  G2 <- 2 * sum( OlogOE[,,] );
}
```

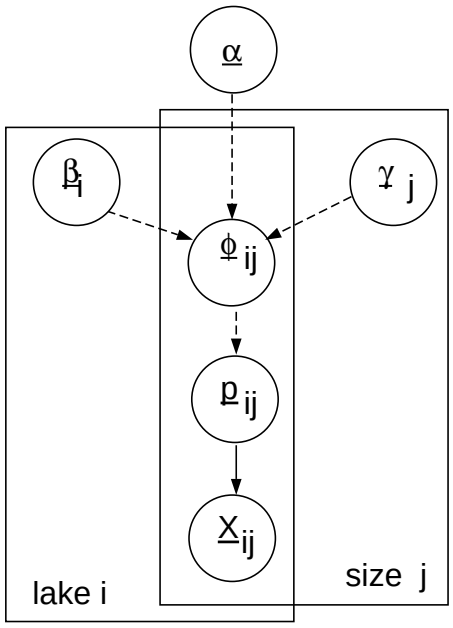

Figure 21: Graphical model for `alligator` example using the multinomial-logistic model

A `BUGS` run of 500 burn-in and 1000 iterations took 148 seconds using the direct method and 36 seconds using the Poisson transformation. Some results are compared below with those of Agresti, where the parameters for each food choice have been constrained to add to zero. Hence we transform the sampled values in `BUGS` as follows: $b_{ik} = \beta_{ik} - \sum_{i=1}^{4} \beta_{ik}/4, \;\; i = 1, 2, 3, 4, \;\; k = 2, 3, 4, 5$. Therefore the displayed values can be interpreted as representing the extent to which the log-odds of selecting each food choice, relative to fish, is attributable to Lake Hancock.

Parameter	Agresti (s.e.) MLE	`BUGS` (s.e.) Multi-logistic	Multi-poisson
b_{12}	-1.76 (.44)	-1.83 (.45)	-1.97 (.47)
b_{13}	-.42 (.56)	-.32 (.58)	-.42 (.65)
b_{14}	.41 (.51)	.60 (.53)	.61 (.56)
b_{15}	.24 (.35)	.29 (.37)	.27 (.37)
G^2	17.1	37.8 (6.7)	38.4 (6.9)

We note that the classical goodness-of-fit (deviance) statistic is 17.1 on 12 degrees of freedom. This should be contrasted with the minimum achieved values of 20.9 and 24.1 under the two Bayesian formulations.

15 Endo: conditional inference in case-control studies

Breslow and Day (1980) analyse a set of data from a case-control study relating endometrial cancer with exposure to estrogens. 183 pairs of cases and controls were studied, and the full data is shown below.

	Status of control	
	Not exposed	Exposed
Status of case		
Not exposed	$n_{00} = 121$	$n_{01} = 7$
Exposed	$n_{10} = 43$	$n_{11} = 12$

We denote estrogen exposure as x_{ij} for the ith case-control pair, where $j = 1$ for a case and $j = 2$ for a control. The conditional likelihood for the log (odds ratio) β is then given by

$$\prod_i \frac{e^{\beta x_{i1}}}{e^{\beta x_{i1}} + e^{\beta x_{i2}}}.$$

We shall illustrate three methods of fitting this model. It is convenient to denote the fixed disease status as a variable $Y_{i1} = 1, Y_{i2} = 0$.

First, Breslow and Day point out that for case-control studies with a single control per case, we may obtain this likelihood by using unconditional logistic regression for each case-control pair. That is

$$\begin{aligned} Y_{i1} &\sim \text{Binomial}(p_i, 2) \\ \text{Logit } p_i &= \beta(x_{i1} - x_{i2}) \end{aligned}$$

Second, the BUGS manual section *Conditional likelihoods in case-control studies* discusses fitting this likelihood directly by assuming the model

$$\begin{aligned} Y_{i.} &\sim \text{Multinomial}(p_{i.}, 1) \\ p_{ij} &= e_{ij} / \sum_{j=1}^{2} e_{ij} \\ \log e_{ij} &= \beta x_{ij} \end{aligned}$$

Finally, the BUGS manual shows how the multinomial-Poisson transformation can be used. In general, this will be more efficient than using the multinomial-logistic parameterisation above, since it avoids the time-consuming evaluation of $\sum_{j=1}^{2} e_{ij}$. However, in the present example this summation is only over J=2 elements, whilst the multinomial-Poisson parameterisation involves estimation of an additional intercept parameter for each of the 183 strata. Consequently the latter is *less* efficient than the multinomial-logistic in this case.

We note that all these formulations may be easily extended to include additional subject-specific covariates, and that the second and third methods can handle arbitrary numbers of controls per case. In addition, the Bayesian approach allows the incorporation of hierarchical structure, measurement error, missing data and so on.

The graph for the conditional likelihood model is shown in Figure 22.

All these techniques are illustrated in the code given below, which includes a transformation of the original summary statistics into full data. In this example, all but the second conditional-likelihood approach are commented out.

Endo: model specification in BUGS

```
model endo;
const
    I = 183, # number of matched sets
    J = 2;   # number of people per set
 var
   n10,n01,n11,n00,  # collapses form of data
   Y[I,J],    # observed disease status
   p[I,J],    # probability of disease status
   e[I,J],    # exp (  beta ' x )
   est[I,J] , # estrogen use
   mu[I,J],   # Poisson means
   beta0[I],  # baseline rates in each stratum
   beta;      # covariate coefficient
data n10,n01,n11,n00 in "endo.dat";
inits in "endo.in";
{
# transform collapsed data into full
    for (i in 1:I){ Y[i,1] <- 1;  Y[i,2] <- 0;}
# loop around strata with case exposed, control not exposed  (n10)
    for (i in 1:n10){ est[i,1] <- 1;   est[i,2] <- 0;}
# loop around strata with case not exposed, control   exposed  (n01)
    for (i in (n10+1):(n10+n01)){  est[i,1] <- 0;     est[i,2] <- 1;}
# loop around strata with case exposed, control   exposed  (n11)
    for (i in (n10+n01+1):(n10+n01+n11)){ est[i,1] <- 1;  est[i,2] <- 1;}
# loop around strata with case not exposed, control not exposed  (n00)
    for (i in (n10+n01+n11+1):I){ est[i,1] <- 0;  est[i,2] <- 0;}

