`tumor` — **An Example of a Binomial/Normal Exchangeable Model**

Consider data from a study from Tarone (1982) and analyzed by Gelman et al (1995) in which the incidence of a particular kind of tumor was recorded for 70 groups of female lab rats of type F344. For each group of rats, the group size n_i and the tumor incidence (proportion of rats with tumors) y_i is recorded. This data is stored in matrix form with two columns, where the first column contains the tumor incidences for all groups and the second column contains the corresponding group sizes. The data matrix, stored in the file `tumor.dat` is displayed in part below:

```
     0    20.0000
     0    20.0000
     0    20.0000
     0    20.0000
     0    20.0000
0.2000    20.0000
  . . .
0.3261    46.0000
0.3750    24.0000
```

The objective is to simultaneously estimate the tumor probabilities $p_1, ..., p_{70}$ for the 70 groups. Suppose that it is believed that the tumor probabilities are exchangeable. We model this belief by the following hierarchical prior distribution.

1. y_i is distributed binomial(n_i, p_i), $i = 1, ..., 70$
2. logit $\theta_i = \log[p_i/(1-p_i)]$ is distributed $\mathrm{N}(\mu, \tau^2)$, $i = 1, ..., 70$
3. μ and τ^2 are independent, μ is distributed $\mathrm{N}(a_1, b_1)$ and $\log \tau^2$ is distributed $\mathrm{N}(a_2, b_2)$

At the first stage, the observed proportions $\{y_i\}$ are sampled from independent binomial distributions with respective tumour probabilities $\{p_i\}$. We reparameterize the probabilities to the logits $\{\theta_i = \log[p_i/(1-p_i)]\}$. To model the belief that the probabilities, or equivalently the logits, are exchangeable, we assume in the second stage that the $\{\theta_i\}$ are a random sample from a common normal distribution with unknown mean μ and variance τ^2. Stage 3 of the model specifies prior distributions for the two normal hyperparameters. The mean and log variance are taken independent from normal distributions with known hyperparameters a_1, b_1, a_2, b_2.

The Matlab function `cihm` will produce an approximate simulated sample from the joint posterior distribution of a general exchangeable model. The algorithm, from Albert (1996), is based on Gibbs sampling with Metropolis/random walk sampling at each conditional simulation.

For the binomial/normal exchangeable model, this command is given by

```
S=cihm(tumor,'f_bin','g_normal','h_normal',theta,lambda,hpar,m,b);
```

To use the function `cihm`, you input the data matrix, the functional form of the model at each of the three stages, starting values for all parameters, the values of hyperparameters, and the length of "burn-in" and total number of iterations to collect in the Gibbs sampling run.

1. **The Data:**

   The matrix in the Matlab workspace (here, `tumor`) that contains the observed data.

2. **The Model:**

   (a) The function which defines the stage 1 sampling density. Here we are assuming binomial sampling and the binomial distribution is defined in the function `f_bin`.

   (b) The function which defines the stage 2 prior. Here we assume a normal density and this is defined in `g_normal`.

   (c) The function which defines the stage 3 prior. At the last stage, we assume independent normal distributions for the mean and the log variance. The definition of this density is contained in `h_normal`.

3. **The Starting Values:**

   (a) The column vector `theta` of starting estimates for the logits $\{\theta_i\}$. Here reasonable starting values are the empirical logits $\{(n_i y_i + .5)/(n_i(1-y_i)+.5)\}$.

   (b) The row vector `lambda` of starting estimates for the hyperparameters at stage 2, μ and $\log\tau^2$. The choice here is not crucial; in the example, we used `lambda = [ 1 1]`.

4. **Hyperparameters:**

   A row vector `hpar` of values of the four hyperparameters at stage 3 of the prior. In this application, we assign the mean μ a vague normal distribution with mean 0 and variance 100. The log variance $\log\tau^2$ is assigned a N(1,1) distribution. So here `hpar = [0 100 1 1]`.

