Bayesian Modeling, Inference, Prediction and Decision-Making

4: Hierarchical and Mixture Modeling

David Draper

Department of Applied Mathematics and Statistics
University of California, Santa Cruz

draper@ams.ucsc.edu
www.ams.ucsc.edu/∼draper

eBay/Google

10 Fridays, 11 Jan–22 Mar 2013 (except 25 Jan)

Short course web page:
www.ams.ucsc.edu/∼draper/eBay-Google-2013.html

Hierarchical Models for Combining Information

Formulating hierarchical models for quantitative outcomes from scientific context

Case Study: *Meta-analysis of effects of* **aspirin** *on heart attacks*. Table 5.1 (Draper et al., 1993a) gives the number of patients and **mortality rate** from all causes, for six **randomized controlled experiments** comparing the use of aspirin and placebo by patients following a heart attack.

Table 5.1. Aspirin meta-analysis data.

	Aspirin		Placebo	
	# of	Mortality	# of	Mortality
Study (i)	Patients	Rate (%)	Patients	Rate (%)
UK-1	615	7.97	624	10.74
CDPA	758	5.80	771	8.30
GAMS	317	8.52	309	10.36
UK-2	832	12.26	850	14.82
PARIS	810	10.49	406	12.81
AMIS	2267	10.85	2257	9.70
Total	5599	9.88	5217	10.73

	Comparison			
	$y_i =$ Diff	$\sqrt{V_i} =$ SE		
Study (i)	(%)	of Diff (%)	$Z_i^{\ddagger}$	$p_i^{\S}$
UK-1	2.77	1.65	1.68	.047
CDPA	2.50	1.31	1.91	.028
GAMS	1.84	2.34	0.79	.216
UK-2	2.56	1.67	1.54	.062
PARIS	2.31	1.98	1.17	.129
AMIS	−1.15	0.90	−1.27	.898
Total	0.86	0.59	1.47	.072

$^{\ddagger}Z_i$ denotes the ratio of the difference in mortality rates over its standard error, assuming a binomial distribution. $^{\S}p_i$ is the one-sided p value associated with Z_i, using the normal approximation.

Meta-Analysis

The first five trials are reasonably consistent in showing a (weighted average) **mortality decline** for aspirin patients of 2.3 percentage points, a **20% drop** from the (weighted average) placebo mortality of 11.5% (this difference is **highly clinically significant**).

However, the sixth and largest trial, AMIS, went the other way: an **increase** of 1.2 percentage points in aspirin mortality (a 12% rise from the placebo baseline of 9.7%).

Some **relevant questions** (Draper, 1995):

$\boxed{\mathbf{Q}_1}$ Why did AMIS get such **different results**?

$\boxed{\mathbf{Q}_2}$ What should be done next to **reduce the uncertainty** about Q_1?

$\boxed{\mathbf{Q}_3}$ If you were a doctor treating a patient like those eligible for the trials in Table 5.1, **what therapy should you employ** while answers to Q_1 and Q_2 are sought?

One possible **paraphrase** of Q_3: $\boxed{\mathbf{Q}_4}$ How should the information from these six experiments be **combined** to produce a **more informative summary** than those obtained from each experiment by itself?

The discipline of **meta-analysis** is devoted to answering questions like Q_4.

One leading school of **frequentist meta-analysis** (e.g., Hedges and Olkin, 1985) looks for methods for combining the Z and p values in Table 5.1, an approach that often leads only to an overall p value.

A Gaussian HM

A **more satisfying** form of meta-analysis (which has both frequentist and Bayesian versions) builds a **hierarchical model (HM)** that indicates how to combine information from the mortality differences in the table.

A **Gaussian meta-analysis model** for the aspirin data, for example (Draper et al., 1993a), might look like

$$\begin{array}{rcll} (\mu, \sigma^2) & \sim & p(\mu, \sigma^2) & \text{(prior)} \\ (\theta_i | \mu, \sigma^2) & \overset{\text{IID}}{\sim} & N(\mu, \sigma^2) & \text{(underlying effects)} \\ (y_i | \theta_i) & \overset{\text{indep}}{\sim} & N(\theta_i, V_i) & \text{(data) .} \end{array} \qquad (1)$$

The bottom level of (1), the **data** level of the HM, says that—because of relevant differences in patient cohorts and treatment protocols—each study has its own **underlying treatment effect** θ_i, and the observed mortality differences y_i are like random draws from a normal distribution with mean θ_i and variance V_i (the normality is reasonable because of the **Central Limit Theorem**, given the large numbers of patients).

In meta-analyses of data like those in Table 5.1 the V_i are typically taken to be **known** (again because the patient sample sizes are so big), $V_i = SE_i^2$, where SE_i is the standard error of the mortality difference for study i in Table 5.1.

The middle level of the HM is where you would bring in the **study-level covariates**, if you have any, to try to explain why the studies differ in their underlying effects.

Here there are no study-level covariates, so the middle level of (1) is equivalent to a **Gaussian regression with no predictor variables**.

A Gaussian HM (continued)

Why the "error" distribution should be **Gaussian** at this level of the HM is not clear—it's a **conventional** option, not a choice that's automatically scientifically reasonable (in fact I'll challenge it later).

σ^2 in this model represents **study-level heterogeneity**.

The top level of (1) is where the **prior** distribution on the regression parameters from the middle level is specified.

Here, with only an intercept term in the regression model, a popular **conventional choice** is the normal/scaled-inverse-χ^2 prior we looked at earlier in our first Gaussian case study.

Fixed effects and random effects. If σ^2 were known somehow to be 0, all of the θ_i would have to be equal **deterministically** to a common value μ, yielding a simpler model: $(y_i|\mu) \overset{\text{indep}}{\sim} N(\mu, V_i)\,, \mu \sim p(\mu)$.

Meta-analysts call this a **fixed-effects** model, and refer to model (1) as a **random-effects** model.

When σ^2 is not assumed to be 0, with this terminology the θ_i are called **random effects** (this parallels the use of this term in the **random-effects Poisson regression** case study).

Approximate Fitting of Gaussian Hierarchical Models: Maximum Likelihood and Empirical Bayes

Fitting HM (1). Some algebra based on model (1) yields that the conditional distributions of the study-level effects θ_i given the data and the parameters (μ, σ^2), have a **simple and revealing form** (I'll show this later):

$$\left(\theta_i | y_i, \mu, \sigma^2\right) \overset{\text{indep}}{\sim} N[\theta_i^*, V_i(1 - B_i)], \tag{2}$$

$$\text{with} \quad \theta_i^* = (1 - B_i)\, y_i + B_i\, \mu \quad \text{and} \quad B_i = \frac{V_i}{V_i + \sigma^2}. \tag{3}$$

In other words, the conditional mean of the effect for study i given y_i, μ, and σ^2 is a **weighted average** of the sample mean for that study, y_i, and the overall mean μ.

The weights are given by the so-called **shrinkage factors** B_i (e.g., Draper et al., 1993a), which in turn depend on how the variability V_i **within study** i compares to the **between-study** variability σ^2: the more accurately y_i estimates θ_i, the more weight the "local" estimate y_i gets in the weighted average.

The term **shrinkage** refers to the fact that, with this approach, unusually high or low individual studies are **drawn back** or "shrunken" toward the overall mean μ when making the calculation $(1 - B_i)\, y_i + B_i\, \mu$.

Note that θ_i^* uses data from all the studies to estimate the effect for study i—this is referred to as **borrowing strength** in the estimation process.

Closed-form expressions for $p(\mu|y)$ and $p(\theta_i|y)$ with $y = (y_1, \ldots, y_k), k = 6$ are not available even with a normal-χ^{-2} prior for (μ, σ^2); **MCMC** is needed (see below).

Maximum Likelihood and Empirical Bayes

In the meantime **maximum likelihood** calculations provide some idea of what to expect: the likelihood function based on model (1) is

$$l\left(\mu, \sigma^2 | y\right) = c \left[\prod_{i=1}^{k} \frac{1}{\sqrt{V_i + \sigma^2}}\right] \exp\left[-\frac{1}{2}\sum_{i=1}^{k} \frac{(y_i - \mu)^2}{V_i + \sigma^2}\right]. \qquad (4)$$

The maximum likelihood estimates (MLEs) $\left(\widehat{\mu}, \widehat{\sigma}^2\right)$ then turn out to be the **iterative** solutions to the following equations:

$$\widehat{\mu} = \frac{\sum_{i=1}^{k} \widehat{W}_i \, y_i}{\sum_{i=1}^{k} \widehat{W}_i} \quad \text{and} \quad \widehat{\sigma}^2 = \frac{\sum_{i=1}^{k} \widehat{W}_i^2 \left[(y_i - \widehat{\mu})^2 - V_i\right]}{\sum_{i=1}^{k} \widehat{W}_i^2}, \qquad (5)$$

$$\text{where} \quad \widehat{W}_i = \frac{1}{V_i + \widehat{\sigma}^2}. \qquad (6)$$

Start with $\widehat{\sigma}^2 = 0$ and **iterate (5–6) to convergence** (if $\widehat{\sigma}^2$ converges to a negative value, $\widehat{\sigma}^2 = 0$ is the MLE); the MLEs of the θ_i are then given by

$$\widehat{\theta}_i = \left(1 - \widehat{B}_i\right) y_i + \widehat{B}_i \, \widehat{\mu} \quad \text{where} \quad \widehat{B}_i = \frac{V_i}{V_i + \widehat{\sigma}^2}. \qquad (7)$$

These are called **empirical Bayes** (EB) estimates of the study-level effects, because it turns out that this analysis approximates a fully Bayesian solution by (in effect) using the data to **estimate** the prior specifications for μ and σ^2.

Large-sample (mainly meaning large k) approximations to the (frequentist) distributions of the MLEs are given by

$$\widehat{\mu} \sim N\left(\mu, \left[\sum_{i=1}^{k} \frac{1}{V_i + \widehat{\sigma}^2}\right]^{-1}\right) \quad \text{and} \quad \widehat{\theta}_i \sim N\left[\theta_i, V_i\left(1 - \widehat{B}_i\right)\right]. \qquad (8)$$

MLEB (continued)

NB The variances in (8) **don't account fully for the uncertainty in** σ^2 and therefore underestimate the actual sampling variances for small k (adjustments are available; see, e.g., Morris (1983, 1988)).

MLEB estimation can be **implemented** simply in about 15 lines of `R` code (Table 5.2).

Table 5.2. `R` program to perform MLEB calculations.

```
mleb <- function( y, V, m ) {
  sigma2 <- 0.0
  for ( i in 1:m ) {
    W <- 1.0 / ( V + sigma2 )
    theta <- sum( W * y ) / sum( W )
    sigma2 <- sum( W^2 * ( ( y - theta )^2 - V ) ) / sum( W^2 )
    B <- V / ( V + sigma2 )
    effects <- ( 1 - B ) * y + B * theta
    se.theta <- 1.0 / sqrt( sum( 1.0 / ( V + sigma2 ) ) )
    se.effects <- sqrt( V * ( 1.0 - B ) )
    print( c( i, theta, se.theta, sigma2 ) )
    print( cbind( W, ( W / sum( W ) ), B, y, effects, se.effects ) )
  }
}
```

With the aspirin data it takes **18 iterations** (less than 0.1 second on a 400MHz `UltraSPARC` Unix machine) to get convergence to **4-digit accuracy**, leading to the summaries in Table 5.3 and the following estimates (standard errors in parentheses):

$$\widehat{\mu} = 1.45 \ (0.809), \quad \widehat{\sigma}^2 = 1.53.$$

Table 5.3. Maximum likelihood empirical Bayes meta-analysis of the aspirin data.

study(i)	$\widehat{W}_i$	normalized $\widehat{W}_i$	$\widehat{B}_i$	y_i	$\widehat{\theta}_i$	$\widehat{SE}\left(\widehat{\theta}_i\right)$
1	0.235	0.154	0.640	2.77	1.92	0.990
2	0.308	0.202	0.529	2.50	1.94	0.899
3	0.143	0.0934	0.782	1.84	1.53	1.09
4	0.232	0.151	0.646	2.56	1.84	0.994
5	0.183	0.120	0.719	2.31	1.69	1.05
6	0.427	0.280	0.346	−1.15	−0.252	0.728

Aspirin Meta-Analysis: Conclusions

Note that (1) AMIS gets **much less weight** (normalized $\widehat{W}_i$) than would have been expected given its small V_i; (2) the **shrinkage factors** ($\widehat{B}_i$) are considerable, with AMIS shrunk almost all the way into positive territory (see Figure 5.1); (3) there is **considerable study-level heterogeneity** ($\widehat{\sigma} \doteq 1.24$ percentage points of mortality); and (4) the standard errors of the effects are by and large smaller than the $\sqrt{V_i}$ (from the **borrowing of strength**) but are still considerable.

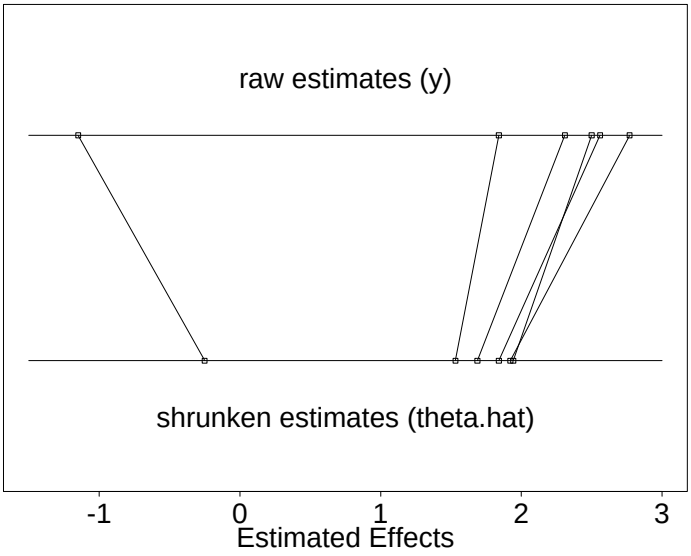


Figure 5.1. **Shrinkage plot** for the aspirin MLEB meta-analysis.

The **95% interval estimate** of μ, the overall underlying effect of aspirin on mortality, from this approach comes out

$$\widehat{\mu} \pm 1.96 \cdot \widehat{SE}(\widehat{\mu}) \doteq (-0.140, 3.03),$$

which if **interpreted Bayesianly** gives

$$\boxed{P(\text{aspirin reduces mortality}|\text{data}) \doteq 1 - \Phi\left(\tfrac{0-1.45}{0.809}\right) = \mathbf{0.96}},$$

where Φ is the **standard normal CDF**.

Thus although the interval includes 0, so that in a frequentist sense the effect is not statistically significant, **in fact from a Bayesian point of view the evidence is running strongly in favor of aspirin's usefulness**.

MCMC Details

In many cases (as with this example) empirical Bayes methods have the advantage of yielding **closed-form solutions**, but I view them at best as approximations to fully Bayesian analyses—which can in any case be carried out with MCMC—so I won't have any more to say about EB methods here (see Carlin and Louis, 1996, for more on this topic).

MCMC details. First let's derive that **likelihood function** I mentioned on page 7: the **model**, once again, is

$$\begin{array}{rcll} (\mu, \sigma^2) & \sim & p(\mu, \sigma^2) & \text{(prior)} \\ (\theta_i|\mu, \sigma^2) & \overset{\text{IID}}{\sim} & N(\mu, \sigma^2) & \text{(underlying effects)} \\ (y_i|\theta_i) & \overset{\text{indep}}{\sim} & N(\theta_i, V_i) & \text{(data)} \ . \end{array} \tag{9}$$

The **parameters** we're interested in here are (μ, σ^2); **Bayes's Theorem** gives (as usual)

$$p(\mu, \sigma^2|y) = c\, p(\mu, \sigma^2)\, p(y|\mu, \sigma^2), \tag{10}$$

so let's look at the **sampling distribution** for a single y_i:

$$\begin{aligned} p(y_i|\mu, \sigma^2) &= \int_{-\infty}^{\infty} p(y_i, \theta_i|\mu, \sigma^2)\, d\theta_i \\ &= \int_{-\infty}^{\infty} p(y_i|\theta_i, \mu, \sigma^2)\, p(\theta_i|\mu, \sigma^2)\, d\theta_i \qquad (11) \\ &= \int_{-\infty}^{\infty} p(y_i|\theta_i)\, p(\theta_i|\mu, \sigma^2)\, d\theta_i \end{aligned}$$

(what we're doing here is **integrating out** the **random effect** θ_i).

Now $p(y_i|\theta_i)$ is **normal** in this model, and $p(\theta_i|\mu, \sigma^2)$ is **also normal**; you could put in the **normal densities** and **grind away** at the **algebra** and **integration**, but there's a **better way**: the last line of (11) is a **mixture representation**, and a **normal mixture of normals is normal**, so I know that $p(y_i|\mu, \sigma^2)$ is **normal**, and the only questions are, what are its **mean** and **variance**?

Adam and Eve

These questions can be answered with **little difficulty** via the **Double Expectation Theorem**, which has **two parts** that are so **central** to **Bayesian calculations** that **Carl Morris** refers to them as **Adam** and **Eve**: for any two random variables X and Y,

$$\begin{array}{rcll} E(Y) & = & E_X\left[E(Y|X)\right] & \text{(Adam)} \\ V(Y) & = & E_X\left[V(Y|X)\right] + V_X\left[E(Y|X)\right] & \text{(Eve),} \end{array} \tag{12}$$

in which E_X and V_X refer to **expectation** and **variance** with respect to the **distribution** of X.

If there's **additional conditioning** going on, you just need to remember to **include it** in all the **relevant places**: for any three random variables X, Y and Z,

$$\begin{array}{rcl} E(Y|Z) & = & E_{(X|Z)}\left[E(Y|X,Z)\right] \\ V(Y|Z) & = & E_{(X|Z)}\left[V(Y|X,Z)\right] + V_{(X|Z)}\left[E(Y|X,Z)\right], \end{array} \tag{13}$$

and **so on**.

The application here is in **two parts** (Adam and Eve):

$$\begin{array}{rcl} E(y_i|\mu,\sigma^2) & = & E_{(\theta_i|\mu,\sigma^2)}\left[E(y_i|\mu,\sigma^2,\theta_i)\right] \\ & = & E_{(\theta_i|\mu,\sigma^2)}\left[E(y_i|\theta_i)\right] \\ & = & E_{(\theta_i|\mu,\sigma^2)}\left[\theta_i\right] \\ & = & \mu, \quad \text{and} \end{array}$$

$$\begin{array}{rcl} V(y_i|\mu,\sigma^2) & = & E_{(\theta_i|\mu,\sigma^2)}\left[V(y_i|\mu,\sigma^2,\theta_i)\right] + V_{(\theta_i|\mu,\sigma^2)}\left[E(y_i|\mu,\sigma^2,\theta_i)\right] \\ & = & E_{(\theta_i|\mu,\sigma^2)}\left[V(y_i|\theta_i)\right] + V_{(\theta_i|\mu,\sigma^2)}\left[E(y_i|\theta_i)\right] \\ & = & E_{(\theta_i|\mu,\sigma^2)}\left[V_i\right] + V_{(\theta_i|\mu,\sigma^2)}\left[\theta_i\right] \\ & = & V_i + \sigma^2. \end{array} \tag{14}$$

Direct Use of Gibbs Sampling

So (a) $(y_i|\mu,\sigma^2) \sim N(\mu, V_i+\sigma^2)$, (b) by **inspection** of the **form** of the **model**, the y_i are **independent** given (μ,σ^2), so

$$\begin{aligned} l(\mu,\sigma^2|y) &= c\,p(y|\mu,\sigma^2) = c\prod_{i=1}^{k} p(y_i|\mu,\sigma^2) \qquad (15)\\ &= c\left[\prod_{i=1}^{k}\frac{1}{\sqrt{V_i+\sigma^2}}\right]\exp\left[-\frac{1}{2}\sum_{i=1}^{k}\frac{(y_i-\mu)^2}{V_i+\sigma^2}\right], \end{aligned}$$

as **desired**.

MCMC: how best to sample from the posterior?

All **MCMC** (with a **parameter space** of **fixed dimension**) is one **special case** or another of the **Metropolis-Hastings** algorithm, but (as usual) we have a **number of possibilities**: **generic** (e.g., **random-walk**) **Metropolis?** Metropolis mixed with **Gibbs** steps? **All Gibbs?** With or without **auxiliary** (e.g., **latent**) **variables**? ...