# PRIORS
  for (i in 1:I) { beta0[i] ~ dnorm(0,1.0E-6) } beta ~ dnorm(0,1.0E-6) ;

# LIKELIHOOD
  for (i in 1:I) {                    # loop around strata
# METHOD 1 - logistic regression
#        Y[i,1] ~ dbin( p[i,1], 1);
# logit(p[i,1]) <- beta * (est[i,1] - est[i,2]);
# METHOD 2 - conditional likelihoods
     Y[i,] ~ dmulti( p[i,],1);
     for (j in 1:2){
             p[i,j] <- e[i,j] / sum(e[i,]);
      log( e[i,j] ) <- beta * est[i,j] ;
     }
# METHOD 3 fit standard Poisson regressions relative to baseline
#    for (j in 1:2) {
#       Y[i,j] ~ dpois(mu[i,j]);
#       log(mu[i,j]) <- beta0[i] +  beta*est[i,j];
#    }
  }
```

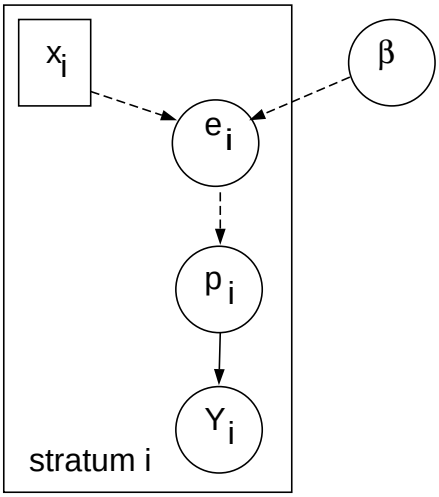

Figure 22: Graphical model for `endo` example using the conditional-likelihood approach.

A `BUGS` run of 500 burn-in and 1000 iterations, took the times shown below and gave the following (very similar) results.

		BUGS		
Parameter	MLE	Logistic	Multinomial logistic	Multinomial poisson
β	1.82	1.90	1.90	1.85
s.e.	*.41*	*.42*	*.42*	*.46*
time (secs)	–	5	21	225

16 Asia: a simple expert system

16.1 Evidence propagation

Lauritzen and Spiegelhalter (1988) introduce a fictitious "expert system" representing the diagnosis of a patient presenting to a chest clinic, having just come back from a trip to Asia and showing dyspnoea (shortness-of-breath). A graphical model for the underlying process is shown in the Figure 23, where each variable is binary. The BUGS code is shown below and the conditional probabilities used are given in Lauritzen and Spiegelhalter (1988).

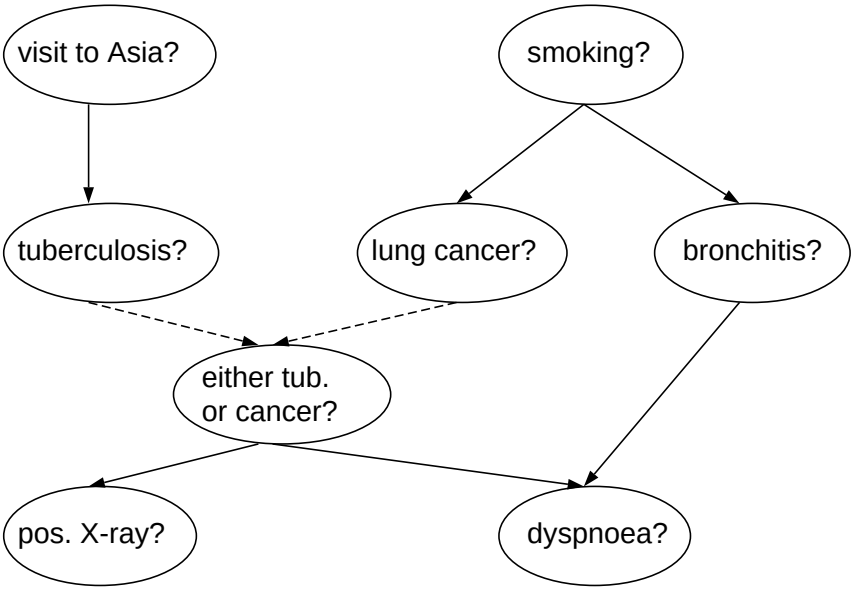

Figure 23: Graphical model for asia example

Asia: model specification in BUGS

```
model Asia;
var
   asia,smoking,tuberculosis,lung.cancer,bronchitis,either,xray,dyspnoea,
   p.asia[2],p.smoking[2],p.tuberculosis[2,2],p.bronchitis[2,2],
   p.lung.cancer[2,2],p.xray[2,2],p.dyspnoea[2,2,2];
data in "asia.dat";
{
   smoking      ~ dcat(p.smoking[]);
   tuberculosis ~ dcat(p.tuberculosis[asia,]);
   lung.cancer  ~ dcat(p.lung.cancer[smoking,]);
   bronchitis   ~ dcat(p.bronchitis[smoking,]);
   either      <- max(tuberculosis,lung.cancer);
   xray         ~ dcat(p.xray[either,]);
   dyspnoea     ~ dcat(p.dyspnoea[either,bronchitis,])
}
```

Note the use of `max` to do the logical-or. All initial values are computed by forward sampling so no initial value file is necessary. The `dcat` distribution is used to sample values with domain (1,2) with probability distribution given by the relevant entries in the conditional probability tables. The *S-Plus* format has been used for the data file, since these conditional probability tables are of different dimensions, and would require 4 separate data files in rectangular format.

Data in *S-Plus* format for `asia` example