5. **The Gibbs Sampler:**

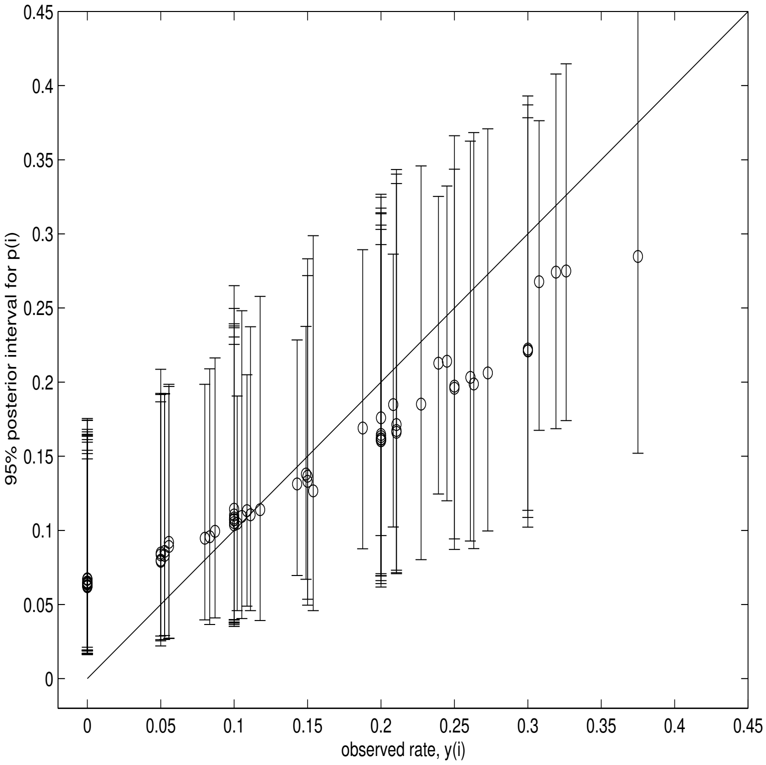
   (a) The number `m` of iterations of the Gibbs sampler.

   (b) The number `b` of burn-in iterations of the sampler. The sizes of the steps of the proposal densities in the Metropolis random walk chain are determined from these burn-in iterations.

In this example, we used a burn-in time of $b = 1000$ iterations and then collected $m = 2000$ simulated values. The simulated values are stored in the matrix S. The first 70 columns contain the simulated values of the 70 logits $\theta_1, ..., \theta_{70}$ and the last two columns contain simulated values of the hyperparameters μ and $\log\tau^2$.

Here we focus on summarizing the posterior distributions for the 70 probabilities $p_1, ..., p_{70}$. In the Matlab code below, we compute the 2.5th, 50th and 97.5th percentiles of each of the 70 simulated samples of the logits $\{\theta_i\}$. These are transformed to give the corresponding percentiles of the probabilities $\{p_i\}$. The 50th percentiles correspond to the posterior medians. The 2.5th and 97.5th percentiles give an approximate 95% credible interval for the probabilities. The percentiles are graphed against the observed tumor rates using errorbars. The line $y = x$ is placed on top of the graph so we can compare the posterior estimates against the observed tumor rates. This graph is similar to a graph of posterior distributions in Gelman et al (1995), p. 134, who use a similar model for exchangeability of the 70 tumor rates.

```
pt=prctile(S(:,1:70),[2.5 50 97.5]);                         % compute percentiles
p=exp(pt)./(1+exp(pt));                                      % transform to probs
errorbar(tumor(:,1),p(2,:),p(2,:)-p(1,:),p(3,:)-p(2,:),'o')% plot errorbars
axis([-.02 .45 -.02 .45]);line([0 .45],[0 .45])              % plot line on top
xlabel('observed rate, y(i)')                                % label
ylabel('95% posterior interval for p(i)')                    % axes
```


95% posterior interval for p(i)
observed rate, y(i)
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45