First let's try **direct Gibbs**, for which we would need the **full conditionals**:

$$\begin{aligned} p(\mu|\sigma^2,y) &= c\,p(\mu,\sigma^2,y) \qquad (16)\\ &= c\,p(\mu,\sigma^2)\,p(y|\mu,\sigma^2). \end{aligned}$$

By virtue of **integrating out** the **random effects** above, we have $p(y|\mu,\sigma^2)$ as a **product** of **independent univariate Gaussians**; what shall we take for the **prior** $p(\mu,\sigma^2)$, given that there's no **conjugate choice**?

Even with **somewhat informative priors** on a **vector of parameters**, for **simplicity** people often assume **independence** of the **components** — in this case, $p(\mu,\sigma^2) = p(\mu)\,p(\sigma^2)$ — on the ground that whatever **correlation** the parameters should have in the **posterior** will be learned via the **likelihood function**; let's make this **simplifying assumption**; then

$$p(\mu|\sigma^2,y) = c\,p(\mu)\,p(\sigma^2)\,p(y|\mu,\sigma^2) = c\,p(\mu)\,p(y|\mu,\sigma^2). \qquad (17)$$

Direct Gibbs; Latent Gibbs

Now the **product of two Gaussians is Gaussian**, so if we take the **prior** for μ to be **Gaussian** we'll have a **Gaussian full conditional** for μ that'll be **easy to sample from**; what about σ^2?

$$\begin{aligned} p(\sigma^2|\mu, y) &= c\, p(\mu, \sigma^2, y) \\ &= c\, p(\mu, \sigma^2)\, p(y|\mu, \sigma^2) \\ &= c\, p(\mu)\, p(\sigma^2)\, p(y|\mu, \sigma^2) \\ &= c\, p(\sigma^2)\, p(y|\mu, \sigma^2). \end{aligned} \tag{18}$$

Here we run into **trouble**: when considered as a **function** of σ^2 for fixed μ and y, $p(y|\mu, \sigma^2)$ is **not recognizable** as a member of a **standard parametric family** (because the y_i (given μ and σ^2) are **independent** but **not identically distributed**); we could choose, e.g., a χ^{-2} prior on σ^2 and use **rejection sampling** to sample from the resulting **non-standard full conditional**, but that would not be especially **pleasant**.

So instead let's use a **trick** that's generally helpful in **random-effects** models: treat the **(latent) random effects** as **auxiliary variables** to be sampled along with (μ, σ^2).

In other words, letting $\theta = (\theta_1, \ldots, \theta_k)$, we're going to sample from the **augmented posterior** $p(\mu, \sigma^2, \theta|y)$; the hope is that this will have **completely tractable full conditionals**; let's see.

$$\begin{aligned} p(\mu|\sigma^2, \theta, y) &= c\, p(\mu, \sigma^2, \theta, y) \\ &= c\, p(\mu, \sigma^2)\, p(\theta|\mu, \sigma^2)\, p(y|\theta, \mu, \sigma^2) \end{aligned} \tag{19}$$

Notice how naturally this **factorization** matches the **hierarchical character** of (9), which starts at the **top** with a model for (μ, σ^2), and then builds a **model** for $(\theta|\mu, \sigma^2)$, and then at the **bottom** there's a model for $p(y|\theta, \mu, \sigma^2)$, which — by virtue of the **hierarchical** structure — can be **simplified** to $p(y|\theta)$.

Latent Gibbs (continued)

Since (a) we're **assuming** that $p(\mu, \sigma^2) = p(\mu)\, p(\sigma^2)$ and (b) $p(y|\theta)$ **doesn't involve** μ, the **full conditional** for μ becomes

$$p(\mu|\sigma^2, \theta, y) = c\, p(\mu)\, p(\theta|\mu, \sigma^2)\,; \tag{20}$$

with a **Gaussian** prior on μ this will be **Gaussian**; how about σ^2?

$$\begin{aligned} p(\sigma^2|\mu, \theta, y) &= c\, p(\mu, \sigma^2, \theta, y) \\ &= c\, p(\mu, \sigma^2)\, p(\theta|\mu, \sigma^2)\, p(y|\theta, \mu, \sigma^2) \\ &= c\, p(\mu)\, p(\sigma^2)\, p(\theta|\mu, \sigma^2)\, p(y|\theta) \\ &= c\, p(\sigma^2)\, p(\theta|\mu, \sigma^2). \end{aligned} \tag{21}$$

Here's another **trick**: instead of **slogging** through the **details**, try to **recognize** situations in which you already know the **conjugate updating**, and just use what you **already know**.

For example, in this calculation $(\theta|\mu, \sigma^2)$ is **Gaussian** with **known** μ and **unknown** σ^2, and we know the **conjugate prior** for σ^2 in that model — χ^{-2} — so with that **prior choice** the **full conditional** for σ^2 will also be χ^{-2}; how about θ?

$$\begin{aligned} p(\theta|\mu, \sigma^2, y) &= c\, p(\mu, \sigma^2, \theta, y) \\ &= c\, p(\mu, \sigma^2)\, p(\theta|\mu, \sigma^2)\, p(y|\theta, \mu, \sigma^2) \\ &= c\, p(\theta|\mu, \sigma^2)\, p(y|\theta). \end{aligned} \tag{22}$$

Here $p(\theta|\mu, \sigma^2)$ and $p(y|\theta)$ are both **Gaussian**, so the **full conditional** for θ — the **product** — will also be **Gaussian**.

Thus using the **latent Gibbs** approach in this **random-effects** model, all of the **full conditionals** have familiar forms; this approach will **work smoothly**; we just need to work out the **details**.

(I recommend this as a **basic Gibbs strategy**: in the first step make a **sketchy pass** through the **full conditionals** without working out all of the details, to ensure that everything **works fine**, and then go back and **fill in the details**.)

Details

(1) **Full conditional** for μ:

$$p(\mu|\sigma^2, \theta, y) = c\, p(\mu)\, p(\theta|\mu, \sigma^2). \tag{23}$$

In this **calculation** (a) σ^2 is known and (b) the **latent vector** $\theta = (\theta_1, \ldots, \theta_k)$ acts like the **data vector** $y = (y_1, \ldots, y_n)$ in the model $\mu \sim N(\mu_0, \sigma^2_{\mu 0}), (y_i|\mu) \overset{\text{IID}}{\sim} N(\mu, \sigma^2)$ $(i = 1, \ldots, n)$, so we already know the **answer**: $(\mu|\sigma^2, \theta, y) \sim N(\mu_k, \sigma^2_k)$, where

$$\mu_k = \frac{k_0 \mu_0 + k\bar{\theta}}{k_0 + k} \quad \text{and} \quad \sigma^2_k = \frac{\sigma^2}{k_0 + k}, \tag{24}$$

and in which the **prior sample size** is $k_0 = \frac{\sigma^2}{\sigma^2_{\mu 0}}$ and $\bar{\theta} = \frac{1}{k} \sum_{i=1}^{k} \theta_i$.

(2) **Full conditional** for σ^2:

$$p(\sigma^2|\mu, \theta, y) = c\, p(\sigma^2)\, p(\theta|\mu, \sigma^2). \tag{25}$$

In **parallel** with the situation with μ, in this **calculation** (a) μ is known and (b) the **latent vector** $\theta = (\theta_1, \ldots, \theta_k)$ acts like the **data vector** $y = (y_1, \ldots, y_n)$ in the model $\sigma^2 \sim \chi^{-2}(\nu_0, \sigma^2_{\sigma 0}), (y_i|\sigma^2) \overset{\text{IID}}{\sim} N(\mu, \sigma^2)$ $(i = 1, \ldots, n)$, so we already know the **answer**: $(\sigma^2|\mu, \theta, y) \sim \chi^{-2}(\nu_k, \sigma^2_k)$, where

$$\nu_k = \nu_0 + k \quad \text{and} \quad \sigma^2_k = \frac{\nu_0\, \sigma^2_{\sigma 0} + k\, v}{\nu_0 + k}, \tag{26}$$

in which $v = \frac{1}{k} \sum_{i=1}^{k} (\theta_i - \mu)^2$.

Details (continued)

(3) **Full conditional** for θ:

$$\begin{aligned} p(\theta|\mu,\sigma^2,y) &= c\,p(\theta|\mu,\sigma^2)\,p(y|\theta) \\ &= c\prod_{i=1}^{k} p(\theta_i|\mu,\sigma^2)\,p(y_i|\theta_i). \end{aligned} \tag{27}$$

Now $(\theta_i|\mu,\sigma^2) \sim N(\mu,\sigma^2)$ and $(y_i|\theta_i) \sim N(\theta_i, V_i)$ (with V_i known), so this is just our **old friend**

{**Gaussian likelihood** (for y_i) with **unknown mean** θ_i and **known variance** V_i + **Gaussian prior** for θ_i with **hyper-parameters** μ and σ^2};

the **(un-normalized) product** $p(\theta_i|\mu,\sigma^2)\,p(y_i|\theta_i)$ is just the **posterior** for θ_i, and the **answer** is therefore the **same** as it was in the **full conditional** for μ:
$(\theta_i|\mu,\sigma^2,y) \sim N(\theta_i^*,\sigma_i^2)$, with

$$\begin{aligned} \theta_i^* &= \frac{\frac{1}{\sigma^2}\mu + \frac{1}{V_i}y_i}{\frac{1}{\sigma^2}+\frac{1}{V_i}} = \frac{V_i\mu+\sigma^2 y_i}{V_i+\sigma^2} = B_i\mu + (1-B_i)y_i \quad \text{and} \\ \sigma_i^2 &= \frac{1}{\frac{1}{\sigma^2}+\frac{1}{V_i}} = \frac{V_i\sigma^2}{V_i+\sigma^2} = V_i(1-B_i), \end{aligned} \tag{28}$$

in which $B_i = \frac{V_i}{V_i+\sigma^2}$ is the **shrinkage factor** for study i (this is the **demonstration** of equations (2) and (3) earlier).

Thus $(\theta|\mu,\sigma^2,y) \sim N_k(\theta^*,\Sigma)$ with $\theta^* = (\theta_1^*,\ldots,\theta_k^*)$ and $\Sigma = \text{diag}(\sigma_i^2)$, and **one scan** of the **Gibbs sampler** can be described as follows:

(a) **draw** μ from $p(\mu|\sigma^2,\theta,y)$, **obtaining** μ_*;

(b) **draw** σ^2 from $p(\sigma^2|\mu_*,\theta,y)$, **obtaining** σ_*^2; and

(c) **draw** θ from $p(\theta|\mu_*,\sigma_*^2,y)$, either **univariately** on the θ_i (one by one) or **multivariately** on θ all at once.

R Code

```
meta.analysis.gibbs <- function( mu.0, sigma2.mu.0, nu.0, sigma2.sigma.0,
  mu.initial, sigma2.initial, theta.initial, y, V, M, B ) {

  k <- length( y )

  mu <- rep( 0, M + B + 1 )

  sigma2 <- rep( 0, M + B + 1 )

  theta <- matrix( 0, M + B + 1, k )

  mu[ 1 ] <- mu.initial

  sigma2[ 1 ] <- sigma2.initial

  theta[ 1, ] <- theta.initial

  for ( m in 2:( M + B + 1 ) ) {

    mu[ m ] <- mu.full.conditional( mu.0, sigma2.mu.0, sigma2[ m - 1 ],
      theta[ m - 1, ], y )

    sigma2[ m ] <- sigma2.full.conditional( nu.0, sigma2.sigma.0,
      mu[ m ], theta[ m - 1, ], y )

    theta[ m, ] <- theta.full.conditional( mu[ m ], sigma2[ m ], y, V )

    if ( m %% 1000 == 0 ) print( m )

  }

  return( cbind( mu, sigma2, theta ) )

}

mu.full.conditional <- function( mu.0, sigma2.mu.0, sigma2.current,
  theta.current, y ) {

  k <- length( y )

  k.0 <- sigma2.current / sigma2.mu.0

  theta.bar <- mean( theta.current )
```

R Code (continued)

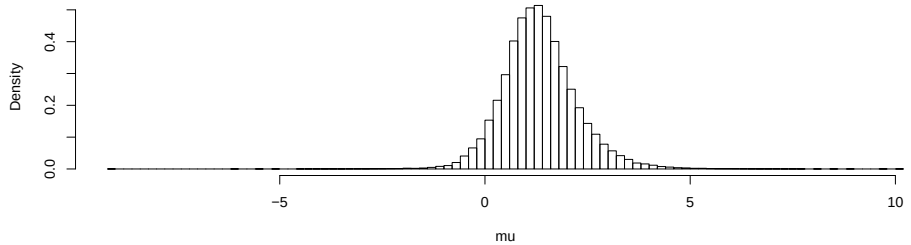
```
  mu.k <- ( k.0 * mu.0 + k * theta.bar ) / ( k.0 + k )

  sigma2.k <- sigma2.current / ( k.0 + k )

  mu.star <- rnorm( n = 1, mean = mu.k, sd = sqrt( sigma2.k ) )

  return( mu.star )

}

sigma2.full.conditional <- function( nu.0, sigma2.sigma.0,
  mu.current, theta.current, y ) {

  k <- length( y )

  nu.k <- nu.0 + k

  v <- mean( ( theta.current - mu.current )^2 )

  sigma2.k <- ( nu.0 * sigma2.sigma.0 + k * v ) / ( nu.0 + k )

  sigma2.star <- rsichi2( 1, nu.k, sigma2.k )

  return( sigma2.star )

}

rsichi2 <- function( n, nu, sigma2 ) {

  sigma2.star <- 1 / rgamma( n, shape = nu / 2,
    rate = nu * sigma2 / 2 )

  return( sigma2.star )

}

theta.full.conditional <- function( mu.current, sigma2.current, y, V ) {

  k <- length( y )

  theta.star <- ( V * mu.current + sigma2.current * y ) /
    ( V + sigma2.current )
```

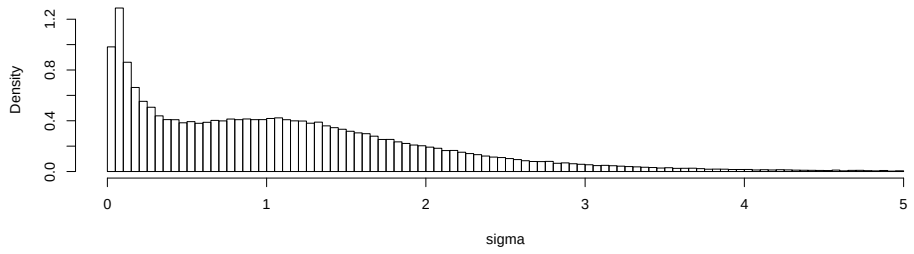
R Code (continued)

```
  sigma2.star <- V * sigma2.current / ( V + sigma2.current )

  theta.sim <- rnorm( n = k, mean = theta.star,
    sd = sqrt( sigma2.star ) )

  return( theta.sim )

}

mu.0 <- 0.0

sigma2.mu.0 <- 100^2

nu.0 <- 0.001

sigma2.sigma.0 <- 1.53

mu.initial <- 1.45

sigma2.initial <- 1.53

theta.initial <- c( 1.92, 1.94, 1.53, 1.84, 1.69, -0.252 )

y <- c( 2.77, 2.50, 1.84, 2.56, 2.32, -1.15 )

V <- c( 1.65, 1.31, 2.34, 1.67, 1.98, 0.90 )^2

M <- 100000

B <- 1000

mcmc.data.set <- meta.analysis.gibbs( mu.0, sigma2.mu.0, nu.0,
  sigma2.sigma.0, mu.initial, sigma2.initial, theta.initial,
  y, V, M, B )

% took 47 seconds

mcmc.data.set <- cbind( mcmc.data.set[ , 1:2 ],
  sqrt( mcmc.data.set[ , 2 ] ), mcmc.data.set[ , 3:8 ] )
```

R Code (continued)

```
apply( mcmc.data.set[ 1001:101001, ], 2, mean )

        mu     sigma2
1.33013835 2.24106295 1.12196766 1.68639681 1.67526967 1.38514567 1.62389213
1.51615795 0.09356775

apply( mcmc.data.set[ 1001:101001, ], 2, sd )

       mu    sigma2
0.9042468 4.4707971 0.9910910 1.1576621 1.0311309 1.2381000 1.1391841
1.1917662 0.9944885

mu.star <- mcmc.data.set[ 1001:101001, 1 ]

sum( mu.star > 0 ) / length( mu.star )

[1] 0.9484605

sigma.star <- mcmc.data.set[ 1001:101001, 3 ]

par( mfrow = c( 2, 1 ) )

hist( mu.star, nclass = 100, main = '', probability = T,
  xlab = 'mu' )

hist( sigma.star[ sigma.star < 5 ], nclass = 100, main = '',
  probability = T, xlab = 'sigma' )
```





WinBUGS Analysis of Aspirin Data

Aspirin meta-analysis revisited. I create three files for WinBUGS: a **model** file, a **data** file, and an **initial values** file (I'm using the most recent release, 1.4.1, of WinBUGS).

The (first) **model** file for the aspirin data:

```
{

  mu ~ dnorm( 0.0, 1.0E-6 )
  tau.theta ~ dgamma( 1.0E-3, 1.0E-3 )

  for ( i in 1:k ) {

    theta[ i ] ~ dnorm( mu, tau.theta )
    y[ i ] ~ dnorm( theta[ i ], tau.y[ i ] )

  }

  sigma.theta <- 1.0 / sqrt( tau.theta )

}
```

WinBUGS Analysis of Aspirin Data

Here μ plays the role of θ in model (10) above to avoid using the name `theta` twice for two different purposes in the `WinBUGS` program.

In specifying a normal distribution `WinBUGS` works not with a **standard deviation** (SD) or a **variance** but with a **precision**—the **reciprocal** of the variance—so that the $N(\mu, \sigma^2)$ distribution is specified by `dnorm( mu, tau )` with $\tau = \frac{1}{\sigma^2}$.

Then the **SD** has to be computed as a derived quantity ($\sigma = \frac{1}{\sqrt{\tau}}$) which is written above as

`sigma.theta <- 1.0 / sqrt( tau.theta )`

If—before the aspirin experiments were performed—I'm relatively **ignorant** about the quantities θ (μ) and σ in model (10), or equivalently μ and $\tau = \frac{1}{\sigma^2}$, I can build a **diffuse** or `flat` prior for both quantities that expresses this relative ignorance.

Since μ lives on $(-\infty, \infty)$ a convenient choice for a flat prior for it is a **normal** distribution with mean (say) 0 and very small precision: `mu` $\sim$ `dnorm( 0.0, 1.0E-6 )`

For `tau.theta`, which lives on $(0, \infty)$, I want something that's flat throughout (almost) all of that range; a convenient choice (to get an **initial idea** of where the posterior distribution for `sigma.theta` is **concentrated**) is a **gamma** distribution with small positive values of both of its parameters.

This is the $\boxed{\Gamma(\epsilon, \epsilon)}$ distribution for some **small** $\epsilon > 0$ like 0.001: `tau.theta` $\sim$ `dgamma( 1.0E-3, 1.0E-3 )`

WinBUGS **Aspirin Analysis (continued)**

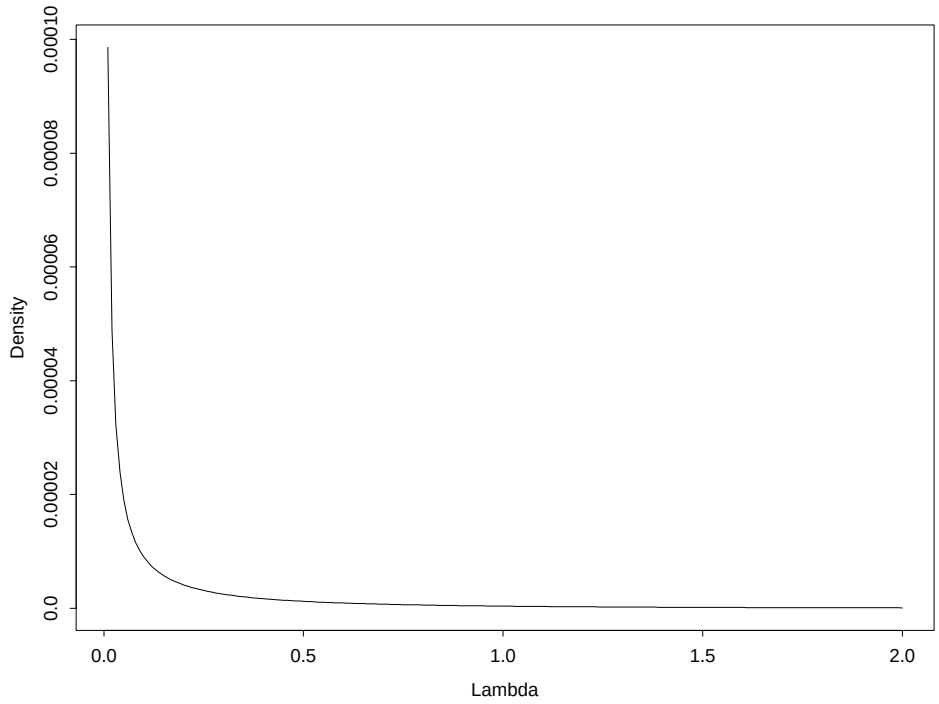


Figure 3.1. The $\Gamma(0.001, 0.001)$ distribution.