```
list(asia = 2, dyspnoea = 2,
     p.asia         = c(0.99, 0.01),
     p.tuberculosis = c(0.99, 0.01,
                        0.95, 0.05),
     p.bronchitis   = c(0.70, 0.30,
                        0.40, 0.60),
     p.smoking      = c(0.50, 0.50),
     p.lung.cancer  = c(0.99, 0.01,
                        0.90, 0.10),
     p.xray         = c(0.95, 0.05,
                        0.02, 0.98),
     p.dyspnoea      = c(0.9, 0.1,
                         0.2, 0.8,
                         0.3, 0.7,
                         0.1, 0.9)
)
```

The observed features (`asia` and `dyspnoea`) are given value 2 in the data-file. 100000 iterations (31 seconds) gave the following posterior probabilities (the exact values are given in brackets): *smoking* .625 (626), *tuberculosis* .089 (.088), *lung cancer* .099 (.100), *bronchitis* .810 (.812), *either* .183 (.182) *x-ray* .220 (.220). Note that these probabilities are obtained by subtracting 1 from the posterior means of the variables `smoking`, `tuberculosis` *etc.* which are actually defined on the domain (1,2).

16.2 Learning about parameters

Spiegelhalter *et al.* (1993) describe techniques for *estimating* parameters (*i.e.* the conditional probabilities) of such a network, where these parameters can be represented by additional nodes connected to a set of networks corresponding to each of a set of cases. The parameters can be given independent Dirichlet distributions and, with complete data, standard conjugate Bayesian updating is straightforward. With incomplete data a number of different analytic approximations have been suggested, but in fact a simulation solution is easily implemented.

Figure 24 illustrates the `asia2` network in which θ_b represents the unknown conditional probability of *bronchitis?* given *smoking?*. Note that θ_b replaces the known conditional probability matrix `p.bronchitis` used in the first `asia` network described above. The observed part of the network is represented by the replicated *plates*. We illustrate learning about θ_b with a dataset of five cases, in which the true value for *smoking* is not observed for case 2, who has bronchitis and dyspnoea, and case 3, whose only positive feature is an x-ray.

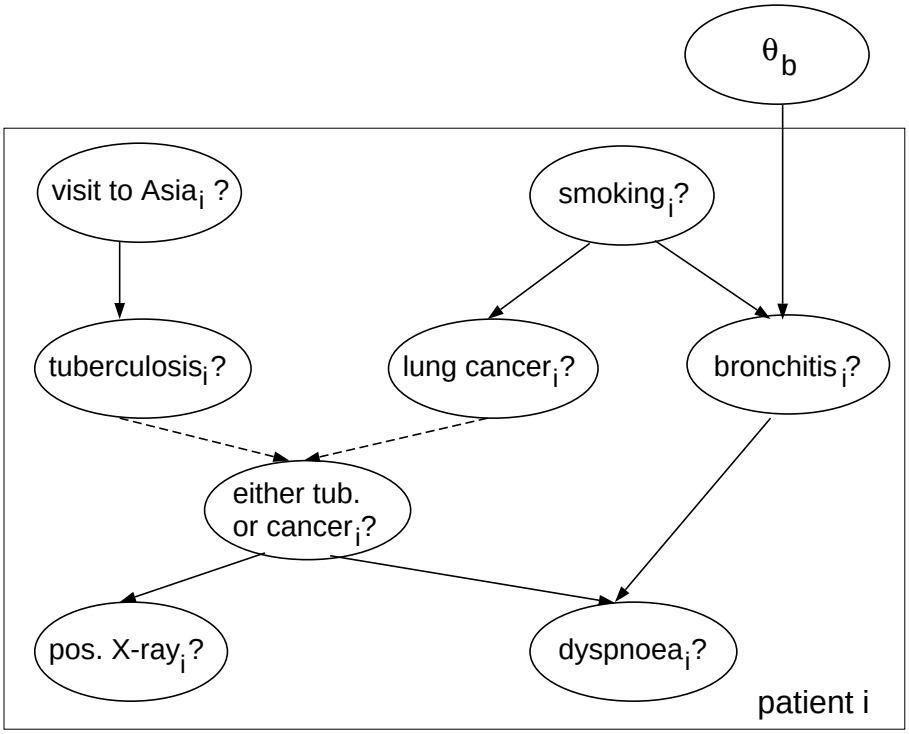

Figure 24: Graphical model for `asia2` example, with additional node θ_b representing the unknown conditional probability of *bronchitis?* given *smoking?*

The data file now does not contain values for `p.bronchitis`, but does have observed data for five cases.

Data for `asia2` example

```
list(p.asia         = c(0.99, 0.01),
     p.tuberculosis = c(0.99, 0.01,
                        0.95, 0.05),
     p.smoking      = c(0.50, 0.50),
     p.lung.cancer  = c(0.99, 0.01,
                        0.90, 0.10),
     p.xray         = c(0.95, 0.05,
                        0.02, 0.98),
     p.dyspnoea      = c(0.9, 0.1,
                         0.2, 0.8,
                         0.3, 0.7,
                         0.1, 0.9)

     asia         = c(1,1,1,1,1),
     smoking      = c(2,NA,NA,1,1),
     tuberculosis = c(1,1,1,1,1),
     lung.cancer  = c(2,1,1,1,1),
     bronchitis   = c(2,2,1,2,1),
     xray         = c(2,1,2,2,1),
     dyspnoea     = c(2,2,1,2,2))
```

The BUGS code (shown below) now requires the observables to be vectors, and has put independent Dirichlet prior probability distributions with parameters (1,1) (*i.e.* uniform priors) on each of the unknown conditional distributions *p(bronchitis | smoking=no)* and *p(bronchitis | smoking=yes)*.

Asia2: model specification in BUGS