The **data** file in the aspirin meta-analysis is

```
list( k = 6, y = c( 2.77, 2.50, 1.84, 2.56, 2.31, -1.15 ),
  tau.y = c( 0.3673, 0.5827, 0.1826, 0.3586, 0.2551, 1.235 ) )
```

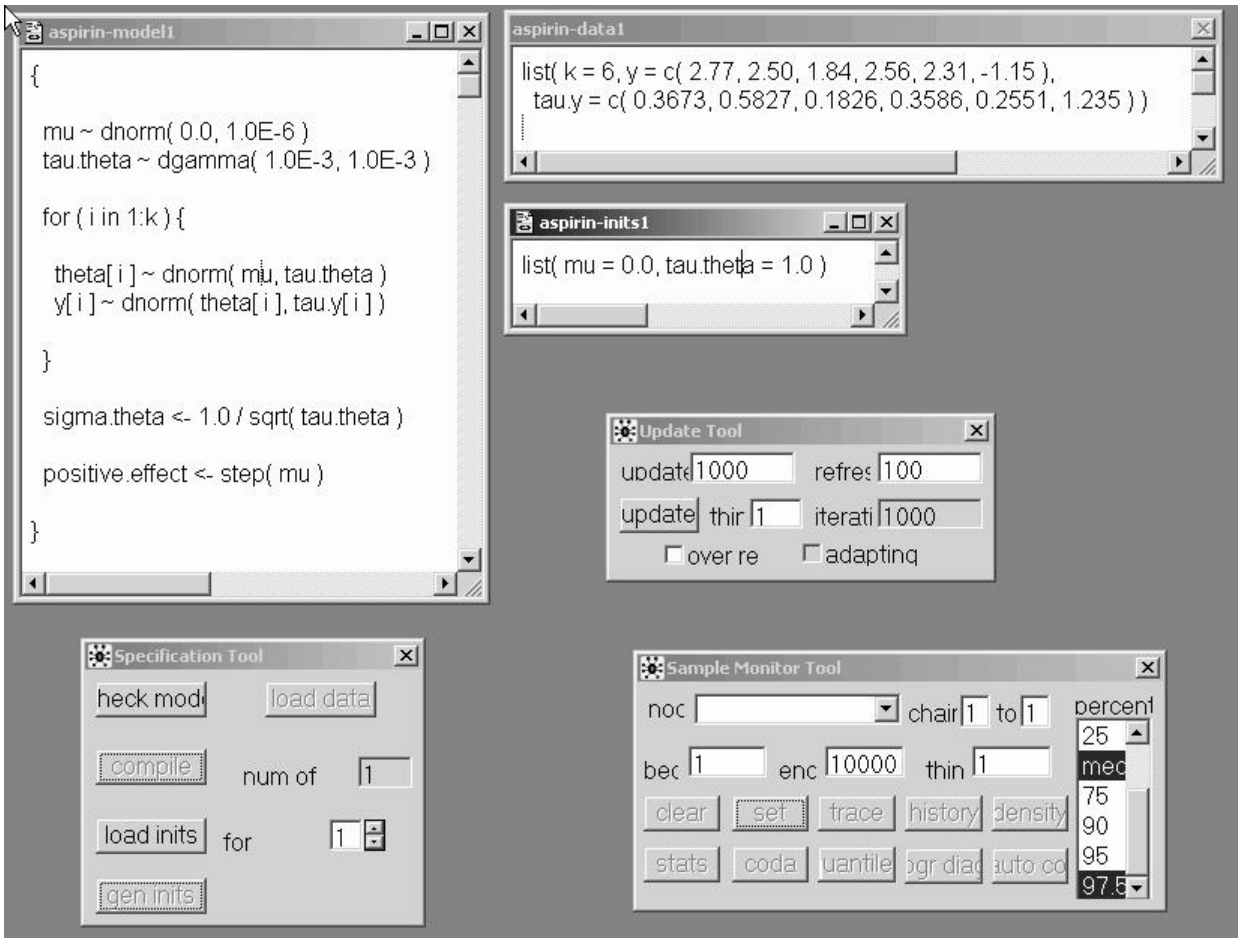
Here, e.g., `tau.y[ 1 ]` $= \frac{1}{1.65^2} \doteq 0.3673$, where 1.65 is the **standard error** of the difference `y[ 1 ]` for experiment 1 in Table 2.1 on p. 20.

Finally, the **initial values** file in the aspirin meta-analysis is

```
list( mu = 0.0, tau.theta = 1.0 )
```

In a simple example like this there's no harm in starting the Markov chain off in a **generic** location: since μ and τ_θ live on $(-\infty, \infty)$ and $(0, \infty)$, convenient generic choices for their starting values are 0 and 1, respectively (more care may be required in models with **more complex random-effects structure**).

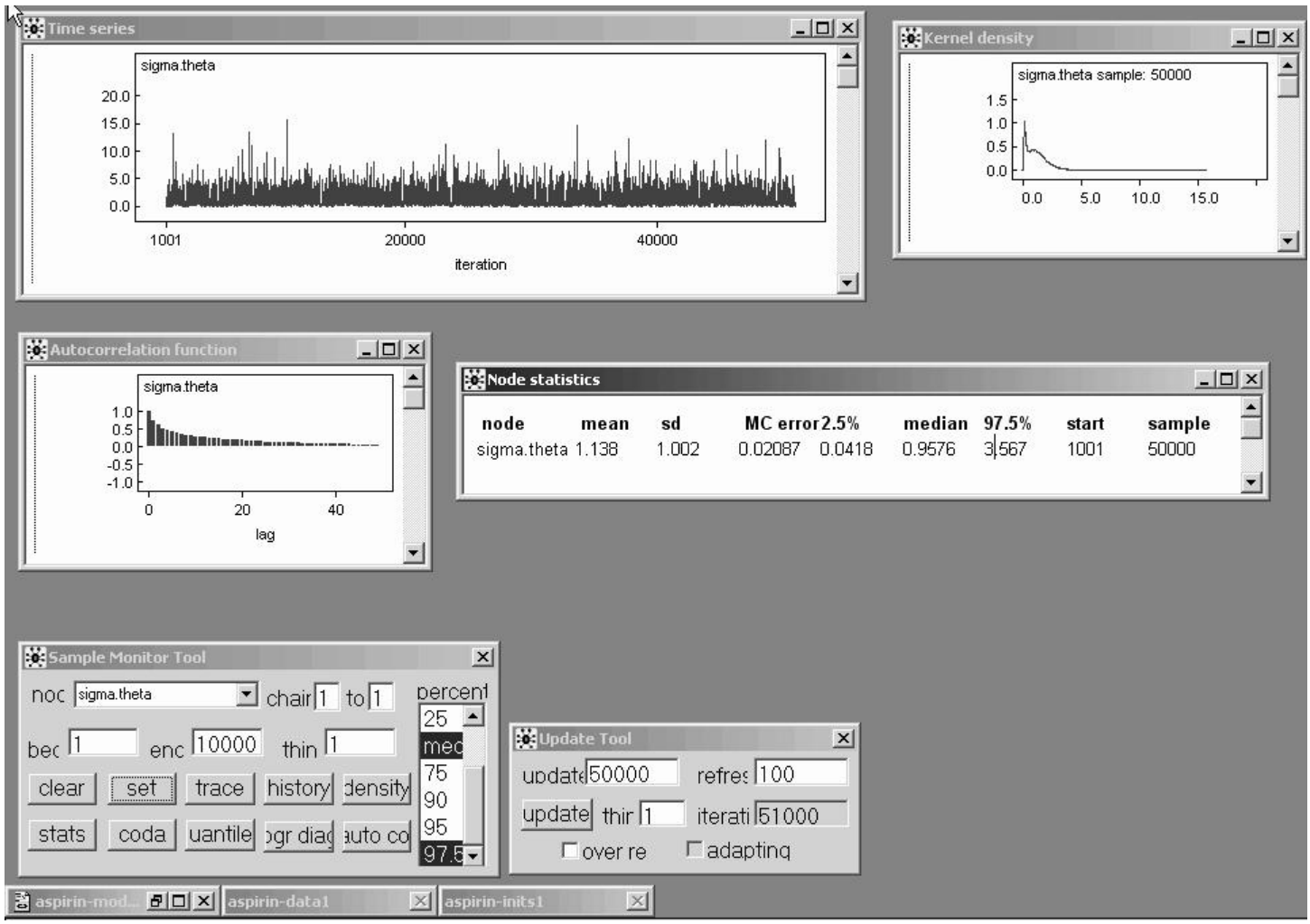
WinBUGS **Aspirin Analysis (continued)**



I (1) get a `Specification Tool` from the `Model` menu, (2) click on the **model** window and click `check model`, (3) click on the **data** window and click `load data` and `compile`, (4) click on the **initial values** window and click `load inits`, and (5) click `gen inits` (because the random effects θ_i were uninitialized in the inits file); I'm now ready to do some MCMC sampling.

I (6) get an `Update Tool` from the `Model` menu, and click `update` to perform a **burn-in** of 1,000 iterations (the default), which takes **0s** at 1.6 Pentium GHz; (7) I then get a `Sample Monitoring Tool` from the `Inference` menu, and type `sigma.theta` and click `set`.

WinBUGS **Aspirin Analysis (continued)**



(8) I type `50000` in the `updates` window in the `Update Tool` and click `update` to get a **monitoring** run of **50,000** iterations (this took **15s**).

Then (9) selecting `sigma.theta` in the node window, all 10 buttons from `clear` through `autoC` are active, and I click on `history` (to get a `Time Series` window), `density` (to get a `Kernel density` window), `autoC` to get an `Autocorrelation function` window, and `stats` (to get a `Node statistics` window), **yielding the screen above**.

The output of an MCMC sampler, when considered as a **time series**, often exhibits **positive autocorrelation**; in fact it often looks like a realization of an **autoregressive** AR_p model of order $p = \mathbf{1}$ ($\theta_t = \alpha + \beta\theta_{t-1} + e_t$) with **positive first-order autocorrelation** ρ.

WinBUGS Aspirin Analysis (continued)

This does not affect the **validity** of Monte Carlo inferences about the unknowns (e.g., the mean of any **stationary stochastic process** is a **consistent** estimator of the underlying process mean), but it does affect the **efficiency** of these inferences: for example, the Monte Carlo variance of the sample mean $\bar{\theta}$ based on M draws from an AR_1 time series is

$$V(\bar{\theta}) = \frac{\sigma_\theta^2}{M}\left(\frac{1+\rho}{1-\rho}\right), \tag{29}$$

and the **sample size inflation factor** $\frac{1+\rho}{1-\rho} \to \infty$ as $\rho \to +1$.

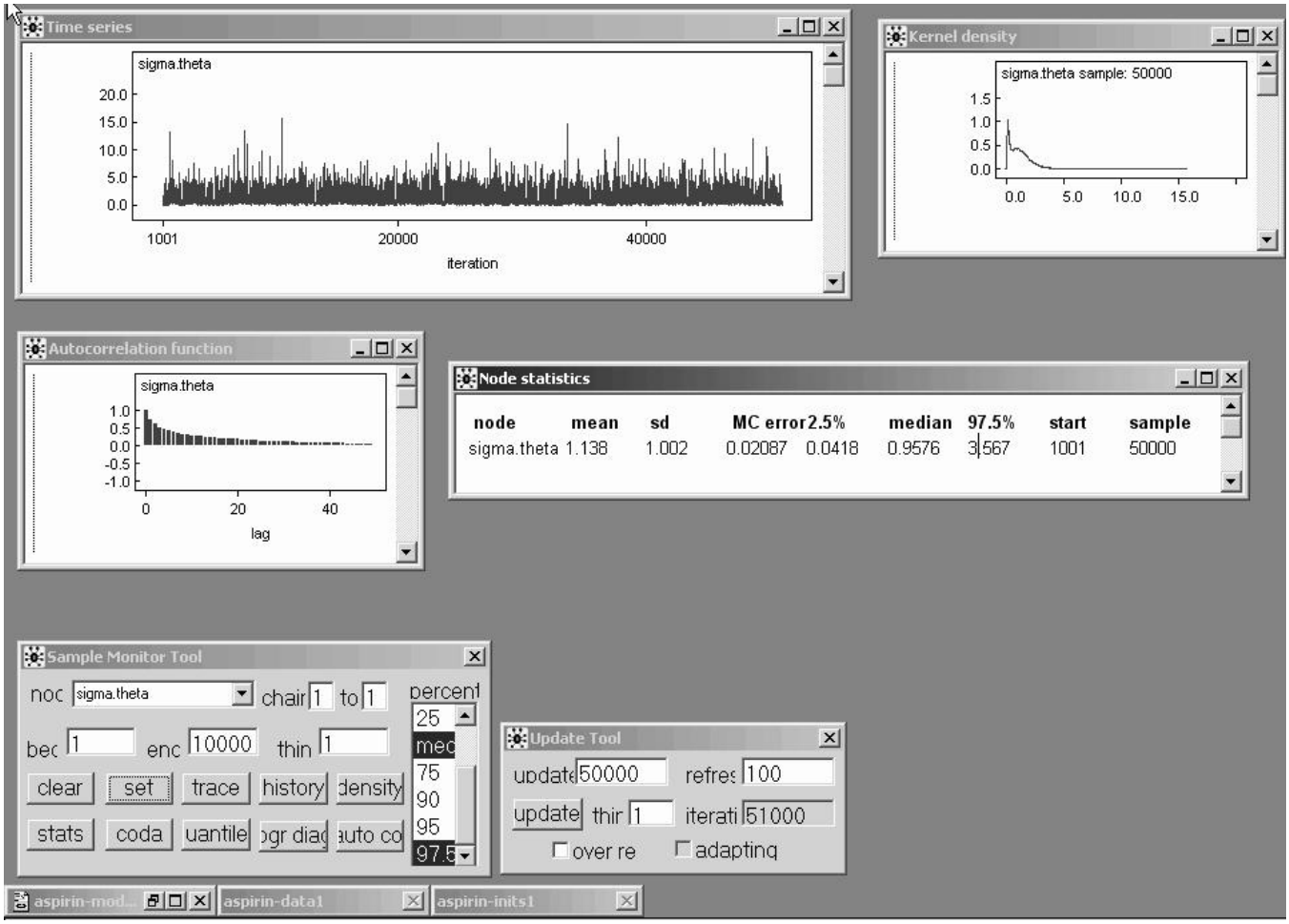
An MCMC sampler which produces output for any given unknown θ with ρ near 0 (if $\rho = 0$ the output is white noise, i.e., equivalent to IID draws from the posterior) is said to be **mixing well** in that unknown.

The time series trace for σ_θ above is only mixing **moderately well**: the autocorrelation function has the familiar ski-slope shape of an AR_1 series with $\rho \doteq 0.7$ (the height of the bar at lag 1).

The **marginal posterior distribution** for σ_θ (from the `Kernel density` window) looks heavily skewed to the right, which makes sense for a scale parameter.

The **posterior mean** and **SD** of σ_θ (using the $\Gamma(\epsilon, \epsilon)$ prior for τ_θ) are estimated to be 1.14 and 1.00, respectively; the **Monte Carlo standard error** of the posterior mean estimate is 0.021 (so that with 50,000 monitoring iterations I don't yet have **3 significant figures** of accuracy for the posterior mean); the **posterior median** is estimated to be 0.96; and a **95% central interval** for σ_θ with this prior is estimated to run from 0.042 to 3.57.

WinBUGS **Aspirin Analysis (continued)**



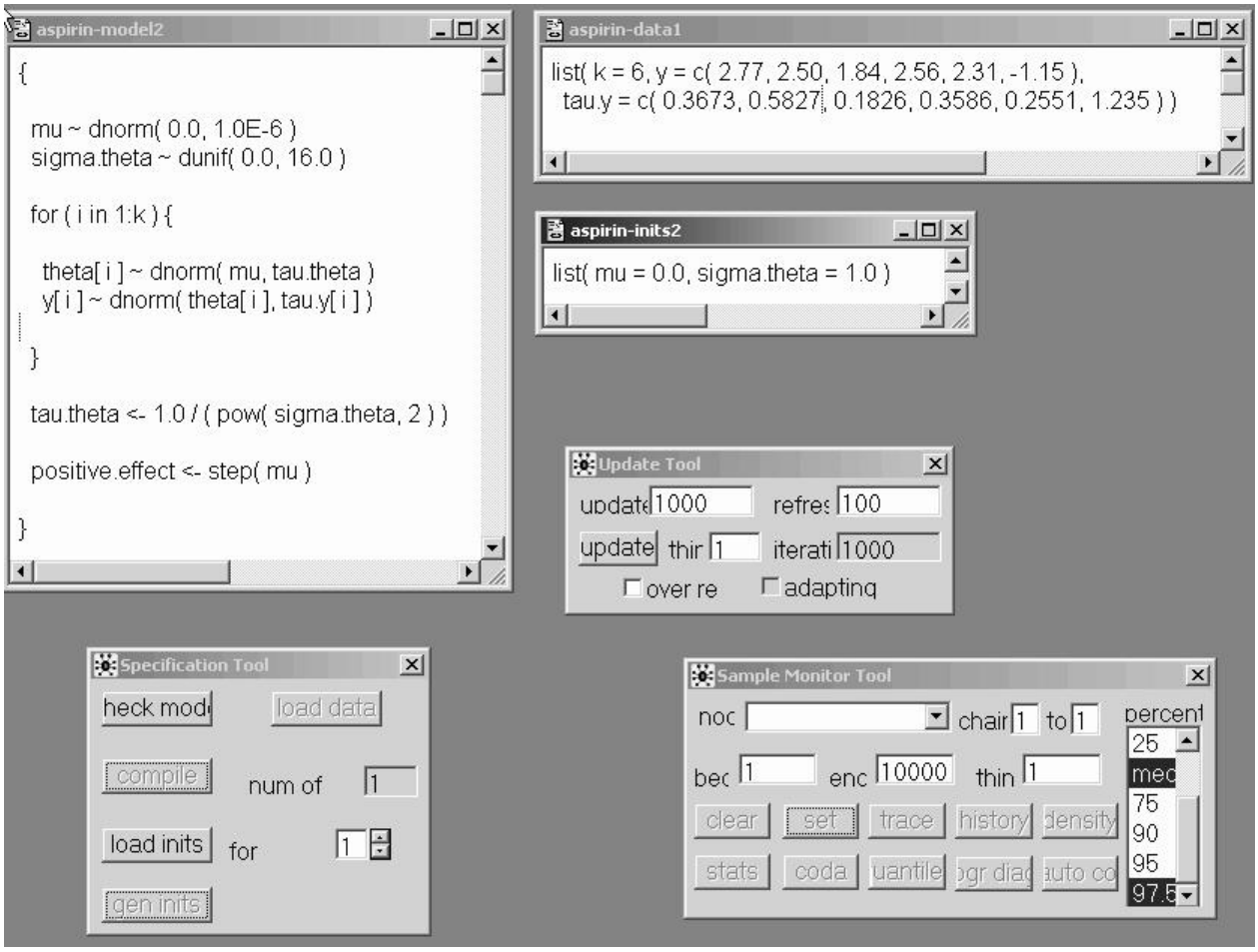
The main thing to notice, however, is that the **range of plausible values** for `sigma.theta` in its posterior is approximately from **0 to 16**.