```
model Asia2;
const
   N = 5;  # number of cases
var
   asia[N],smoking[N],tuberculosis[N],lung.cancer[N],
   bronchitis[N],either[N],xray[N],dyspnoea[N],
   p.asia[2],p.smoking[2],p.tuberculosis[2,2],theta.b[2,2],
   p.lung.cancer[2,2],p.xray[2,2],p.dyspnoea[2,2,2],prior[2];
data in "asia2.dat";
{
for (i in 1:N){
   smoking[i]      ~ dcat(p.smoking[]);
   tuberculosis[i] ~ dcat(p.tuberculosis[asia[i],]);
   lung.cancer[i]  ~ dcat(p.lung.cancer[smoking[i],]);
   bronchitis[i]   ~ dcat(theta.b[smoking[i],]);
   either[i]      <- max(tuberculosis[i],lung.cancer[i]);
   xray[i]         ~ dcat(p.xray[either[i],]);
   dyspnoea[i]      ~ dcat(p.dyspnoea[either[i],bronchitis[i],])
   }
#  priors for unknown probabilities
for (j in 1:2){
   theta.b[j,] ~ ddirch(prior[]);   # theta.b = p(bronchitis | smoking)
   prior[j] <- 1;
   }
}
```

Analysis

100000 iterations after a 1000 iteration burn-in took 36 seconds and led to posterior mean estimates (standard deviations) of *p(bronchitis | smoking=no)* = .52 (.21) and *p(bronchitis | smoking=yes)* = .66 (.23). In addition we estimate that for cases 2 and 3 respectively, there is a probability .56 and .37 that they are smokers.

17 `Pigs`: genetic counselling and pedigree analysis

Spiegelhalter (1990) uses exact methods to analyse a small pedigree. This pedigree was previously used by Cannings and Thompson (1981) to illustrate their 'peeling' procedure to provide likelihoods for gene frequencies and probabilities for individuals being affected or carriers. The pedigree is shown in Figure 25. We assume these are pigs which have the possibility of carrying a recessive gene: thus each pig has a genotype a_1a_1, a_1a_2 or a_2a_2, in which only those with a_2a_2 are affected with the trait, while those with a_1a_2 are carriers of the defective allele a_2. We assume that Ian (the consequence of a mating between Fred and his Aunt Clare) is yet to be born, and all that is known is that Fred's niece Jane has the trait. We wish to estimate the prevalence p of the allele a_2, and predict the chance of Ian being affected.

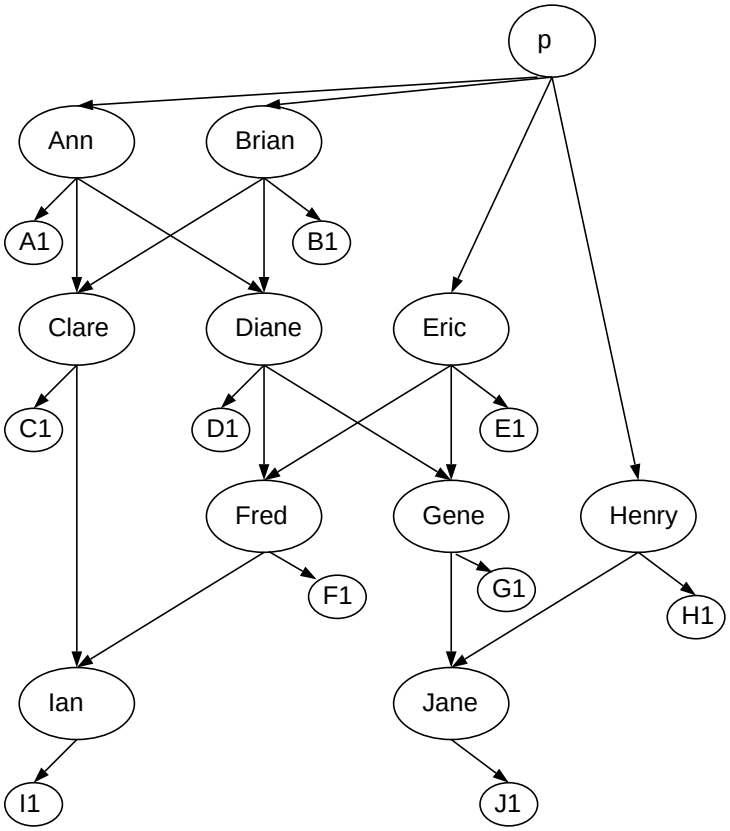

Figure 25: Graphical model for `pigs` example

The conditional probability distributions are as follows. For the genotype of the founder nodes Ann, Brian, Eric and Henry we assume a binomial distribution

$$\mathit{Founder} \sim \text{Binomial}(q, 2)$$

where *Founder* takes values 0, 1 or 2 for genotypes a_2a_2, a_1a_2 and a_1a_1 respectively, and $q = 1-p$ is the prevalence of the allele a_1. This is equivalent to assuming Hardy Weinberg equilibrium, giving $P(a_1a_1) = q^2, P(a_1a_2) = 2q(1-q), P(a_2a_2) = (1-q)^2$.

For the genotype of offspring we have the standard Mendelian inheritance probabilities given by the following table.

Genotype of parents	Genotype of offspring		
	1 (a_1a_1)	2 (a_1a_2)	3 (a_2a_2)
	Prob (genotype)		
a_1a_1 a_1a_1	1		
a_1a_1 a_1a_2	.5	.5	
a_1a_1 a_2a_2		1	
a_1a_2 a_1a_2	.25	.5	.25
a_1a_2 a_2a_2		.5	.5
a_2a_2 a_2a_2			1

For a recessive gene the genotype-to-phenotype penetrance probabilities are given by:

Genotype of individual	Phenotype of individual	
	1 (normal)	2 (affected)
	Prob (phenotype)	
a_1a_1	1	
a_1a_2	1	
a_2a_2		1

The necessary inheritance probabilities are read in from the data file as an array (note the use of the more convenient *S-Plus* format for this file).

Data in *S-Plus* object format

```
list(p.mendelian = c(1.0, 0.0, 0.0,
                     0.5, 0.5, 0.0,
                     0.0, 1.0, 0.0,
                     0.5, 0.5, 0.0,
                    0.25, 0.5,0.25,
                     0.0, 0.5, 0.5,
                     0.0, 1.0, 0.0,
                     0.0, 0.5, 0.5,
                     0.0, 0.0, 1.0),
     p.recessive = c(1.0, 0.0,
                     1.0, 0.0,
                     0.0, 1.0),
              A1 = 1, B1 = 1, C1 = 1,
              D1 = 1, E1 = 1, F1 = 1,
              G1 = 1, H1 = 1, J1 = 2)
```

Model specification in BUGS for the pigs example

```
model pigs;
var
   Ann,Ann1,A1, Brian,Brian1,B1, Clare,C1, Diane,D1, Eric,Eric1,E1,
   Fred,F1, Gene,G1, Henry,Henry1,H1, Ian,I1, Jane,J1,
   a,b,c,d,e,f,g,h,i[3],p,q, p.mendelian[3,3,3], p.recessive[3,2];