It has recently been shown that the **simplest diffuse prior** on σ_θ that has **good calibration properties** (i.e., such that 95% **nominal** Bayesian interval estimates for all of the parameters in model (10) do in fact have **actual coverage close to 95%**) is

$$\sigma_\theta \sim U(0, c), \tag{30}$$

where c is chosen to be (roughly) the **smallest value that doesn't truncate the likelihood function** for σ_θ; here it's evident that $c \doteq 16$ will **work well**.

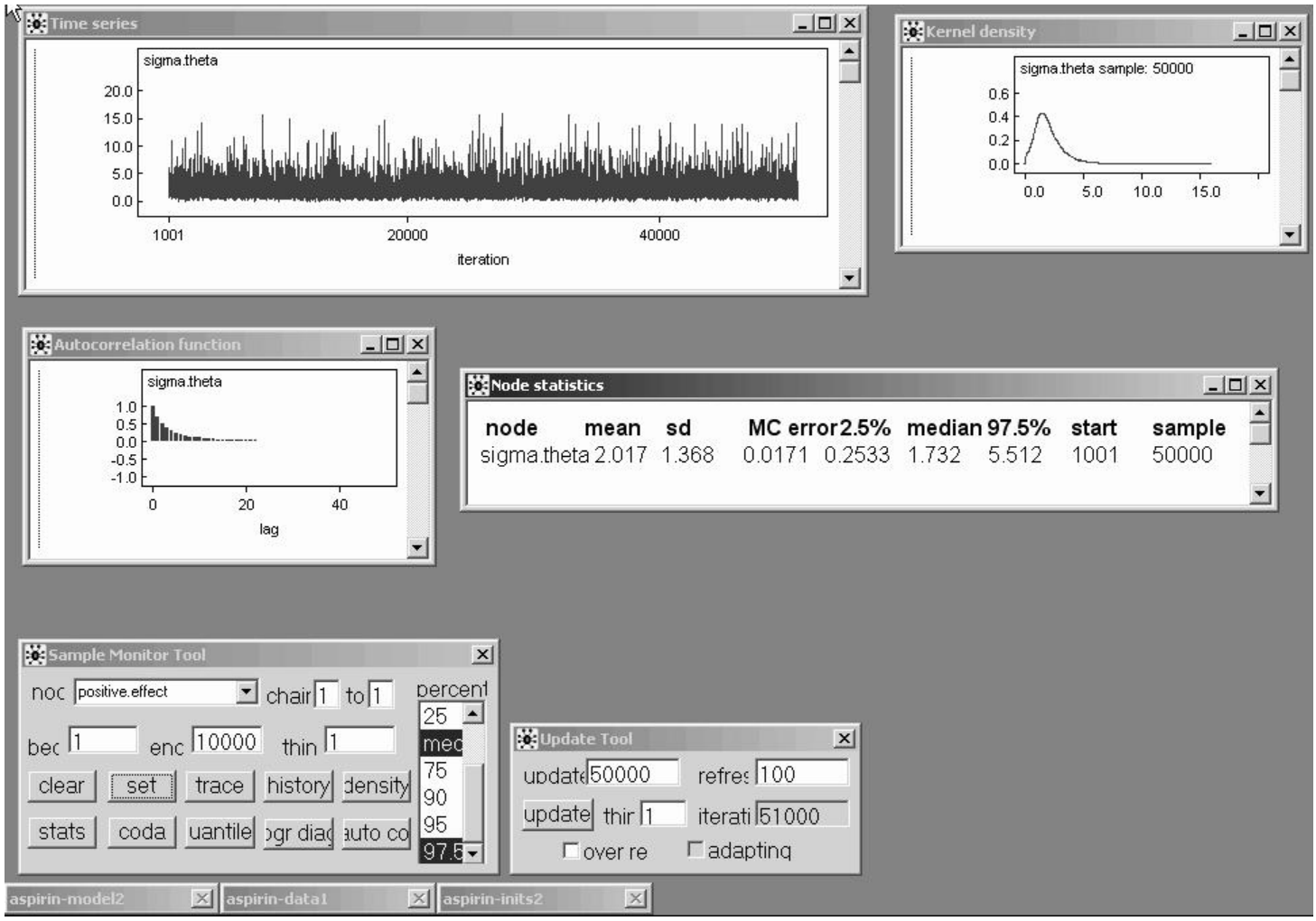
WinBUGS **Aspirin Analysis (continued)**



So I estimate a **second model** placing a Uniform$(0, c)$ prior on σ_θ (this model also requires a **new initial values file** that initializes `sigma.theta` instead of `tau.theta`).

This time in the `Sample Monitor Tool` I `set` all of the **interesting** quantities: `mu`, `sigma.theta`, `theta`, and `positive.effect`, and I use the same MCMC strategy as before (a **burn-in of 1,000** followed by a **monitoring run of 50,000**).

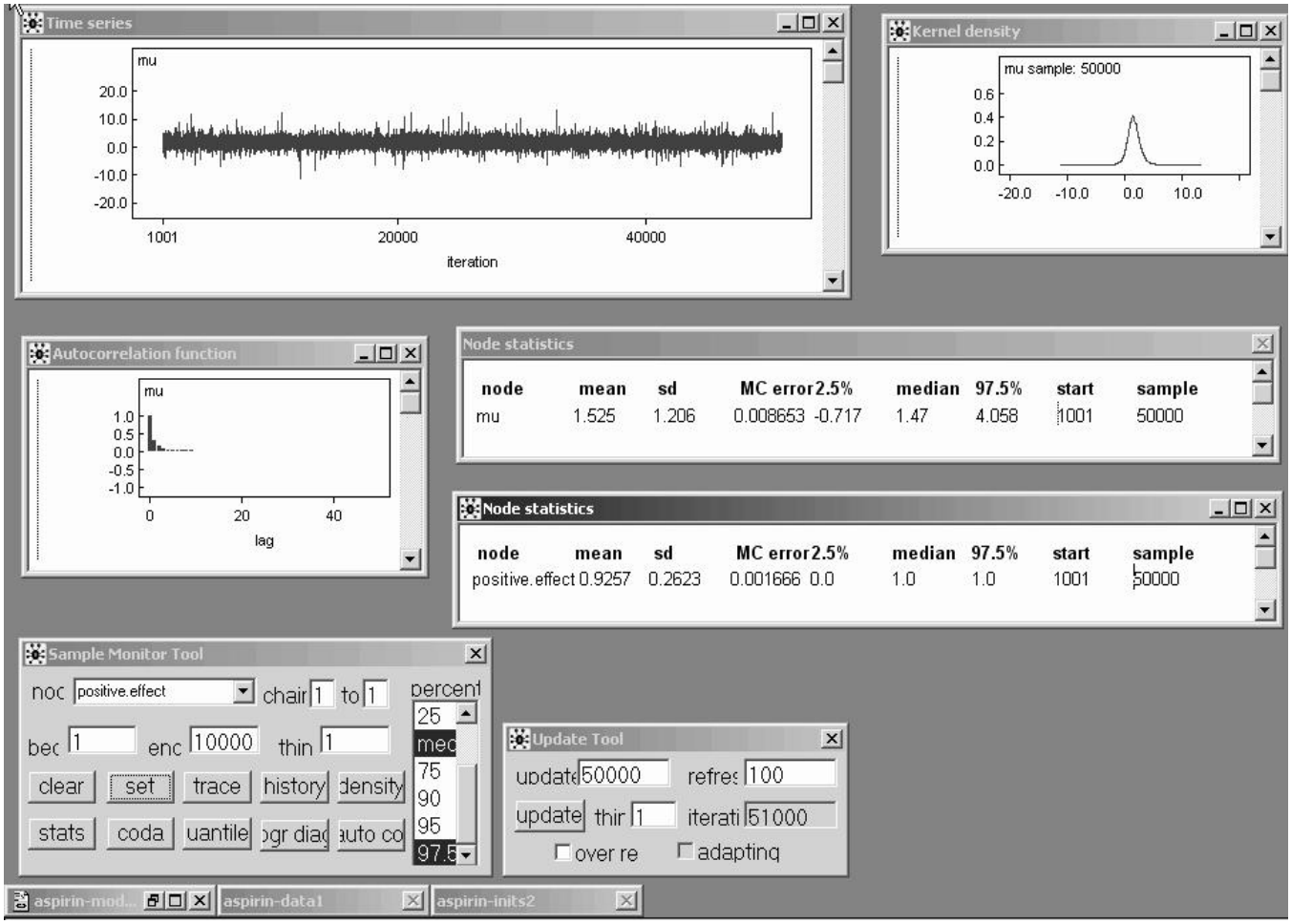
WinBUGS **Aspirin Analysis (continued)**



With the Uniform$(0, c)$ prior on σ_θ the posterior mean of σ_θ is now **sharply higher** than before (**2.02** versus the **1.14** value I got with the initial $\Gamma(\epsilon, \epsilon)$ prior (this sort of **discrepancy** will only arise when the number of studies k is **small**; when it does arise I **trust** the results from the Uniform$(0, c)$ prior).

Note that the posterior mean of σ_θ is also **quite a bit bigger** than the value (**1.24**) obtained from **MLEB** back on page 25—this is a reflection of the **tendency of MLEB to understate the between-study heterogeneity** in model (10) with small k.

WinBUGS **Aspirin Analysis (continued)**



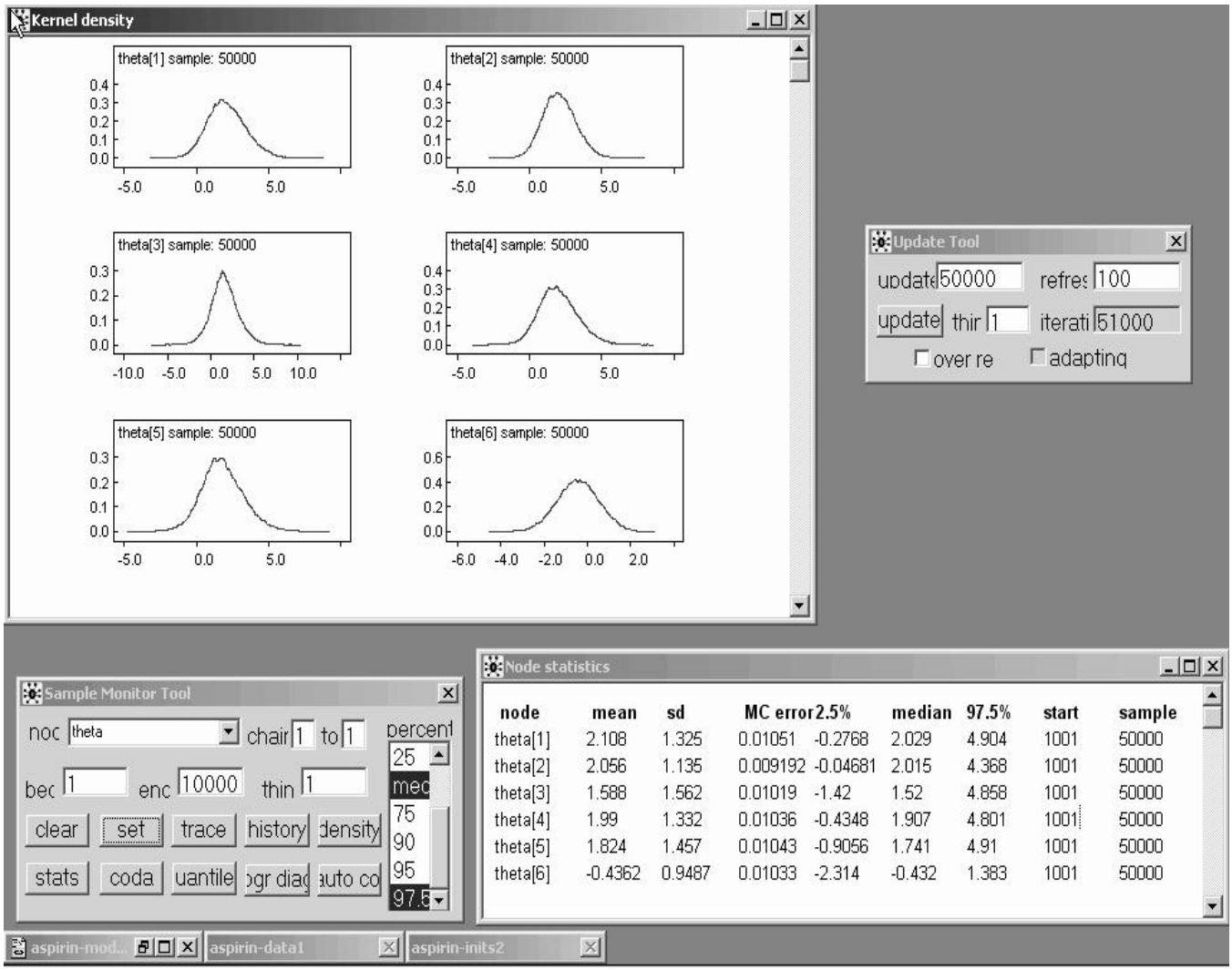
On pp. 25–26 above we saw that the MLEB estimate of μ was **1.45** with an approximate standard error of **0.809**, and an approximate 95% confidence interval for μ ran from -0.14 to $+3.03$.

The corresponding **Bayesian** results are: posterior **mean 1.52**, posterior **SD 1.21**, 95% **interval (–0.72, 4.06)**.

As is often true, the simple MLEB approximations leading to these estimates have **underestimated the actual uncertainty** about μ: the Bayesian 95% interval with the Uniform prior is **50% wider**.

It's easy to monitor the **posterior probability that aspirin is beneficial**, with the built-in `step` function applied to `mu`: $P(\mu > 0|\text{data, diffuse prior information}) \doteq$ **0.93**, i.e., posterior betting odds of about **12.5 to 1 that aspirin reduces mortality**.

WinBUGS **Aspirin Analysis (continued)**



The marginal density plots of the θ_i values show **interesting departures from normality**, and the Bayesian estimates (a) exhibit **rather less shrinkage** and (b) have **27–43% larger** uncertainty estimates.

Table 3.1. MLEB and Bayesian (posterior mean) estimates of the θ_i.

	Maximum Likelihood		Bayesian Posterior	
study(i)	$\hat{\theta}_i$	$\widehat{SE}(\hat{\theta}_i)$	mean	SD
1	1.92	0.990	2.11	1.33
2	1.94	0.899	2.06	1.14
3	1.53	1.09	1.59	1.56
4	1.84	0.994	1.99	1.33
5	1.69	1.05	1.82	1.46
6	−0.252	0.728	−0.44	0.95

Hierarchical Model Expansion

Looking at the **shrinkage plot** on p. 26 or the **raw data values** themselves, it's evident that a **Gaussian** model for the θ_i may not be appropriate: study 6 is so different than the other 5 that a **heavier-tailed distribution** may be a better choice.

This suggests **expanding** the HM (10), by embedding it in a **richer model class** of which it's a **special case** (this is the main Bayesian approach in practice to **dealing with model inadequacies**).

A **natural choice** would be a t model for the θ_i with **unknown degrees of freedom** ν:

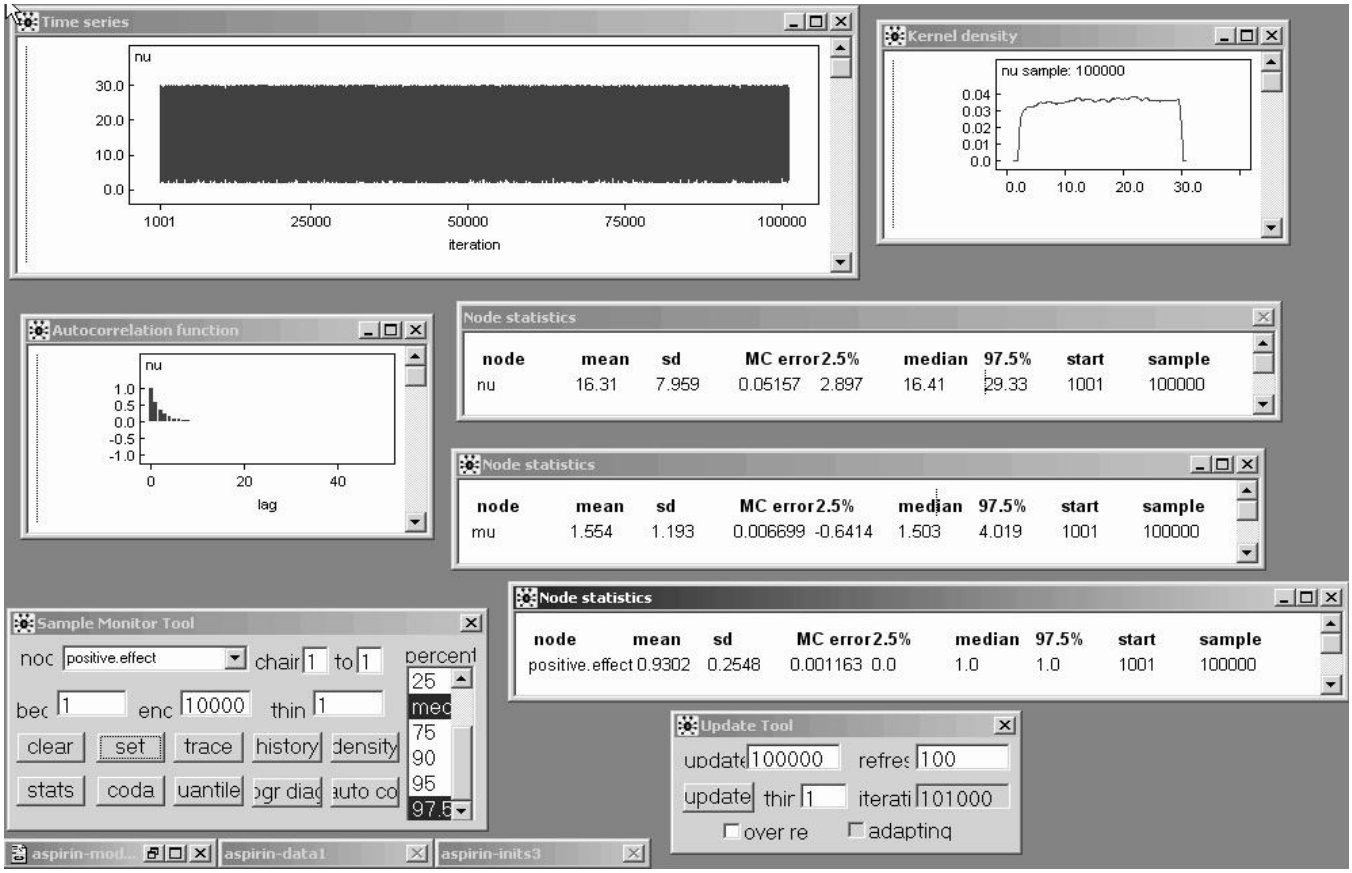
$$\begin{array}{rcll} (\theta, \sigma^2, \nu) & \sim & p(\theta, \sigma^2, \nu) & \text{(prior)} \\ (\theta_i | \theta, \sigma^2, \nu) & \overset{\text{IID}}{\sim} & t\big(\theta, \sigma^2, \nu\big) & \text{(underlying effects)} \\ (y_i | \theta_i) & \overset{\text{indep}}{\sim} & N(\theta_i, V_i) & \text{(data)} \ . \end{array} \tag{31}$$

Here $\eta \sim t\big(\theta, \sigma^2, \nu\big)$ just means that $\left(\frac{\eta - \theta}{\sigma}\right)$ follows a **standard** t **distribution** with ν degrees of freedom. This is **amazingly easy** to implement in `WinBUGS` (it is considerably more difficult to carry out an **analogous ML analysis**).

The new model file is

```
{

  mu ~ dnorm( 0.0, 1.0E-6 )
  sigma.theta ~ dunif( 0.0, 16.0 )
  nu ~ dunif( 3.0, 30.0 )

  for ( i in 1:k ) {

    theta[ i ] ~ dt( mu, tau.theta, nu )
    y[ i ] ~ dnorm( theta[ i ], tau.y[ i ] )

  }

  tau.theta <- 1.0 / pow( sigma.theta, 2 )

}
```

Model Expansion (continued)



To express comparative prior ignorance about ν I use a **uniform** prior on the interval from 2.0 to 30.0 (below $\nu = 2$ the t distribution has **infinite variance**, and above about 30 it starts to be **indistinguishable** in practice from the Gaussian).

A **burn-in** of 1,000 and a **monitoring run** of 100,000 iterations takes **about twice as long as with 50,000 iterations in the Gaussian model** (i.e., about the **same speed per iteration**) and yields the **posterior summaries** above.