data  in "pigs.dat";
inits in "pigs.in";
{
   q ~ dunif(0,1);                          # prevalence of a1
   p <- 1 - q;                              # prevalence of a2
   Ann1   ~ dbin(q,2);   Ann <- Ann1 + 1;   # geno. dist. for founder
   Brian1 ~ dbin(q,2);   Brian <- Brian1 + 1;
   Clare  ~ dcat(p.mendelian[Ann,Brian,]);  # geno. dist. for child
   Diane  ~ dcat(p.mendelian[Ann,Brian,]);
   Eric1  ~ dbin(q,2);   Eric <- Eric1 + 1;
   Fred   ~ dcat(p.mendelian[Diane,Eric,]);
   Gene   ~ dcat(p.mendelian[Diane,Eric,]);
   Henry1 ~ dbin(q,2);   Henry <- Henry1 + 1;
   Ian    ~ dcat(p.mendelian[Clare,Fred,]);
   Jane   ~ dcat(p.mendelian[Gene,Henry,]);
   A1 ~ dcat(p.recessive[Ann,]);            # phenotype distribution
   B1 ~ dcat(p.recessive[Brian,]);
   C1 ~ dcat(p.recessive[Clare,]);
   D1 ~ dcat(p.recessive[Diane,]);
   E1 ~ dcat(p.recessive[Eric,]);
   F1 ~ dcat(p.recessive[Fred,]);
   G1 ~ dcat(p.recessive[Gene,]);
   H1 ~ dcat(p.recessive[Henry,]);
   I1 ~ dcat(p.recessive[Ian,]);
   J1 ~ dcat(p.recessive[Jane,]);
   a <- equals(Ann, 2);                     # event that Ann is carrier
   b <- equals(Brian, 2);
   c <- equals(Clare, 2);
   d <- equals(Diane, 2);
   e <- equals(Eric, 2);
   f <- equals(Fred, 2);
   g <- equals(Gene, 2);
   h <- equals(Henry, 2);
   for (J in 1:3) {
          i[J] <- equals(Ian, J)      # i[1] = a1 a1
                                      # i[2] = a1 a2
                                      # i[3] = a2 a2 (i.e. Ian affected)
   }
}
```

We note a number of important tricks. First, each genotype is a 3-valued categorical variable with conditional probabilities either determined by the binomial (Hardy-Weinberg equilibrium) distribution (for founder nodes) or from the Mendelian inheritance probabilities which are stored as a 3-dimensional matrix `p.mendelian`. In the latter case, the genotype of the parents picks which row of the matrix is used for the distribution. However, the rows of this matrix are indexed by values 1, 2 or 3, whilst the genotypes of the founder nodes take values 0, 1 or 2. Since `BUGS` does not allow subscripts to be functions of variables, we must first add 1 to the genotype of the parents (for example, `Ann = Ann1 + 1`) and use these new variables as subscripts to the matrix `p.mendelian`. The genotype-to-phenotype distribution is handled similarly in a matrix `p.recessive`. Second, the `equals` function `equals(Ann, 2)` allows the calculation of P(Ann's genotype = 2) (i.e. a carrier), whilst `equals(Ian, J)` calculates P(Ian's genotype = J), where J=3 implies that Ian is affected.

Analysis

A simple `BUGS` run took only 2 seconds for 2000 iterations after a 1000 iteration burn-in, and gave the following output.

variable	estimate	s.d.
p	0.662	0.154
$P(i)$ = `i[3]` = Prob(Ian is affected)	0.053	0.224

This can be compared to the exact answer, which is a polynomial in p with maximum at 0.67, conditional on which $P(i) = 0.06$.

We note that considerable care is required in doing Gibbs sampling in pedigrees due to the possibility that certain configurations of genotypes are not reachable from a single starting point, and hence the Markov Chain is not irreducible.

18 Cosmos: flexible mean and variance relationships using Legendre polynomial basis functions

We are grateful to Dr. David Mackay of the University of Cambridge Engineering Department for providing us with the following example.

The most accurate means of calculating the distances to galaxies involves measuring the period and magnitude (luminosity) of a class of supergiant variable stars known as Cepheids. Empirically, the magnitude and log period of nearby Cepheids show a linear relationship with a small amount of scatter. Hence measurement of the magnitude and period of a number of Cepheids in a galaxy gives direct distance information in the form of a constant offset between their magnitude–period line and the magnitude–period line of the nearby Cepheids. Such information may be used, for example, to deduce the value of the Hubble constant (Freedman *et al.*, 1994).

The standard analysis assumes a linear regression model with independent Normal errors. However, such assumptions may be inappropriately strong since it is not known that an exact straight line relationship holds, nor that the scatter is Gaussian and uniform. Here we use `BUGS` to model the distance between two simulated populations of Cephids assuming a polynomial relationship between magnitude and period and a noise level dependent on the magnitude.

This underlying relationship between magnitude (x) and period (t) is described by the following model:

$$\begin{aligned} t_i &\sim \text{Normal}(\mu_i, \tau_i) \\ \mu_i &= \sum_{h=1}^{K_w} w_h \phi_h(x_i) \\ \tau_i &= \exp\left(\sum_{h=1}^{K_b} b_h \phi_h(x_i)\right) \end{aligned}$$

where i indexes Cephids. The basis functions $\phi()$ are Legendre polynomials defined as follows:

$$\begin{aligned} \phi_1(x_i) &= 1 \\ \phi_2(x_i) &= x_i \\ \phi_h(x_i) &= \frac{(2h-3) \times x_i \times \phi_{h-1}(x_i) - (h-2) \times \phi_{h-2}(x_i)}{h-1} \qquad h > 2 \end{aligned}$$

The 'weight' parameters $\{w_h\}$ $(h = 1, .., K_w)$ and $\{b_h\}$ $(h = 1, .., K_b)$ are assumed to follow independent Normal distributions with population precision parameters ω_w and ω_b respectively. The latter are given non-informative gamma prior distributions. Note that we obtain the special case of uniform noise by setting $K_b = 1$ and the special case of a straight line relationship by setting $K_w = 2$. The quantity of interest, namely the distance between the two Cephid populations, is modelled by adding a scalar offset d to the mean μ_i of each Cephid in the second population. The graph for this model is shown in Figure 26.

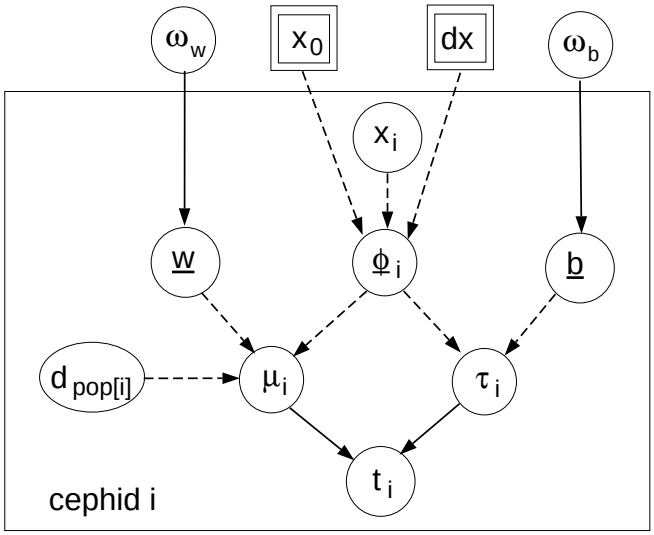

Figure 26: Graphical model for the `cosmos` example

Simulated Data

The above model was used to simulate (x, t) pairs for 50 Cephids from each of two populations using true values $K_b = K_w = 3$ and $d = 3.0$. These data are shown in Figure 27.

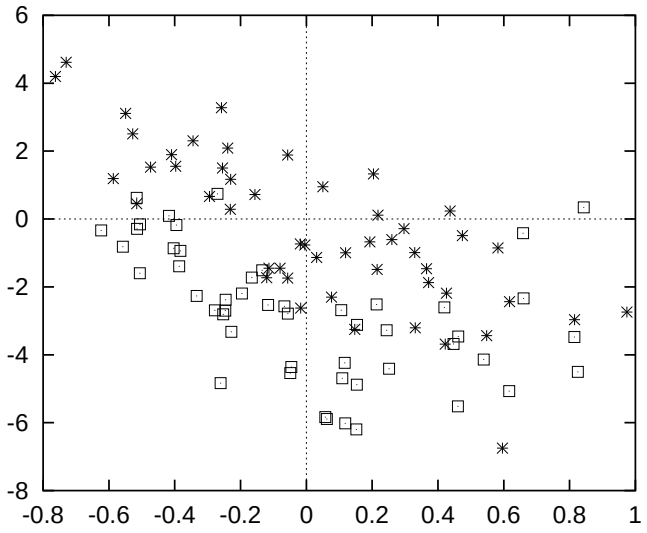

Figure 27: Simulated data from two Cephid populations differing by an offset d

Cosmos: model specification in BUGS