It's clear that there's **little information in the likelihood function** about ν: the prior and posterior for this parameter **virtually coincide**.

The results for μ and the θ_i are **almost unchanged**; this would not necessarily be the case if study 6 had been **more extreme**.

Educational Meta-Analysis

Incorporating Study-Level Covariates

Case Study: *Meta-analysis of the effect of teacher expectancy on student IQ* (Bryk and Raudenbush, 1992).
Do **teachers' expectations** influence students' intellectual development,
as measured by IQ scores?

Table 5.4. Results from 19 experiments estimating the effects of teacher expectancy on pupil IQ.

Study (i)	Weeks of Prior Contact (x_i)	Estimated Effect Size (y_i)	Standard Error of $y_i = \sqrt{V_i}$
1. Rosenthal et al. (1974)	2	0.03	0.125
2. Conn et al. (1968)	3	0.12	0.147
3. Jose & Cody (1971)	3	-0.14	0.167
4. Pellegrini & Hicks (1972)	0	1.18	0.373
5. Pellegrini & Hicks (1972)	0	0.26	0.369
6. Evans & Rosenthal (1969)	3	-0.06	0.103
7. Fielder et al. (1971)	3	-0.02	0.103
8. Claiborn (1969)	3	-0.32	0.220
9. Kester & Letchworth (1972)	0	0.27	0.164
10. Maxwell (1970)	1	0.80	0.251
11. Carter (1970)	0	0.54	0.302
12. Flowers (1966)	0	0.18	0.223
13. Keshock (1970)	1	-0.02	0.289
14. Henrickson (1970)	2	0.23	0.290
15. Fine (1972)	3	-0.18	0.159
16. Greiger (1970)	3	-0.06	0.167
17. Rosenthal & Jacobson (1968)	1	0.30	0.139
18. Fleming & Anttonen (1971)	2	0.07	0.094
19. Ginsburg (1970)	3	-0.07	0.174

Teacher Expectancy

Raudenbush (1984) found $k = 19$ experiments, published between 1966 and 1974, estimating **the effect of teacher expectancy on student IQ** (Table 5.4).

In each case the experimental group was made up of children for whom teachers were (**deceptively**) encouraged to have high expectations (e.g., experimenters gave treatment teachers lists of students, **actually chosen at random**, who allegedly displayed dramatic potential for intellectual growth), and the controls were students about whom no particular expectations were encouraged.

The estimated **effect sizes** $y_i = \frac{\bar{T}_i - \bar{C}_i}{\text{SD}_{i:\text{pooled}}}$ (column 3 in Table 5.4) ranged from $-.32$ to $+1.18$; why?

One good reason: the studies differed in **how well the experimental teachers knew their students** at the time they were given the deceptive information—this time period x_i (column 2 in Table 5.4) ranged from 0 to 3 weeks.

Figure 5.2 plots y_i against x_i—you can see that **the studies with bigger x_i had smaller IQ effects on average**.

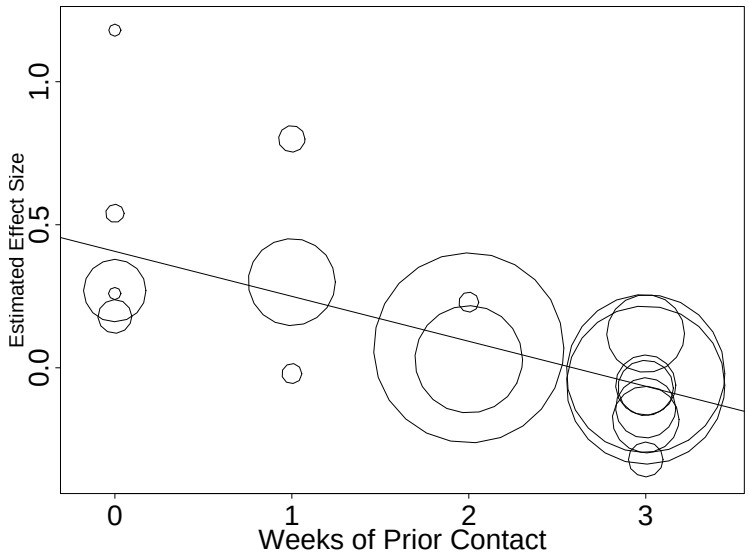


Figure 5.2. Scatterplot of estimated effect size against weeks of prior contact in the IQ meta-analysis. Radii of circles are proportional to $w_i = V_i^{-1}$ (see column 4 in Table 5.4); fitted line is from weighted regression of y_i on x_i with weights w_i.

Conditional Exchangeability

Evidently model (1) will not do here — it says that your predictive uncertainty about all the studies is **exchangeable** (similar, i.e., according to (1) the underlying study-level effects θ_i are like IID draws from a normal distribution), whereas Figure 5.2 **clearly shows** that the x_i are useful in predicting the y_i.

This is another way to say that your uncertainty about the studies is **not unconditionally exchangeable** but

conditionally exchangeable given x

(Draper et al., 1993b).

In fact Figure 5.2 suggests that the y_i (and therefore the θ_i) are related **linearly** to the x_i.

Bryk and Raudenbush, working in the **frequentist** paradigm, fit the following HM to these data:

$$\begin{aligned} (\theta_i|\alpha,\beta,\sigma_\theta^2) &\overset{\text{indep}}{\sim} N\left(\alpha+\beta\, x_i, \sigma_\theta^2\right) \quad \text{(underlying effects)} \\ (y_i|\theta_i) &\overset{\text{indep}}{\sim} N(\theta_i, V_i) \quad \text{(data).} \end{aligned} \tag{32}$$

According to this model the estimated effect sizes y_i are like **draws from a Gaussian** with mean θ_i and variance V_i, the squared standard errors from column 4 of Table 5.4—here as in model (1) the V_i are taken to be known—and the θ_i themselves are like **draws from a Gaussian** with mean $\alpha+\beta x_i$ and variance σ_θ^2.

The top level of this HM in effect assumes, e.g., that the 5 studies with $x=0$ are sampled **representatively** from {all possible studies with $x=0$}, and similarly for the other values of x.

This (and the Gaussian choice on the top level) are **conventional assumptions, not automatically scientifically reasonable**—for example, if you know of some way in which (say) two of the studies with $x=3$ differ from each other that's **relevant** to the outcome of interest, then you should **include** this in the model as a study-level covariate along with x.

An MLEB Drawback

Bryk and Raudenbush used MLEB methods, based on the **EM algorithm**, to fit this model.

As in Section 5.2, this estimation method combines the two levels of model (9) to construct a **single likelihood** for the y_i, and then **maximizes** this likelihood as usual in the ML approach.

They obtained $(\widehat{\alpha}, \widehat{\beta}) = (.407 \pm .087, -.157 \pm .036)$ and $\widehat{\sigma_\theta}^2 = 0$, naively indicating that **all of the study-level variability** has been "explained" by the covariate x.

However, from a **Bayesian** point of view, this model is **missing a third layer**:

$$\begin{array}{rcl} (\alpha, \beta, \sigma_\theta^2) & \sim & p(\alpha, \beta, \sigma_\theta^2) \\ (\theta_i | \alpha, \beta, \sigma_\theta^2) & \overset{\text{indep}}{\sim} & N\big(\alpha + \beta(x_i - \bar{x}), \sigma_\theta^2\big) \\ (y_i | \theta_i) & \overset{\text{indep}}{\sim} & N(\theta_i, V_i)\,. \end{array} \tag{33}$$

(it will help **convergence** of the sampling-based MCMC methods to make α and β uncorrelated by **centering** the x_i at 0 rather than at $\bar{x}$).

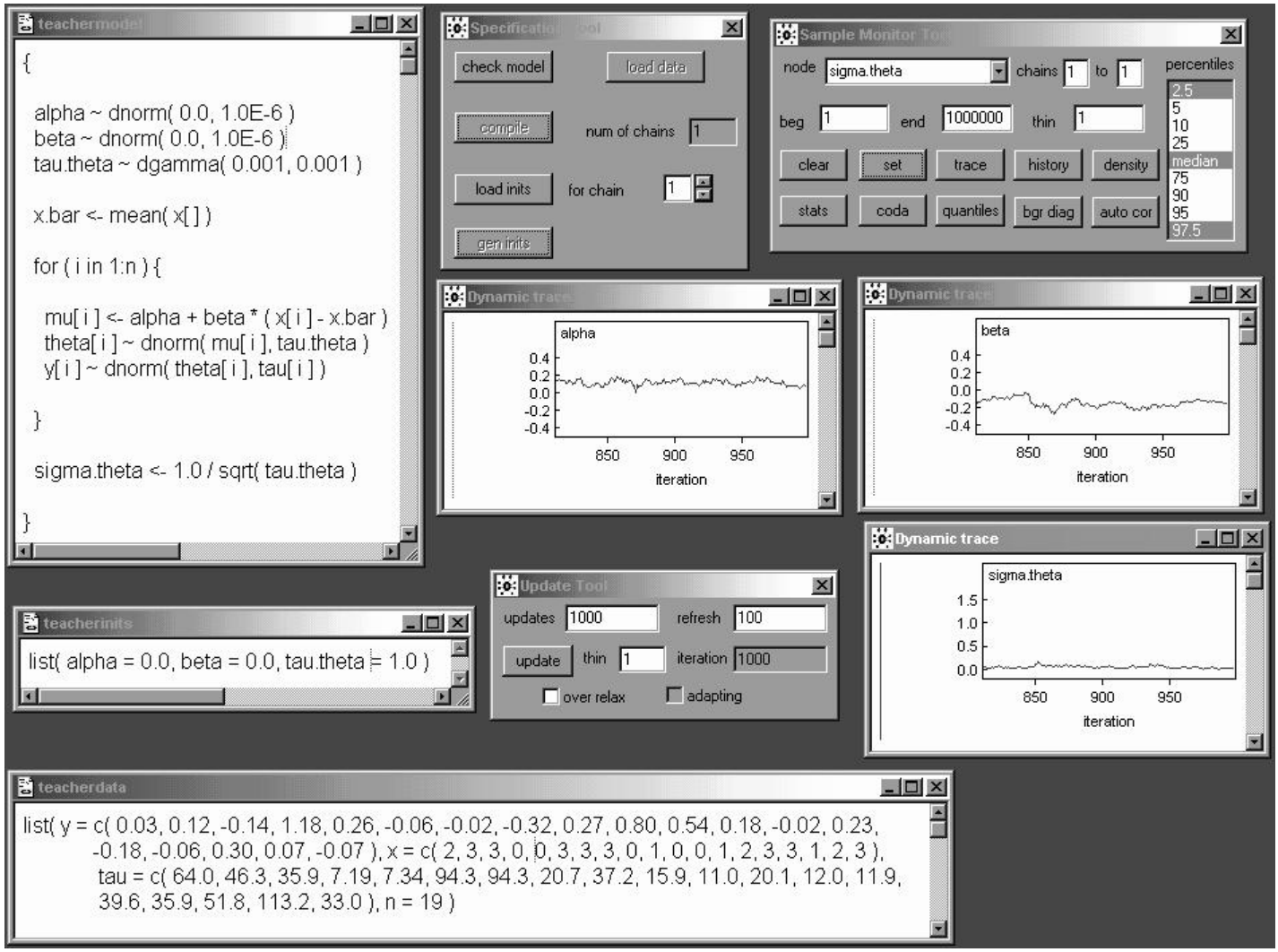
As will subsequently become clear, the trouble with MLEB is that in Bayesian language **it assumes in effect that the posterior for σ_θ^2 is point-mass on the MLE**. This is bad (e.g., Morris, 1983) for two reasons:

- If the posterior for σ_θ^2 is highly **skewed**, the mode will be a **poor summary**; and

- Whatever point-summary you use, pretending the posterior SD for σ^2 is zero **fails to propagate uncertainty** about σ_θ^2 through to uncertainty about α, β, and the θ_i.

The best way to carry out a fully Bayesian analysis of model (10) is with **MCMC** methods.

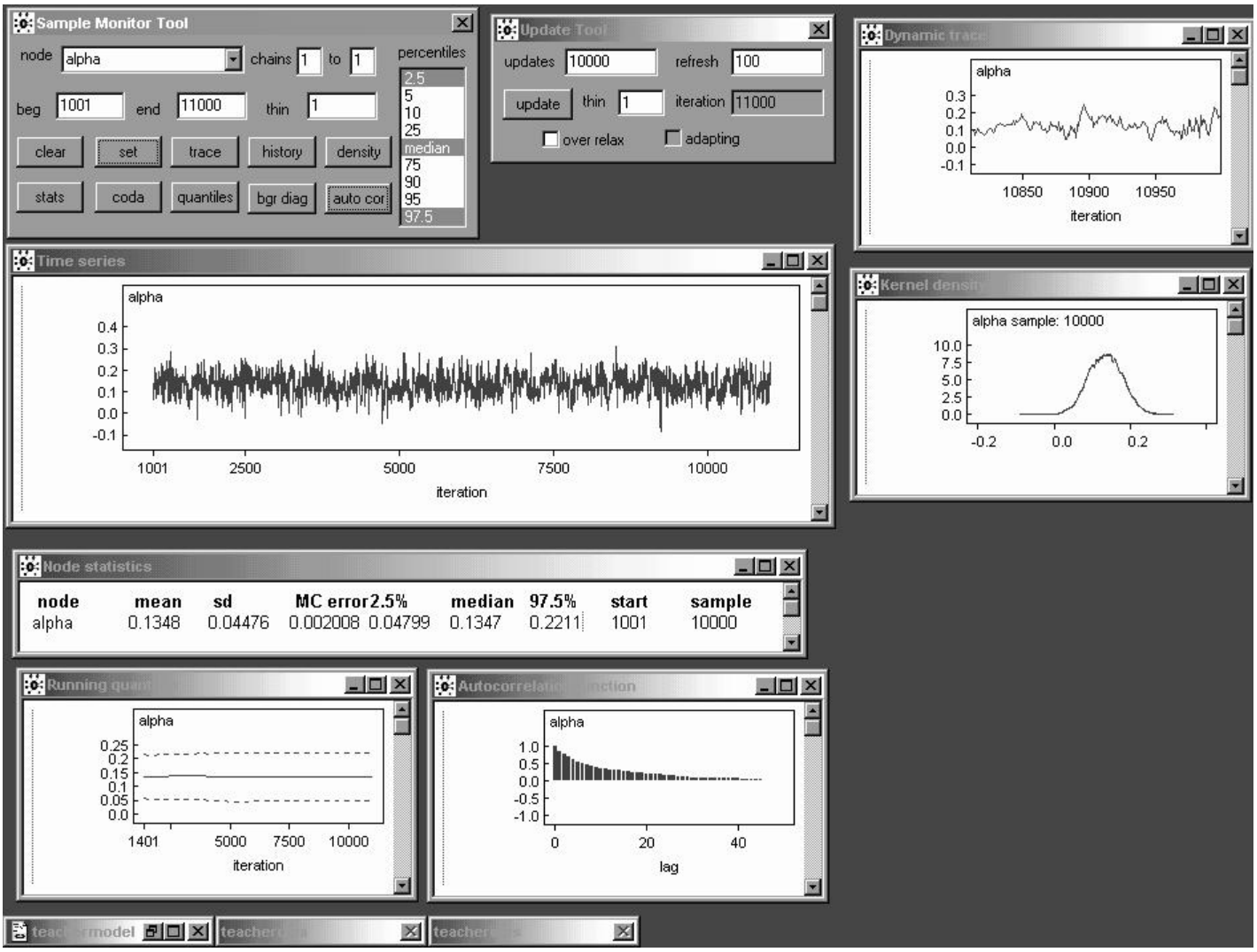
WinBUGS Implementation



For $p(\alpha, \beta, \sigma_\theta^2)$ in model (10) I've chosen the **usual** WinBUGS **diffuse prior** $p(\alpha)p(\beta)p(\sigma_\theta^2)$: since α and β live on the whole real line I've taken marginal Gaussian priors for them with mean 0 and precision 10^{-6}, and since $\tau_\theta = \frac{1}{\sigma^2}$ is positive I use a $\Gamma(0.001, 0.001)$ prior for it.

Model (10) treats the variances V_i of the y_i as **known** (and equal to the squares of column 4 in Table 5.4); I've **converted these into precisions** in the data file (e.g., $\tau_1 = \frac{1}{0.125^2} = 64.0$).

WinBUGS Implementation

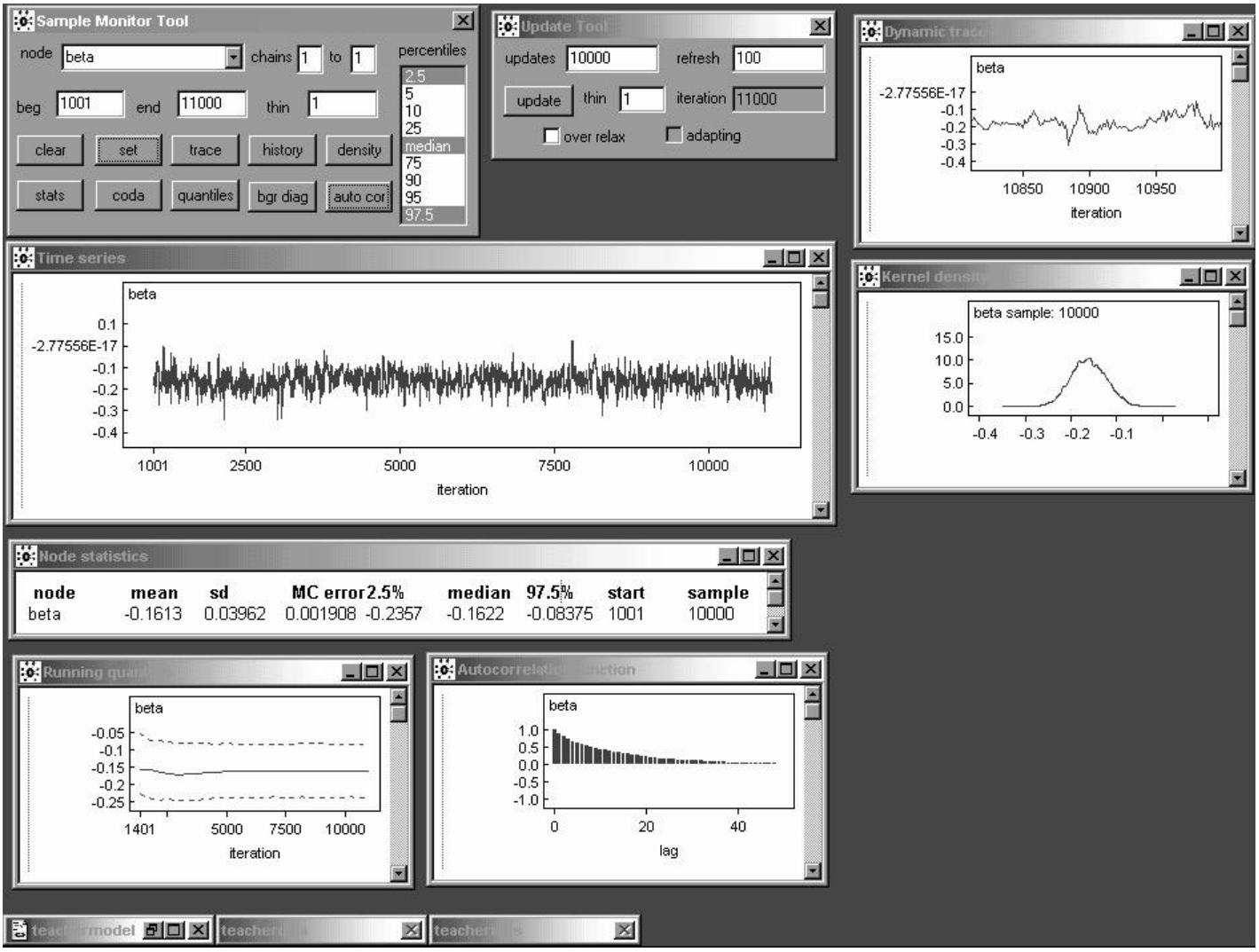


A burn-in of 1,000 (**certainly longer than necessary**) from **default** initial values $(\alpha, \beta, \tau_\theta) = (0.0, 0.0, 1.0)$ and a monitoring run of 10,000 yield the following **preliminary MCMC results**.

Because this is a **random-effects model** we don't expect anything like IID mixing: the output for α behaves like an AR_1 time series with $\widehat{\rho}_1 \doteq 0.86$.

The posterior mean for α, 0.135 (with an MCSE of 0.002), shows that α in model (10) and α in model (9) are **not comparable** because of the **recentering** of the predictor x in model (10): the MLE of α in (9) was 0.41 ± 0.09.

WinBUGS Implementation

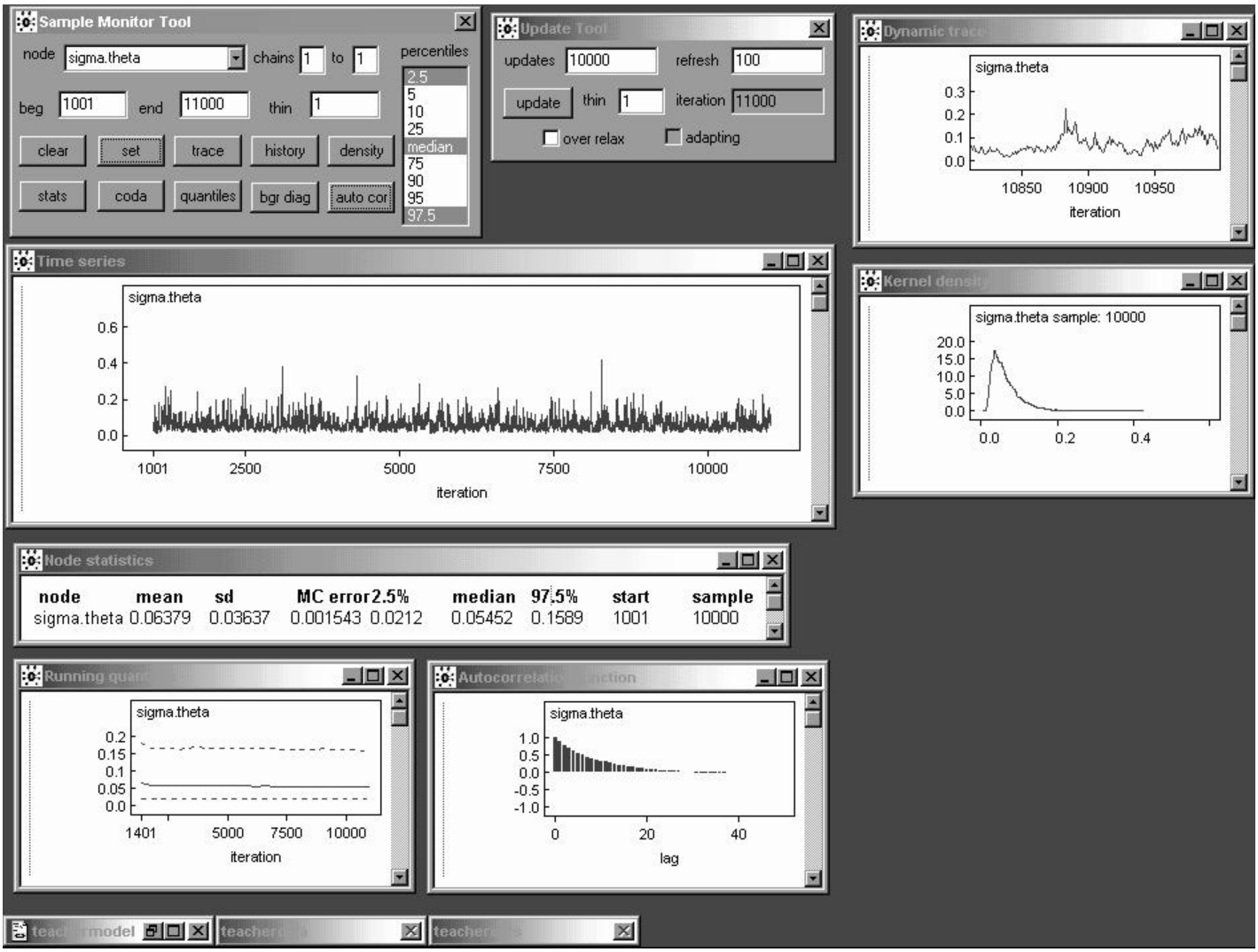


But β **means the same thing in both models** (9) and (10): its posterior mean in (10) is -0.161 ± 0.002, which is not far from the MLE -0.157.

Note, however, that the posterior SD for β, 0.0396, is **10% larger** than the standard error of the maximum likelihood estimate of β (0.036).

This is a reflection of the **underpropagation of uncertainty** about σ_θ in maximum likelihood mentioned on page 15.

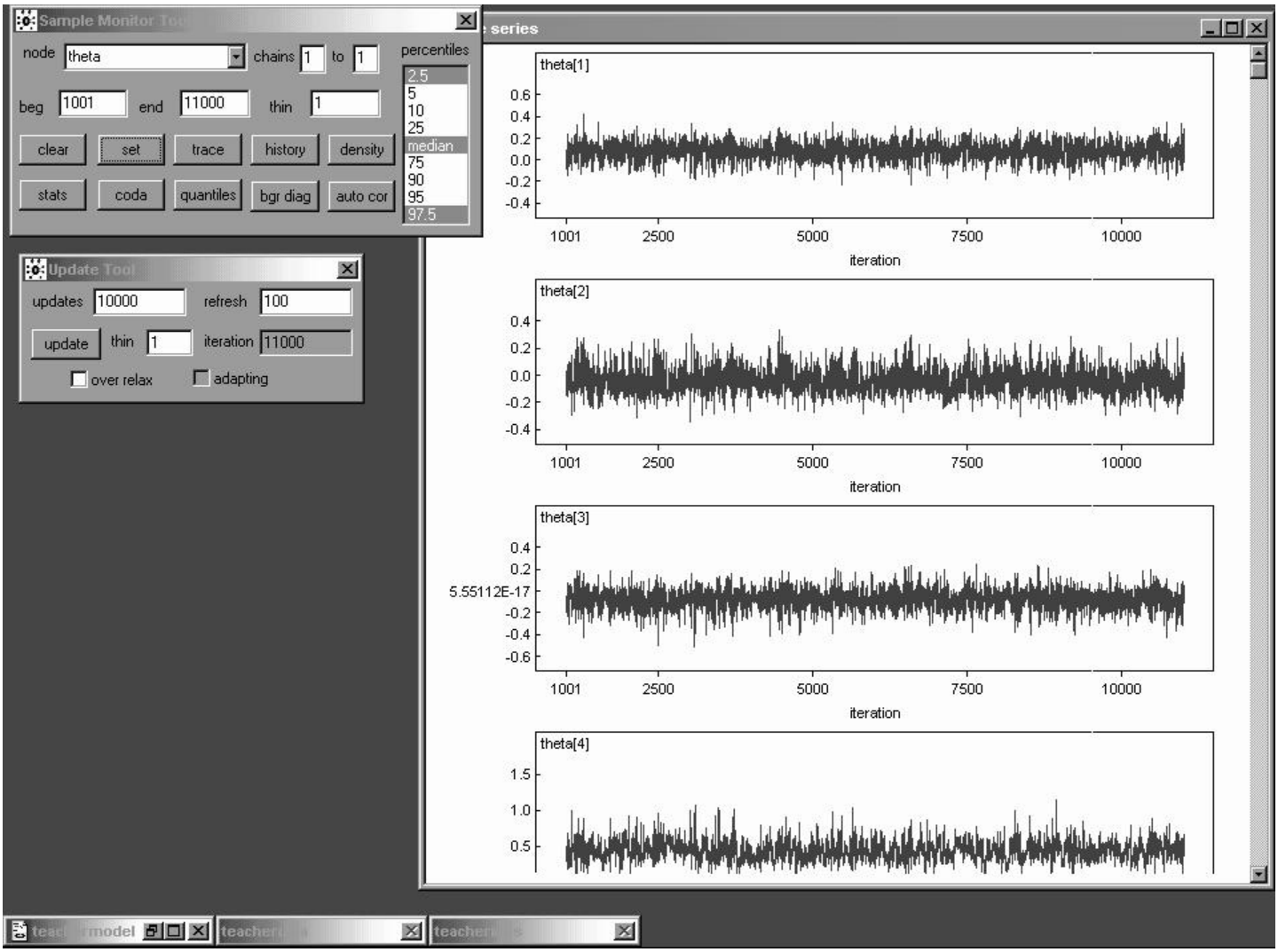
WinBUGS Implementation



In these preliminary results σ_θ has posterior mean 0.064 ± 0.002 and SD 0.036, providing **clear evidence** that the MLE $\hat{\sigma}_\theta = 0$ is a **poor summary**.

Note, however, that the likelihood for σ_θ may be **appreciable in the vicinity of 0** in this case, meaning that some **sensitivity analysis** with diffuse priors other than $\Gamma(0.001, 0.001)$—such as $U(0, c)$ for c around 0.5—would be in order.

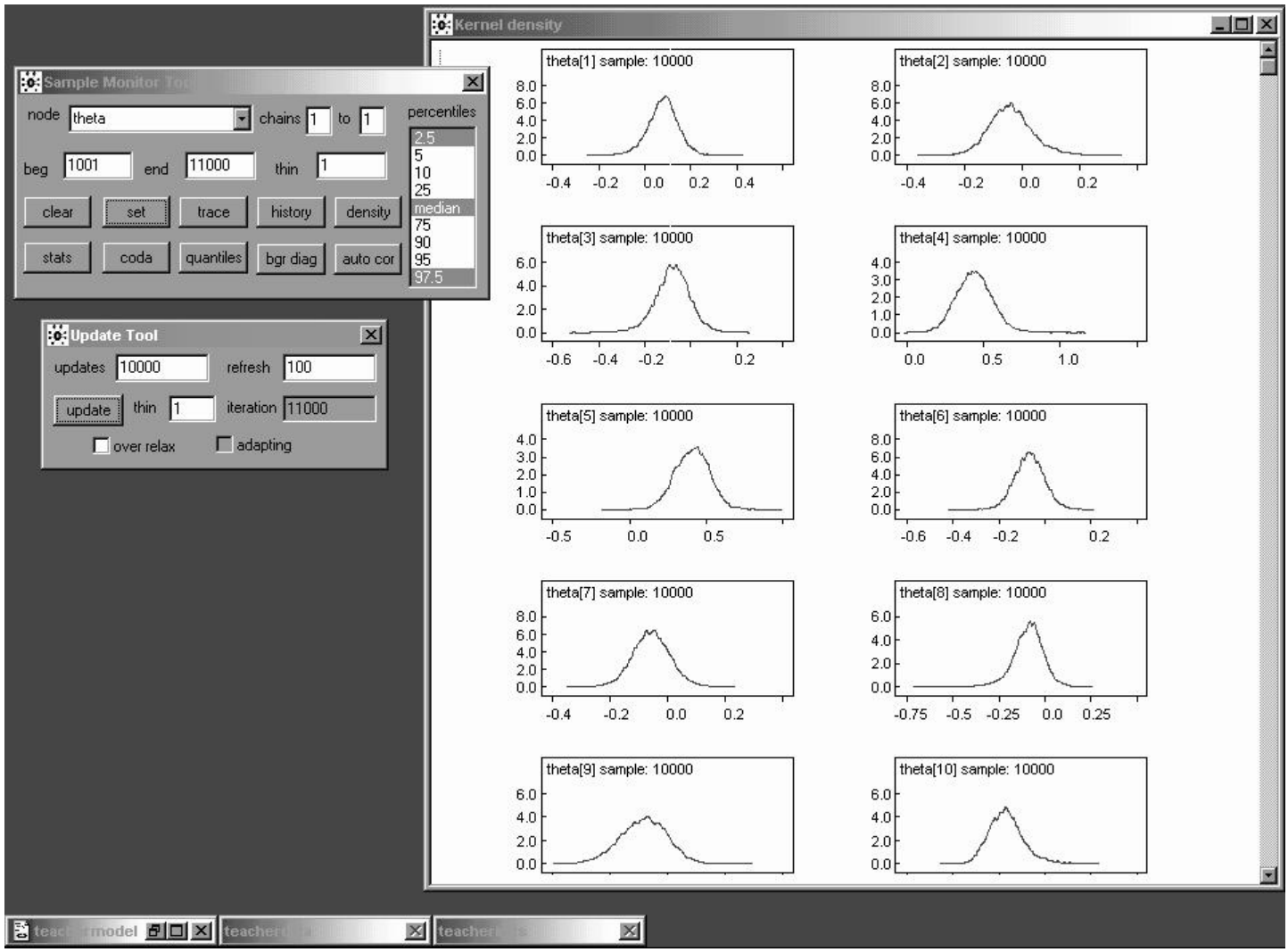
WinBUGS Implementation



When you specify node `theta` in the `Sample Monitor Tool` and then look at the results, you see that `WinBUGS` presents **parallel findings with a single click** for all elements of the vector θ.

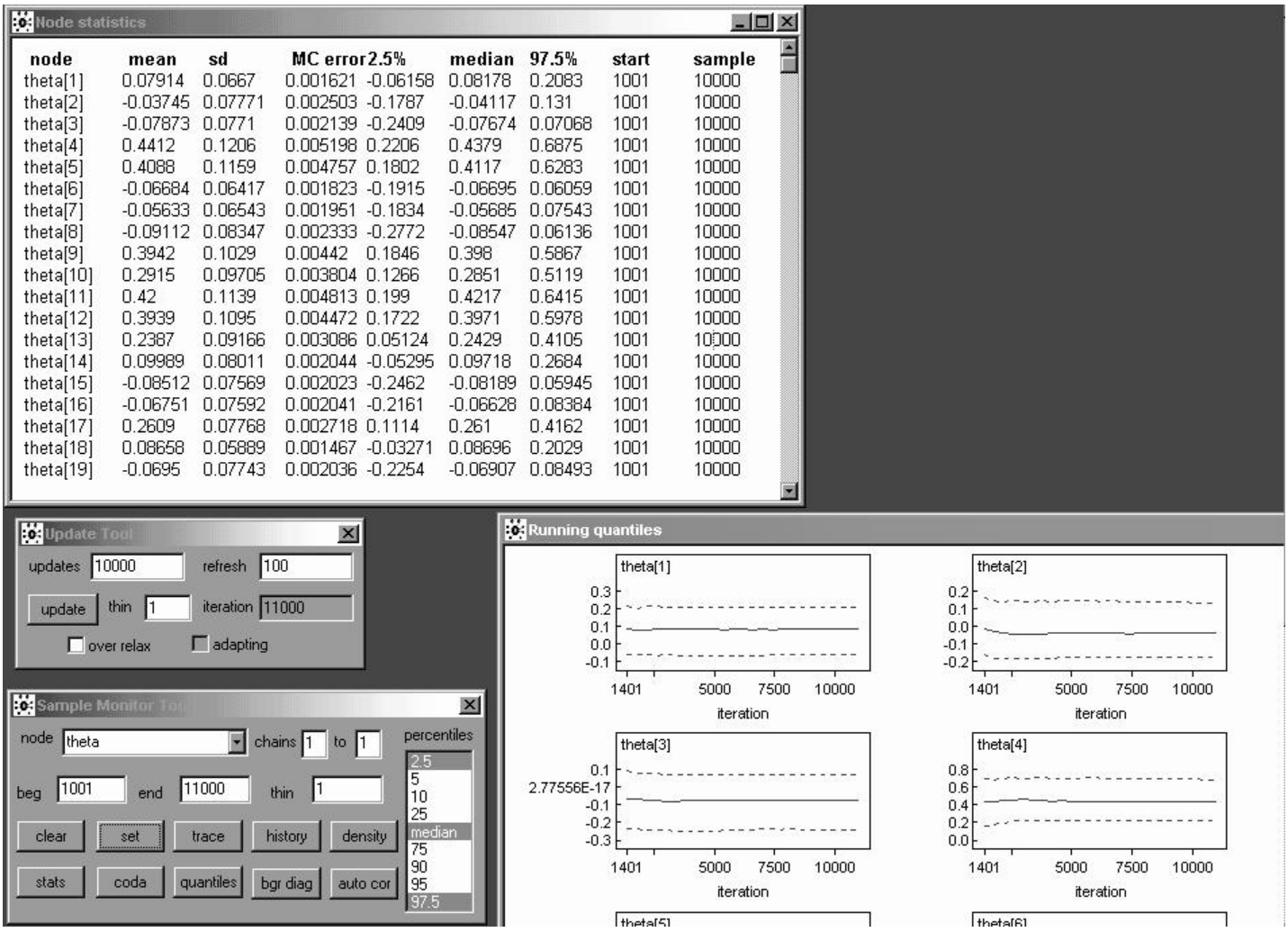
Some of the θ_i are evidently
mixing better than others.

WinBUGS Implementation



The **marginal density traces** of the θ_i look rather like t distributions with fairly low degrees of freedom (**fairly heavy tails**).

WinBUGS Implementation

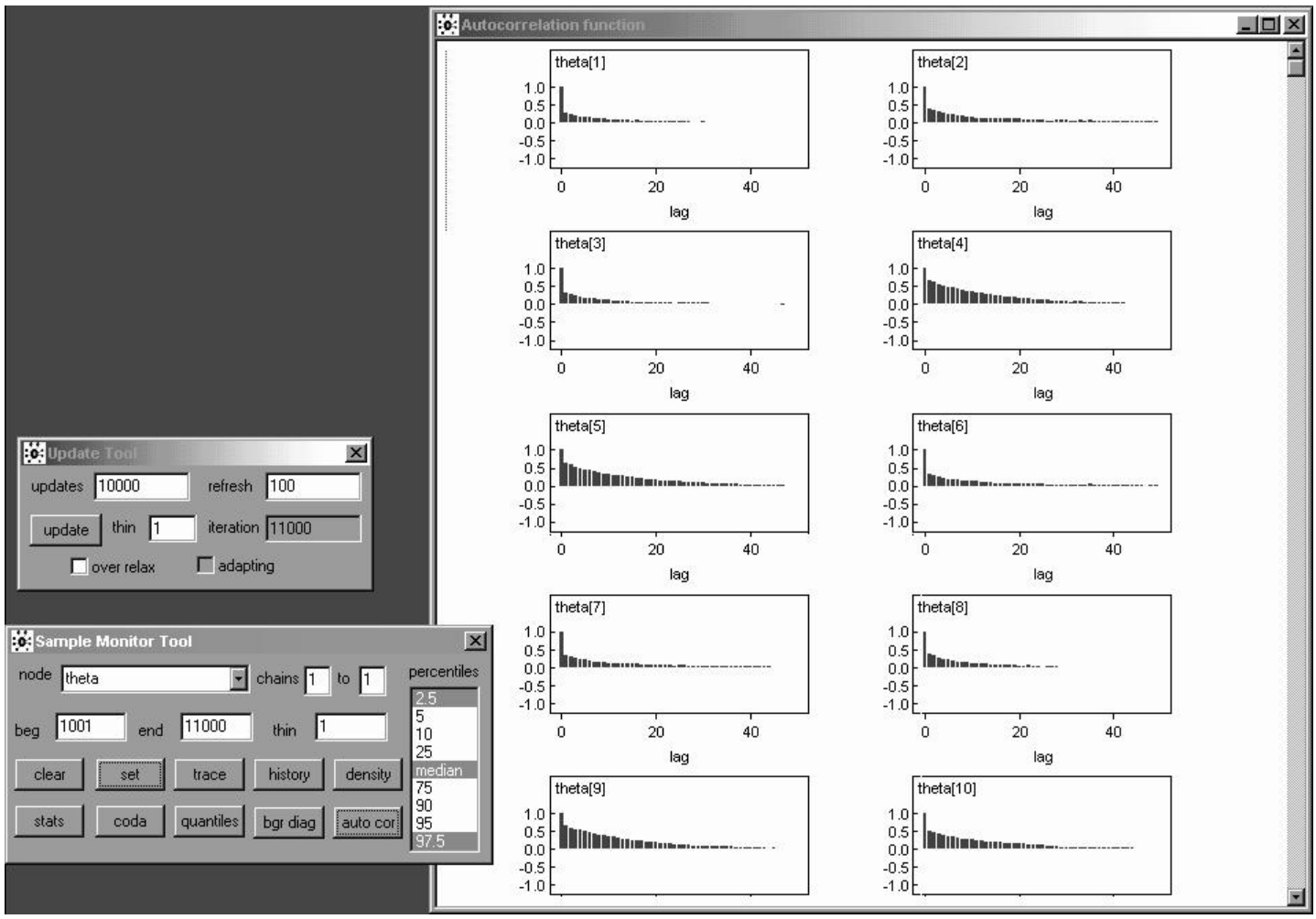


Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
theta[1]	0.07914	0.0667	0.001621	-0.06158	0.08178	0.2083	1001	10000
theta[2]	-0.03745	0.07771	0.002503	-0.1787	-0.04117	0.131	1001	10000
theta[3]	-0.07873	0.0771	0.002139	-0.2409	-0.07674	0.07068	1001	10000
theta[4]	0.4412	0.1206	0.005198	0.2206	0.4379	0.6875	1001	10000
theta[5]	0.4088	0.1159	0.004757	0.1802	0.4117	0.6283	1001	10000
theta[6]	-0.06684	0.06417	0.001823	-0.1915	-0.06695	0.06059	1001	10000
theta[7]	-0.05633	0.06543	0.001951	-0.1834	-0.05685	0.07543	1001	10000
theta[8]	-0.09112	0.08347	0.002333	-0.2772	-0.08547	0.06136	1001	10000
theta[9]	0.3942	0.1029	0.00442	0.1846	0.398	0.5867	1001	10000
theta[10]	0.2915	0.09705	0.003804	0.1266	0.2851	0.5119	1001	10000
theta[11]	0.42	0.1139	0.004813	0.199	0.4217	0.6415	1001	10000
theta[12]	0.3939	0.1095	0.004472	0.1722	0.3971	0.5978	1001	10000
theta[13]	0.2387	0.09166	0.003086	0.05124	0.2429	0.4105	1001	10000
theta[14]	0.09989	0.08011	0.002044	-0.05295	0.09718	0.2684	1001	10000
theta[15]	-0.08512	0.07569	0.002023	-0.2462	-0.08189	0.05945	1001	10000
theta[16]	-0.06751	0.07592	0.002041	-0.2161	-0.06628	0.08384	1001	10000
theta[17]	0.2609	0.07768	0.002718	0.1114	0.261	0.4162	1001	10000
theta[18]	0.08658	0.05889	0.001467	-0.03271	0.08696	0.2029	1001	10000
theta[19]	-0.0695	0.07743	0.002036	-0.2254	-0.06907	0.08493	1001	10000