```
model cosmos;
const
  N=100,        # Total number of Cephids
  Kw=10,        # Number of basis functions for mu   [linear relation = 2]
  Kb=10,        # Number of basis functions for tau  [uniform noise = 1]
  K=10,         # K=max(Kw,Kb)
  x0=0.0, dx=1.0;
var
  x[N], t[N], pop[N], mu[N], tau[N], sigma[N],
  w[K], b[K],  phi[K,N], omega.w, omega.b, d[2];
data x, t, pop in "cosmos.dat" ;
inits in "cosmos.in"  ;
{
  for (i in 1:N) {
# Recurrence relation to define Legendre polynomials:
      phi[1,i] <- 1.0 ;
      phi[2,i] <- (x[i]-x0)/dx ;
      for (h in 3:K) { phi[h,i] <- ( ( 2*h-3 ) * (x[i]-x0)/dx * phi[h-1,i]
                                   - ( h-2 ) * phi[h-2,i] ) / ( h-1 ); }
  }
# Model:
  for (i in 1:N) {
      t[i]  ~  dnorm(mu[i], tau[i]);
      mu[i] <- d[pop[i]] + inprod(w[], phi[,i]);
      tau[i] <- exp(inprod(b[], phi[,i]));  sigma[i] <- 1/sqrt(tau[i]);
  }
  d[1] <- 0.0;                    # => d[pop[i]] = 0 if pop[i]=1
  d[2] ~ dnorm(0.0, 0.0001);      # offset between two populations

# Priors:
  for (h in 1:Kw) {
      w[h] ~ dnorm(0.0, omega.w);
  }
  for (h in Kw+1:K) {
      w[h] <- 0.0;            # Fill out array with zeros if Kw < K
  }
  for (h in 1:Kb) {
      b[h] ~ dnorm(0.0, omega.b);
  }
  for (h in Kb+1:K) {
      b[h] <- 0.0;            # Fill out array with zeros if Kb < K
  }
  omega.b ~ dgamma(1.0E-3, 1.0E-3);
  omega.w ~ dgamma(1.0E-3, 1.0E-3);
}
```

Note the use of the function `inprod()` to calculate $\sum_{h=1}^{K_w} w_h \phi_h(x_i)$ and $\sum_{h=1}^{K_b} b_h \phi_h(x_i)$. We also introduce $K = \max(K_w, K_b)$ and if $K_b < K$ the extra parameters $\{b_h\}_{K_b+1}^{K}$ are set to zero: this allows the use of a single matrix $\phi_h(x_i)$ when computing both the above inner products.

Results

We estimated 4 different models: $K_w = 2, K_b = 1$ (standard model); $K_w = 2, K_b = 10$ (linear model with non-uniform noise); $K_w = 10, K_b = 1$ (polynomial model with uniform noise); $K_w = 10, K_b = 10$ (polynomial model with non-uniform noise). For each, `BUGS` was run for a burn–in period of 500 iterations followed by 1000 further iterations, taking approximately 8 minutes. The posterior mean, standard deviation and 95% credible interval for the offset d are given below.

K_b	1			10		
K_w	mean	sd	95% interval	mean	sd	95% interval
2	2.442	0.3005	(1.857, 3.020)	2.761	0.2603	(2.250, 3.240)
10	2.542	0.3074	(1.983, 3.156)	2.714	0.2710	(2.180, 3.231)

The over–simple model $K_b = 1, K_w = 2$ gives a 95% interval for d that only just includes the true value of 3.0. Changing from the over–simple model to the model with $K_w = 10$ produces a slight increase in the uncertainty of d. The increase is only slight because there are two opposing effects: first, for any particular value of noise, the higher degree polynomial is less well determined and the uncertainty in d increases; but second, the greater flexibility of the magnitude-period relationship allows it to fit the curving shape of the data and makes smaller noise levels probable. Small noise levels give more accurate inferences. A similar effect occurs as we increase the number of terms in the representation of τ ($K_b = 10$). The estimation of d can become more precise, in intuitive terms, because the model is able to discover that some values of x give more reliable measurements than others, so that the inference of d can be based on them, ignoring the more noisy measurements. The net effect is that when we change from the over–simple model to the most flexible model ($K_w = 10, K_b = 10$), the 95% interval becomes smaller and more accurate. Whether this will happen for the real Cepheid data remains to be seen.

19 `Marsbars`: order constraints in two-way ANOVA

Gelfand *et al.* (1992) analyse fictitious data arranged in a two way table, intended to represent response data as might occur in consumer preference studies. We wish to impose the constraints that the row effects are decreasing while the column effects increase to the middle column and then decrease.

Simulated ordered two-way ANOVA data

	j =1	2	3	4	5
i =1	.982	1.902	3.797	-1.531	.570
i =2	-1.417	1.356	1.287	-3.629	-3.413
i =3	-1.601	4.713	.814	.834	-2.082
i =4	-4.912	-4.541	-4.768	-9.051	-2.744

Then we assume

$$\begin{aligned} Y_{ij} &\sim \text{Normal}(\mu_{ij}, \tau) \\ \mu_{ij} &= \alpha_i + \beta_j \\ \alpha_i &\sim \text{Normal}(\mu_\alpha, \tau_\alpha) \\ \beta_j &\sim \text{Normal}(\mu_\beta, \tau_\beta) \\ \tau &\sim \text{gamma}(a, b) \end{aligned}$$

Following Gelfand *et al.* (1992) we set $\mu_\alpha = \mu_\beta = 0, \tau_\alpha = \tau_\beta = .2$, and $a = 0, b = 1$. We wish to impose the constraints that $\alpha_1 > \alpha_2 > \alpha_3 > \alpha_4$, and $\beta_1 < \beta_2 < \beta_3 > \beta_4 > \beta_5$ The appropriate graph is shown in Figure 28, where undirected dashed lines are used to represent the logical order constraints.

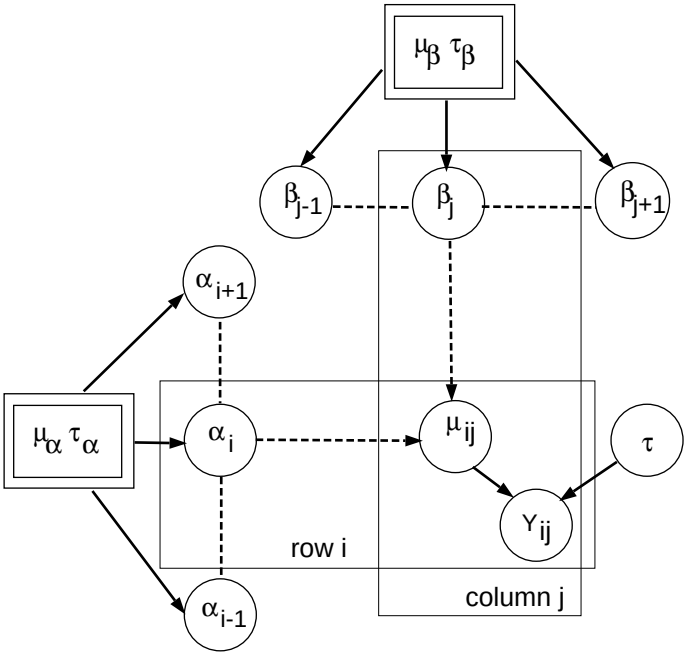

Figure 28: Graphical model for `marsbars` example

The BUGS code for this model is given below. We note the use of the `I(.,.)` notation to denote constraints, and the convenience with which other unknowns may be included into constraints.

Marsbars: model specification in BUGS