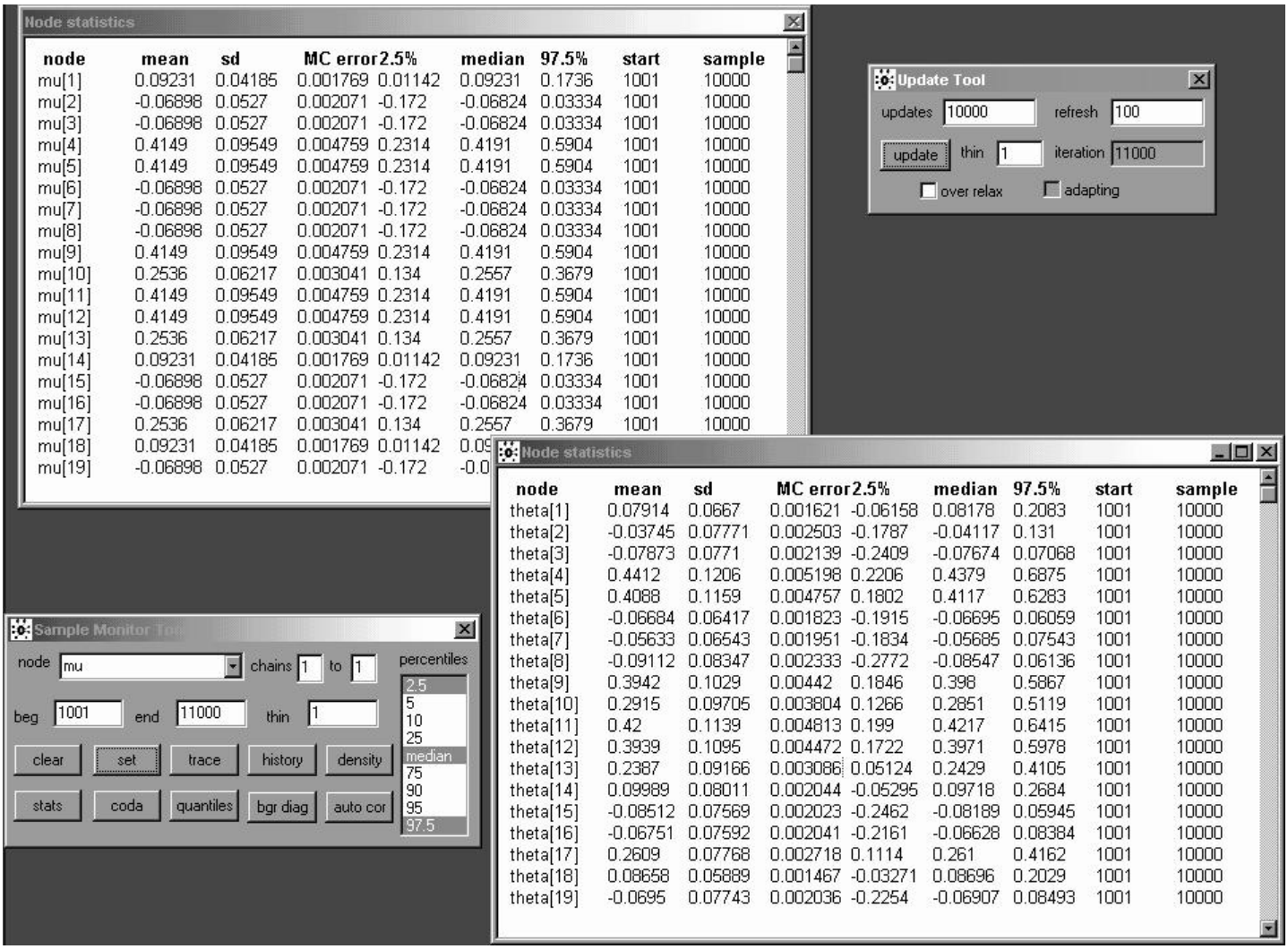
Many of the θ_i have **posterior probability concentrated near 0**, but not all; $\theta_4, \theta_5, \theta_9, \theta_{11}$, and θ_{12} are particularly large (looking back on page 12, what's **special** about the corresponding studies?).

WinBUGS Implementation



Some of the θ_i are **not far from white noise**; others are **mixing quite slowly**.

WinBUGS Implementation



Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
mu[1]	0.09231	0.04185	0.001769	0.01142	0.09231	0.1736	1001	10000
mu[2]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[3]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[4]	0.4149	0.09549	0.004759	0.2314	0.4191	0.5904	1001	10000
mu[5]	0.4149	0.09549	0.004759	0.2314	0.4191	0.5904	1001	10000
mu[6]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[7]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[8]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[9]	0.4149	0.09549	0.004759	0.2314	0.4191	0.5904	1001	10000
mu[10]	0.2536	0.06217	0.003041	0.134	0.2557	0.3679	1001	10000
mu[11]	0.4149	0.09549	0.004759	0.2314	0.4191	0.5904	1001	10000
mu[12]	0.4149	0.09549	0.004759	0.2314	0.4191	0.5904	1001	10000
mu[13]	0.2536	0.06217	0.003041	0.134	0.2557	0.3679	1001	10000
mu[14]	0.09231	0.04185	0.001769	0.01142	0.09231	0.1736	1001	10000
mu[15]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[16]	-0.06898	0.0527	0.002071	-0.172	-0.06824	0.03334	1001	10000
mu[17]	0.2536	0.06217	0.003041	0.134	0.2557	0.3679	1001	10000
mu[18]	0.09231	0.04185	0.001769	0.01142	0.09			
mu[19]	-0.06898	0.0527	0.002071	-0.172	-0.0			

Update Tool

updates 10000 refresh 100

update thin 1 iteration 11000

over relax adapting

Sample Monitor Tool

node mu chains 1 to 1 percentiles

beg 1001 end 11000 thin 1

clear set trace history density

stats coda quantiles bgr diag auto cor

2.5 5 10 25 median 75 90 95 97.5

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
theta[1]	0.07914	0.0667	0.001621	-0.06158	0.08178	0.2083	1001	10000
theta[2]	-0.03745	0.07771	0.002503	-0.1787	-0.04117	0.131	1001	10000
theta[3]	-0.07873	0.0771	0.002139	-0.2409	-0.07674	0.07068	1001	10000
theta[4]	0.4412	0.1206	0.005198	0.2206	0.4379	0.6875	1001	10000
theta[5]	0.4088	0.1159	0.004757	0.1802	0.4117	0.6283	1001	10000
theta[6]	-0.06684	0.06417	0.001823	-0.1915	-0.06695	0.06059	1001	10000
theta[7]	-0.05633	0.06543	0.001951	-0.1834	-0.05685	0.07543	1001	10000
theta[8]	-0.09112	0.08347	0.002333	-0.2772	-0.08547	0.06136	1001	10000
theta[9]	0.3942	0.1029	0.00442	0.1846	0.398	0.5867	1001	10000
theta[10]	0.2915	0.09705	0.003804	0.1266	0.2851	0.5119	1001	10000
theta[11]	0.42	0.1139	0.004813	0.199	0.4217	0.6415	1001	10000
theta[12]	0.3939	0.1095	0.004472	0.1722	0.3971	0.5978	1001	10000
theta[13]	0.2387	0.09166	0.003086	0.05124	0.2429	0.4105	1001	10000
theta[14]	0.09989	0.08011	0.002044	-0.05295	0.09718	0.2684	1001	10000
theta[15]	-0.08512	0.07569	0.002023	-0.2462	-0.08189	0.05945	1001	10000
theta[16]	-0.06751	0.07592	0.002041	-0.2161	-0.06628	0.08384	1001	10000
theta[17]	0.2609	0.07768	0.002718	0.1114	0.261	0.4162	1001	10000
theta[18]	0.08658	0.05889	0.001467	-0.03271	0.08696	0.2029	1001	10000
theta[19]	-0.0695	0.07743	0.002036	-0.2254	-0.06907	0.08493	1001	10000

It's also useful to monitor the $\mu_i = \alpha + \beta(x_i - \bar{x})$, because they represent an **important part of the shrinkage story** with model (10).

Shrinkage Estimation

In a manner parallel to the situation with the simpler model (1), the posterior means of the **underlying study effects** θ_i should be at least approximately related to the **raw effect sizes** y_i and the μ_i via the **shrinkage equation**

$$E(\theta_i|y) \doteq \left(1 - \widehat{B}_i\right) y_i + \widehat{B}_i E(\mu_i|y)\,; \qquad (34)$$

here $\widehat{B}_i = \frac{V_i}{V_i + \widehat{\sigma}^2_\theta}$ and $\widehat{\sigma}^2_\theta$ is the posterior mean of σ^2_θ.

This is **easy to check** in R:

```
> mu <- c( 0.09231, -0.06898, -0.06898, 0.4149, 0.4149, -0.06898, -0.06898,
    -0.06898, 0.4149, 0.2536, 0.4149, 0.4149, 0.2536, 0.09231, -0.06898,
    -0.06898, 0.2536, 0.09231, -0.06898 )

> y <- c( 0.03, 0.12, -0.14, 1.18, 0.26, -0.06, -0.02, -0.32, 0.27, 0.80,
    0.54, 0.18, -0.02, 0.23, -0.18, -0.06, 0.30, 0.07, -0.07 )

> theta <- c( 0.08144, -0.03455, -0.07456, 0.4377, 0.4076, -0.0628,
    -0.05262, -0.08468, 0.3934, 0.289, 0.4196, 0.3938, 0.2393, 0.1014,
    -0.08049, -0.06335, 0.2608, 0.08756, -0.06477 )

> V <- 1 / tau

> B.hat <- V / ( V + 0.064^2 )

> theta.approx <- ( 1 - B.hat ) * y + B.hat * mu
```

The Shrinkage Story (continued)

```
> cbind( y, theta, mu, sigma.2, B.hat, theta.approx )

          y    theta       mu        V     B.hat theta.approx
 [1,]  0.03  0.08144  0.09231 0.015625 0.7923026   0.07936838
 [2,]  0.12 -0.03455 -0.06898 0.021609 0.8406536  -0.03886671
 [3,] -0.14 -0.07456 -0.06898 0.027889 0.8719400  -0.07807482
 [4,]  1.18  0.43770  0.41490 0.139129 0.9714016   0.43678060
 [5,]  0.26  0.40760  0.41490 0.136161 0.9707965   0.41037637
 [6,] -0.06 -0.06280 -0.06898 0.010609 0.7214553  -0.06647867
 [7,] -0.02 -0.05262 -0.06898 0.010609 0.7214553  -0.05533688
 [8,] -0.32 -0.08468 -0.06898 0.048400 0.9219750  -0.08856583
 [9,]  0.27  0.39340  0.41490 0.026896 0.8678369   0.39574956
[10,]  0.80  0.28900  0.25360 0.063001 0.9389541   0.28695551
[11,]  0.54  0.41960  0.41490 0.091204 0.9570199   0.42027681
[12,]  0.18  0.39380  0.41490 0.049729 0.9239015   0.39702447
[13,] -0.02  0.23930  0.25360 0.083521 0.9532511   0.24080950
[14,]  0.23  0.10140  0.09231 0.084100 0.9535580   0.09870460
[15,] -0.18 -0.08049 -0.06898 0.025281 0.8605712  -0.08445939
[16,] -0.06 -0.06335 -0.06898 0.027889 0.8719400  -0.06783002
[17,]  0.30  0.26080  0.25360 0.019321 0.8250843   0.26171609
[18,]  0.07  0.08756  0.09231 0.008836 0.6832663   0.08524367
[19,] -0.07 -0.06477 -0.06898 0.030276 0.8808332  -0.06910155
```

You can see that equation (11) is indeed a **good approximation** to what's going on: the posterior means of the θ_i (column 3 of this table, counting the leftmost column of study indices) all fall between the y_i (column 2) and the posterior means of the μ_i (column 4), with the closeness to y_i or $E(\mu_i|y)$ expressed through the **shrinkage factor** $\widehat{B}_i$.

Since $\widehat{\sigma}^2_\theta$ is small (i.e., most—but not quite all—of the between-study variation has been explained by the covariate x), the raw y_i values are shrunken **almost all of the way toward the regression line** $\alpha + \beta(x_i - \overline{x})$.

Hierarchical Model Selection: A Case Study

Case Study: *In-home geriatric assessment (IHGA)*. In an experiment conducted in the 1980s (Hendriksen et al. 1984), 572 elderly people living in a number of villages in Denmark were randomized, 287 to a **control** (C) group (who received standard care) and 285 to an **experimental** (E) group (who received standard care plus IHGA: a kind of **preventive medicine** in which each person's medical and social needs were assessed and acted upon individually).

One important outcome was the number of **hospitalizations** during the two-year life of the study (Table 4.1).

Table 4.1. Distribution of number of hospitalizations in the IHGA study over a two-year period.

	Number of Hospitalizations										
Group	0	1	2	3	4	5	6	7	n	Mean	SD
Control	138	77	46	12	8	4	0	2	287	0.944	1.24
Experimental	147	83	37	13	3	1	1	0	285	0.768	1.01

Evidently IHGA lowered the mean hospitalization rate (for these elderly Danish people, at least) by $(0.944 - 0.768) = \mathbf{0.176}$, which is about a $100\left(\frac{0.768-0.944}{0.944}\right) = \mathbf{19\%}$ reduction from the control level, a difference that's **large in clinical terms**.

Modeling the IHGA Data

An **off-the-shelf** analysis of this experiment might pretend (**Model 0**) that the data are Gaussian,

$$\begin{aligned} \left(C_i|\mu_C,\sigma_C^2\right) &\overset{\text{IID}}{\sim} N\left(\mu_C,\sigma_C^2\right), i = 1,\ldots,n_C, \\ \left(E_j|\mu_E,\sigma_E^2\right) &\overset{\text{IID}}{\sim} N\left(\mu_E,\sigma_E^2\right), j = 1,\ldots,n_E, \end{aligned} \quad (35)$$

and use the ordinary frequentist **two-independent-samples "z-machinery"**:

```
rosalind 15> R

R : Copyright 2001, The R Development Core Team
Version 1.2.1  (2001-01-15)

> C <- c( rep( 0, 138 ), rep( 1, 77 ), rep( 2, 46 ),
    rep( 3, 12 ), rep( 4, 8 ), rep( 5, 4 ), rep( 7, 2 ) )

> print( n.C <- length( C ) )

[1] 287                 # sample size in the control group

> mean( C )

[1] 0.9442509           # control group mean

> sqrt( var( C ) )

[1] 1.239089            # control group
                        # standard deviation (SD)
> table( C )

   0  1  2  3 4 5 7     # control group
 138 77 46 12 8 4 2     # frequency distribution
```

Analysis of Model 0

```
> E <- c( rep( 0, 147 ), rep( 1, 83 ), rep( 2, 37 ),
    rep( 3, 13 ),rep( 4, 3 ), rep( 5, 1 ), rep( 6, 1 ) )

> print( n.E <- length( E ) )

[1] 285                 # sample size in the
                        # experimental group
> mean( E )

[1] 0.7684211           # experimental group mean

> sqrt( var( E ) )

[1] 1.008268            # experimental group SD

> table( E )

   0  1  2  3 4 5 6     # experimental group
 147 83 37 13 3 1 1     # frequency distribution

> print( effect <- mean( E ) - mean( C ) )

[1] -0.1758298          # mean difference ( E - C )

> effect / mean( C )

[1] -0.1862109          # relative difference ( E - C ) / C

> SE.effect <- sqrt( var( C ) / n.C + var( E ) / n.E )

[1] 0.09442807          # standard error of the difference

> print( CI <- c( effect - 1.96 * SE.effect,
    effect + 1.96 * SE.effect ) )

[1] -0.3609 0.009249   # the 95% confidence interval from
                       # model 0 runs from -.36 to +.01
```

Deficiencies of Model 0

The frequentist analysis of Model 0 is equivalent to a Bayesian analysis of the same model with **diffuse priors** on the control and experimental group means and SDs $(\mu_C, \sigma_C, \mu_E, \sigma_E)$, and is summarized in Table 4.2.

Table 4.2. Summary of analysis of Model 0.

	Posterior		
	Mean	SD	95% Interval
Treatment effect $(\mu_E - \mu_C)$	-0.176	0.0944	$(-0.361, 0.009)$

However, both distributions have long right-hand tails; in fact they look rather **Poisson**.

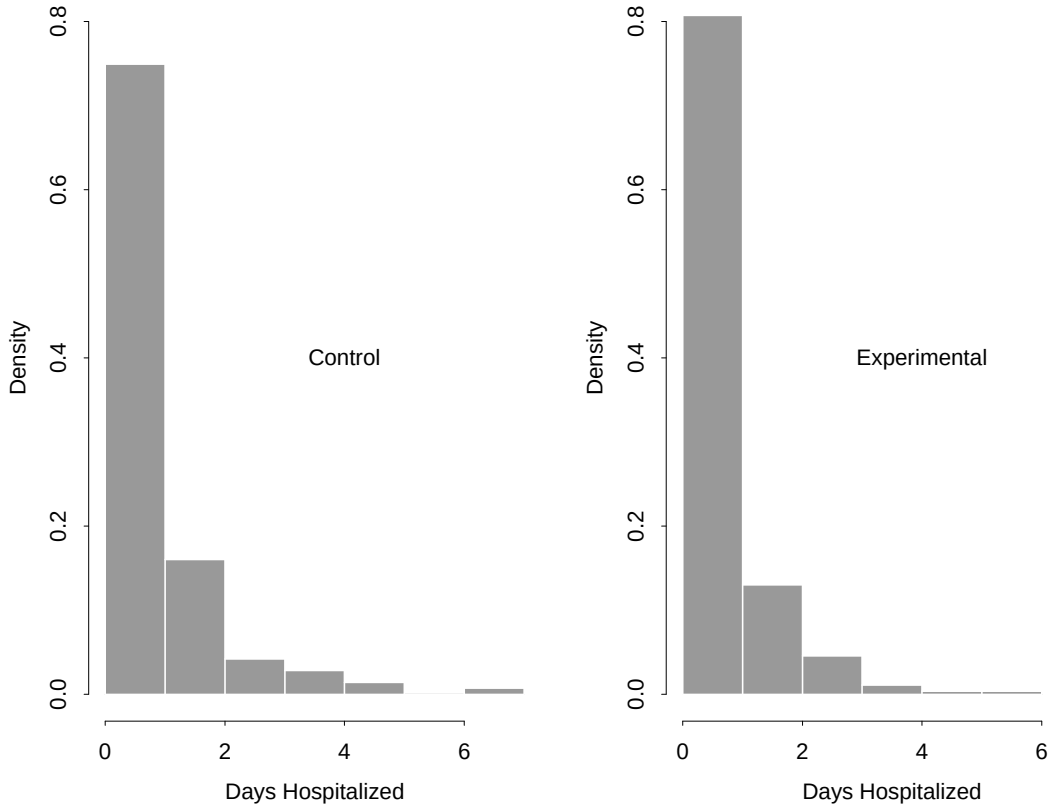


Figure 4.1. Histograms of control and experimental numbers of hospitalizations.