```
model MarsBars;
const
   cols = 5, rows = 4,
   tau.alpha = 0.20, mu.alpha = 0.0,
   tau.beta = 0.20, mu.beta = 0.0;
var
   Y[rows,cols], mu[rows,cols], tau, beta[cols],
   alpha[rows], bound, sigma;
data Y in "marsbars.dat";
inits in "marsbars.in";
{
   for(j in 1:cols) {
      for (i in 1:rows) {
          mu[i,j] <- alpha[i] + beta[j];
          Y[i,j]   ~ dnorm(mu[i,j],tau)
      }
   }
   tau      ~ dgamma(1.0E-3,1.0E-3);
   sigma   <- 1/sqrt(tau);
   alpha[1] ~ dnorm(mu.alpha,tau.alpha)I(alpha[2],);
   alpha[2] ~ dnorm(mu.alpha,tau.alpha)I(alpha[3],alpha[1]);
   alpha[3] ~ dnorm(mu.alpha,tau.alpha)I(alpha[4],alpha[2]);
   alpha[4] ~ dnorm(mu.alpha,tau.alpha)I(,alpha[3]);
   bound   <- max(beta[2],beta[4]);
   beta[1]  ~ dnorm(mu.beta,tau.beta)I(,beta[2]);
   beta[2]  ~ dnorm(mu.beta,tau.beta)I(beta[1],beta[3]);
   beta[3]  ~ dnorm(mu.beta,tau.beta)I(bound,);
   beta[4]  ~ dnorm(mu.beta,tau.beta)I(beta[5],beta[3]);
   beta[5]  ~ dnorm(mu.beta,tau.beta)I(,beta[4]);
}
```

Analysis

A run of 1000 iterations only took 1 second after a 500 iteration burn-in, and gave the following estimates, which may be compared to the posterior plots provided in Gelfand *et al.* (1992).

variable	estimate	95% interval
α_1	1.53	-0.34, 3.56
α_2	0.22	-1.44, 2.04
α_3	-0.38	-2.20, 1.46
α_4	-4.07	-6.21, -1.82

20 Stagnant: a changepoint problem

Carlin *et al.* (1992) analyse data from Bacon and Watts (1971) concerning a changepoint in a linear regression.

i	x_i	Y_i	i	x_i	Y_i	i	x_i	Y_i
1	-1.39	1.12	11	-.12	.60	21	.44	.13
2	-1.39	1.12	12	-.12	.59	22	.59	-.01
3	-1.08	.99	13	.01	.51	23	.70	-.13
4	-1.08	1.03	14	.11	.44	24	.70	.14
5	-.94	.92	15	.11	.43	25	.85	-.30
6	-.80	.90	16	.11	.43	26	.85	-.33
7	-.63	.81	17	.25	.33	27	.99	-.46
8	-.63	.83	18	.25	.30	28	.99	-.43
9	-.25	.65	19	.34	.25	29	1.19	-.65
10	-.25	.67	20	.34	.24			

We assume a model with two straight lines that meet at a certain changepoint x_k — this is slightly different from the model of Carlin *et al.* (1992) who do not constrain the two straight lines to cross at the changepoint. We assume

$$\begin{aligned} Y_i &\sim \text{Normal}(\mu_i, \tau) \\ \mu_i &= \alpha + \beta_{J[i]}(x_i - x_k) J[i] = 1 \quad \text{if} \quad i \le k;\ J[i] = 2 \quad \text{if} \quad i > k \end{aligned}$$

giving $\mathrm{E}(Y) = \alpha$ at the changepoint, with gradient β_1 before, and gradient β_2 after the changepoint. $\alpha, \beta_1, \beta_2, \tau$ are given independent "noninformative" priors. The appropriate graph is shown in Figure 29, and the BUGS code follows.

We note that to be able to update the changepoint in the current version of BUGS we are required to have the changepoint as a discrete random variable; this could be a discretised X although here we have followed Carlin *et al.* (1992) and forced the change to occur at one of the design points.

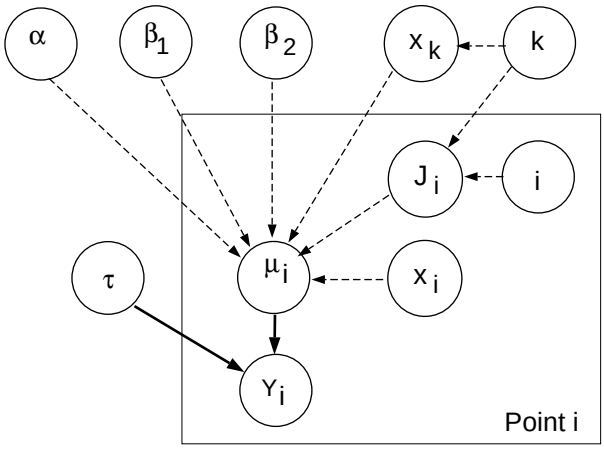

Figure 29: Graphical model for `stagnant` example

Model specification for the stagnant example

```
model stagnant;
const
   N = 29;  # number of points
var
    x[N], mu[N], Y[N], punif[N], J[N],
    k, alpha, beta[2], tau, sigma;
data Y, x, punif in  "stagnant.dat";
inits in "stagnant.in";
{
  k  ~ dcat(punif[]);  # uniform prior over changepoint observation
   for (i in 1:N) {
           J[i] <- 1 + step(i - (k+0.5));     # J[i]=1 if i<=k; 2 if i>k
           mu[i]      <- alpha + beta[J[i]]*(x[i] - x[k]);
           Y[i]        ~ dnorm(mu[i],tau)
  }
  alpha  ~ dnorm(0,1.0E-6);
  beta1  ~ dnorm(0,1.0E-6);
  beta2  ~ dnorm(0,1.0E-6);
  tau    ~ dgamma(1.0E-3,1.0E-3);
  sigma <- 1.0/sqrt(tau);
}
```

Analysis

A BUGS run took 29 seconds for 1000 iterations after a 500 iteration burn-in and gave the following output.

variable	estimate	95% interval
α	0.469	(0.452, 0.486)
β_1	-0.449	$(-0.471, -0.427)$
β_2	-1.037	$(-1.072, -1.007)$
k	15	(14, 16)

The parameter estimates are very similar to those of Carlin *et al.* (1992) and Bacon and Watts (1971). The entire posterior distribution for k lay in the range 14-16, all of which correspond to a changepoint at $X = .11$. Plotting the data supports this finding, and contrasts slightly with the analysis of Carlin *et al.* (1992) whose posterior mode lay at $k = 12$. However, their posterior median of $k = 13$ corresponds to the first 13 points being on one line, and the remainder on another, which is essentially our finding.

References

Agresti, A. (1990). *Categorical Data Analysis*. John Wiley and Sons, New York.

Bacon, D. W. and Watts, D. G. (1971). Estimating the transition betwen two intersecting straight lines. *Biometrika*, **58**, 525–34.

Berkson, J. (1950). Are there two regressionsΓ *Journal of the American Statistical Association*, **45**, 164–80.

Berry, D. A. (1987). Logarithmic transformations in anova. *Biometrics*, **43**, 439–56.

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems (with discussion). *Journal of the Royal Statistical Society, Series B*, **36**, 192–236.

Bliss, C. I. (1935). The calculation of the dosage-mortality curve. *Annals of Applied Biology*, **22**, 134–67.

Bowmaker, J. K., Jacobs, G. H., Spiegelhalter, D. J., and Mollon, J. D. (1985). Two types of trichromatic squirrel monkey share a pigment in the red-green region. *Vision Research*, **25**, 1937–46.

Breslow, N. E. and Clayton, D. G. (1993). Approximate inference in generalized linear mixed models. *Journal American Statistical Association*, **88**, 9–25.

Breslow, N. E. and Day, N. E. (1980). *Statistical Methods on Cancer Research Volume 1: Case-Control Studies* . International Agency for Cancer Research, Lyon.

Cannings, C. and Thompson, E. A. (1981). *Genealogical and Genetic Structures*. Cambridge University Press, Cambridge, UK.

Carlin, B. P. and Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B*, **57**, 473–84.

Carlin, B. P., Galfand, A. E., and Smith, A. F. M. (1992). Hierarchical Bayesian analysis of changepoint problems. *Applied Statistics*, **41**, 389–408.

Carlin, B. P. and Gelfand, A. E. (1991). An iterative Monte Carlo method for nonconjugate Bayesian analysis. *Statistics and Computing*, **1**, 119–28.

Carroll, R., Gail, M., and Lubin, J. (1993). Case-control studies with errors in covariates. *Journal of the American Statistical Association*, **88**, 185–99.

Clayton, D. G. and Kaldor, J. (1987). Empirical Bayes estimates of age-standardised relative risks for use in disease mapping. *Biometrics*, **43**, 671–81.

Cox, D. R. and Hinkley, D. (1974). Chapman and Hall, London.

Dobson, A. J. (1983). *An introduction to statistical modelling*. Chapman and Hall, London.

Elston, R. C. and Grizzle, J. F. (1962). Estimation of time response curves and their confidence bands. *Biometrics*, **18**, 148–59.

Farewell, V. T. and Sprott, D. A. (1988). The use of a mixture model in the analysis of count data. *Biometrics*, **44**, 1191–4.

Freedman *et al.* (1994). *Nature*, **371**, 757–62.

Gelfand, A. E., Hills, S. E., Racine-Poon, A., and Smith, A. F. M. (1990). Illustration of Bayesian inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association*, **85**, 972–85.

Gelfand, A. E., Smith, A. F. M., and Lee, T.-M. (1992). Bayesian analysis of constrained parameter and trucated data problems using Gibbs sampling. *Journal of the American Statistical Association*, **87**, 523–32.

Goldstein, H. (1979). *The design and analysis of longitudinal studies.* Academic Press, London.

Lauritzen, S. L. and Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems (with discussion). *Journal of the Royal Statistical Society, Series B*, **50**, 157–224.

Lindley, D. V. (1970). The estimation of many parameters. In *Foundation of statistical inference.* Holt, Rinehart and Winston, Toronto.

Prosser, R., Rasbash, J., and Goldstein, H. (1991*a*). *ML3: Software for three-level analysis.* London.

Prosser, R., Rasbash, J., and Goldstein, H. (1991*b*). *ML3: Software for three-level analysis. Users' guide for V.2.* Institute of Education, University of London.

Ratkowsky, D. (1983). *Nonlinear regression modelling.* Marcel Dekker, New York.

Robert, C. (1994). Mixtures of distributions: inference and estimation. In *Markov chain Monte Carlo in practice*, (ed. W. Gilks, S. Richardson, and D. Spiegelhalter). Chapman and Hall. (to appear).

Spiegelhalter, D. J. (1990). Fast algorithms for probabilistic reasoning in influence diagrams, with applications in genetics and expert systems. In *Influence Diagrams, Belief Nets and Decision Analysis*, (ed. R. M. Oliver and J. Q. Smith), pp. 361–84. Wiley, Chichester.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., and Cowell, R. G. (1993). Bayesian analysis in expert systems (with discussion). *Statistical Science*, **8**, 219–83.

Spiegelhalter, D. J. and Stovin, P. G. I. (1983). An analysis of repeated biopsies following cardiac transplantation. *Statistics in Medicine*, **2**, 33–40.

Stephens, D. and Dellaportas, P. (1992). Bayesian analysis of generalised linear models with covariate measurement error. In *Bayesian Statistics 4*, (ed. J. Bernardo, J. Berger, A. Dawid, and A. Smith). Clarendon Press, Oxford, UK.

Whittemore, A. and Keller, J. (1988). Approximations for regression with covariate measurement error. *Journal of the American Statistical Association*, **83**, 1057–66.

Williams, E. (1959). *Regression analysis.* Wiley, New York.