4.1.1 Poisson Fixed-Effects Modeling

R code to make the histograms:

```
> x11( )                              # to open a
                                      #   graphics window
> par( mfrow = c( 1, 2 ) )            # to plot two histograms

> hist( C, nclass = 8, probability = T,
    xlab = 'Days Hospitalized', ylab = 'Density',
    xlim = c( 0, 7 ), ylim = c( 0, 0.8 ) )

> text( 4, 0.4, 'Control' )

> hist( E, nclass = 8, probability = T,
    xlab = 'Days Hospitalized', ylab = 'Density',
    xlim = c( 0, 7 ), ylim = c( 0, 0.8 ) )

> text( 4, 0.4, 'Experimental' )
```

So I **created** a classicBUGS file called poisson1.bug that looked like this:

```
model poisson1;

const

  n.C = 287, n.E = 285;

var

  lambda.C, lambda.E, C[ n.C ], E[ n.E ], effect;

data C in "poisson-C.dat", E in "poisson-E.dat";

inits in "poisson1.in";
```

Initial Poisson Modeling (continued)

```
{

  lambda.C ~ dgamma( 0.001, 0.001 );
  lambda.E ~ dgamma( 0.001, 0.001 );

  for ( i in 1:n.C ) {

    C[ i ] ~ dpois( lambda.C );

  }

  for ( j in 1:n.E ) {

    E[ j ] ~ dpois( lambda.E );

  }

  effect <- lambda.E - lambda.C;

}
```

`poisson1.in` initializes both λ_C and λ_E to 1.0; the $\Gamma(0.001, 0.001)$ priors for λ_C and λ_E are chosen (as usual to create diffuseness) to be **flat** in the region in which the **likelihood** is **appreciable**:

```
> sqrt( var( C ) / n.C )

[1] 0.07314114

> sqrt( var( E ) / n.E )

[1] 0.05972466

> c( mean( C ) - 3.0 * sqrt( var( C ) / n.C ),
    mean( C ) + 3.0 * sqrt( var( C ) / n.C ) )
```

Initial Poisson Modeling (continued)

```
[1] 0.7248275 1.1636743

> c( mean( E ) - 3.0 * sqrt( var( E ) / n.E ),
    mean( E ) + 3.0 * sqrt( var( E ) / n.E ) )

[1] 0.5892471 0.9475950

> lambda.grid <- seq( 0.01, 2.0, 0.01 )

> plot( lambda.grid, 0.001 * dgamma( lambda.grid, 0.001 ),
    type = 'l', xlab = 'Lambda', ylab = 'Density' )
```

The likelihood under the Gaussian model is **concentrated** for λ_C from about 0.7 to 1.2, and that for λ_E from about 0.6 to 1; you can see from the plot that across those ranges the $\Gamma(0.001, 0.001)$ prior is **essentially constant**.

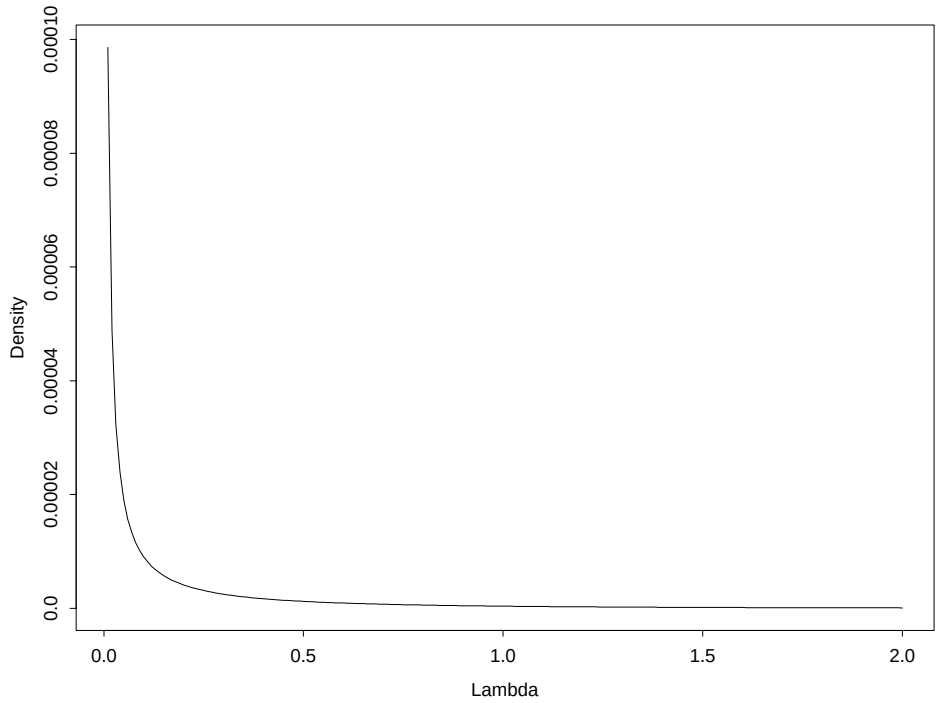
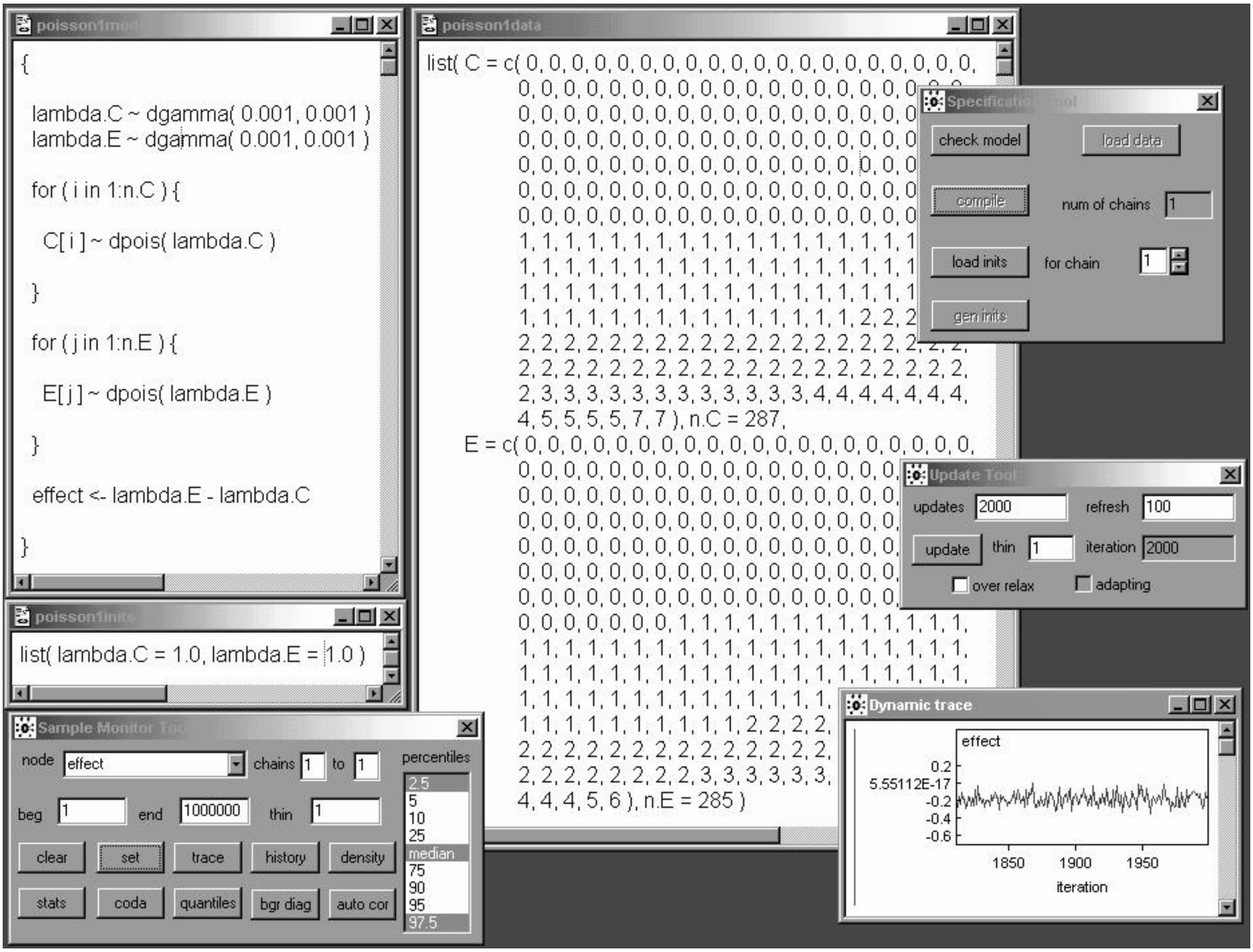


Figure 4.2. The $\Gamma(0.001, 0.001)$ distribution in the region in which the likelihoods for λ_C and λ_E are appreciable.

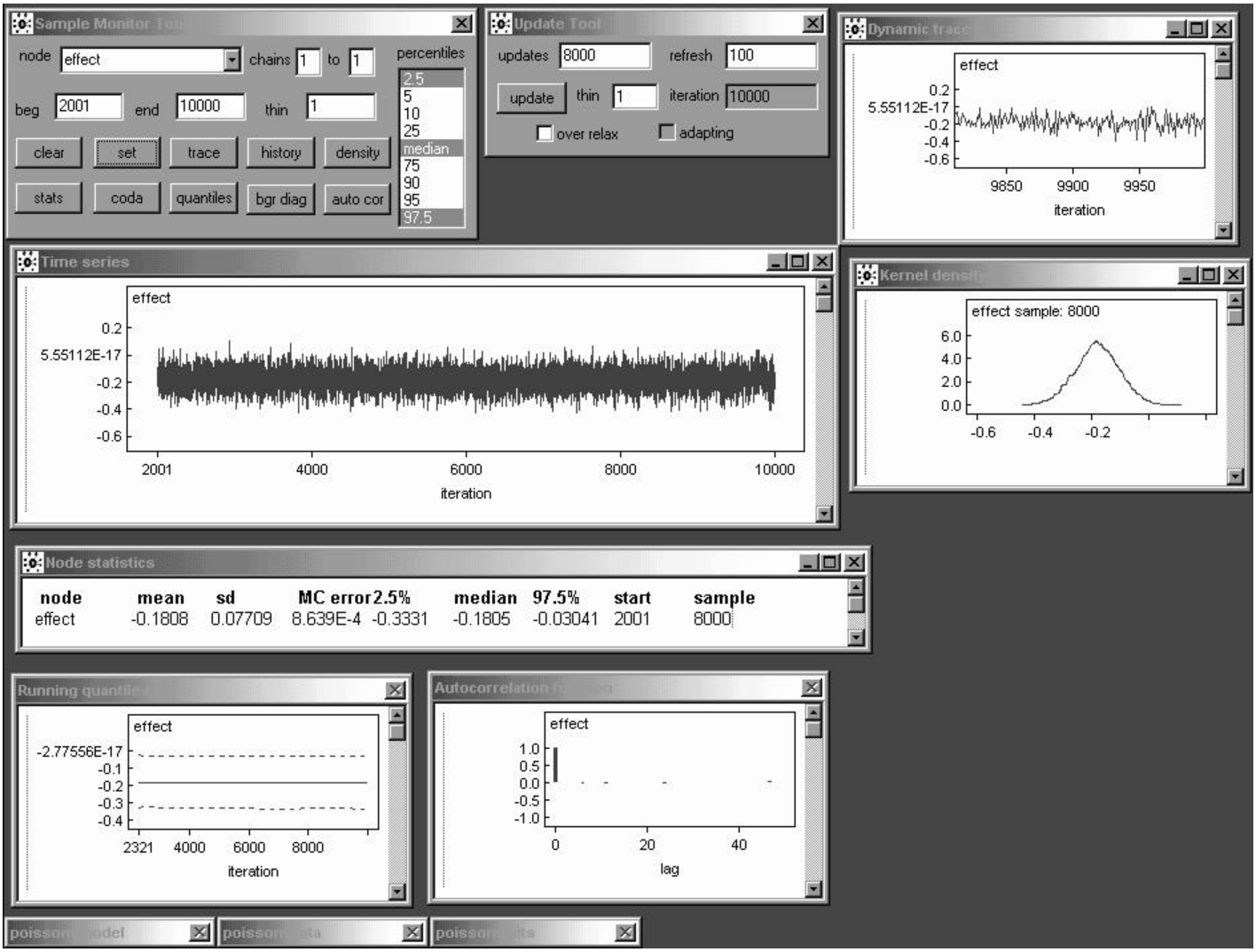
WinBUGS Implementation



The **screendump** above presents part of the results of fitting the **2-independent-samples additive Poisson model** at the top of page 8 in WinBUGS.

A **burn-in of 2,000** was almost **instantaneous at 2.0 PC GHz** and revealed **good mixing** for the three main quantities of interest.

WinBUGS Implementation (continued)



A monitoring run of 8,000 reveals that the effect parameter in the **2-independent-samples Poisson model** is behaving like **white noise**, so that already with only 8,000 iterations the posterior mean has a Monte Carlo standard error of **less than 0.001**.

Initial Poisson Modeling (continued)

Thus a burn-in of 2,000 and a monitoring run of 8,000 yields **good MCMC diagnostics** and permits a comparison between model 0 (Gaussian) and model 1 (Poisson), as in Table 4.3.

Table 4.3. Comparison of inferential conclusions from models 0 and 1.

$\boxed{\lambda_C}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
Gaussian	0.944	0.0731	$(0.801, 1.09)$
Poisson	0.943	0.0577	$(0.832, 1.06)$

$\boxed{\lambda_E}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
Gaussian	0.768	0.0597	$(0.651, 0.885)$
Poisson	0.769	0.0521	$(0.671, 0.875)$

$\boxed{\Delta = \lambda_E - \lambda_C}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
Gaussian	-0.176	0.0944	$(-0.361, 0.009)$
Poisson	-0.174	0.0774	$(-0.325, -0.024)$

The two models produce **almost identical point estimates**, but the Poisson model leads to **sharper inferences** (e.g., the posterior SD for the treatment effect $\Delta = \lambda_E - \lambda_C$ is **22%** larger in model 0 than in model 1).

Additive and Multiplicative Treatment Effects

This is the same point we noticed with the NB10 data—when a location parameter is the only thing at issue, the Gaussian is a **conservative** modeling choice (intuitively, the Poisson gains its "extra accuracy" from the variance and the mean being equal, which permits **second-moment** information to help in estimating the λ values along with the usual first-moment information).

Both the Gaussian and Poisson models so far implicitly assume that the treatment effect is **additive**:

$$E \stackrel{\text{st}}{=} C + \text{effect}, \tag{36}$$

where $\stackrel{\text{st}}{=}$ means *is stochastically equal to*; in other words, apart from random variation the effect of the IHGA is to **add or subtract a constant** to or from each person's underlying rate of hospitalization.

However, since the outcome variable is non-negative, it is plausible that a **better model** for the data is

$$E \stackrel{\text{st}}{=} (1 + \text{effect})\, C. \tag{37}$$

Additive vs. Multiplicative Effect

Here the treatment effect is **multiplicative**—in other words, apart from random variation the effect of the IHGA is to **multiply** each person's underlying rate of hospitalization by a constant above or below 1.

A **qqplot** of the control and experimental outcome values can in some cases be helpful in choosing between additive and multiplicative models:

```
> CEqq <- qqplot( C, E, plot = F )

> table( CEqq$y, CEqq$x )

                Interpolated C values
     0 0.965  1 1.5  2 2.82 3 3.91 4 4.96 5 6.99 7
  0 137     1  9   0  0    0 0    0 0    0 0    0 0
  1   0     0 66   1 16    0 0    0 0    0 0    0 0
  2   0     0  0   0 29    1 7    0 0    0 0    0 0
E 3   0     0  0   0  0    0 4    1 7    1 0    0 0
  4   0     0  0   0  0    0 0    0 0    0 3    0 0
  5   0     0  0   0  0    0 0    0 0    0 0    1 0
  6   0     0  0   0  0    0 0    0 0    0 0    0 1

> symbols( c( 0, 0.964798, 1, 1, 1.5, 2, 2, 2.823944, 3, 3,
    3.908447, 4, 4.964813, 5, 6.985962, 7 ), c( rep( 0, 3 ),
    rep( 1, 3 ), rep( 2, 3 ), rep( 3, 4 ), 4, 5, 6 ),
    circles = c( 137, 1, 9, 66, 1, 16, 29, 1, 7, 4, 1, 7, 1,
    3, 1, 1 ), xlab = 'C', ylab = 'E' )
```

Additive vs. Multiplicative Effect

```
> abline( 0, 1 )                    # E = C (no effect)

> abline( 0, 0.793, lty = 2 )       # E = 0.816 C
                                    #   (multiplicative)

> abline( -0.174, 1, lty = 3 )      # E = C - 0.174 (additive)
```

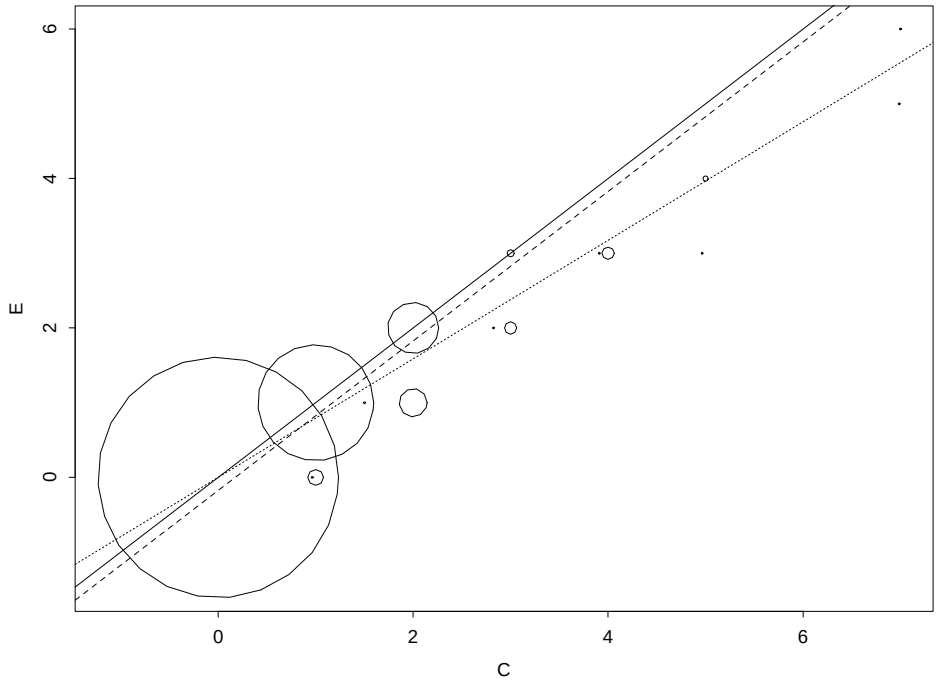


Figure 4.3. QQplot of E versus C values, with the radii of the plotted circles proportional to the number of observations at the indicated point. The solid line corresponds to no treatment effect, the small dotted line to the best-fitting multiplicative model ($E \stackrel{\text{st}}{=} 0.816\,C$), and the large dotted line to the best-fitting additive model ($E \stackrel{\text{st}}{=} C - 0.174$).

Here, because the Poisson model has only **one parameter** for both location and scale, the multiplicative and additive formulations **fit equally well**, but the multiplicative model **generalizes** more readily (see below).

A Multiplicative Poisson Model

A simple way to write the multiplicative model is to re-express the data in the form of a **regression** of the outcome y on a **dummy variable** x which is 1 if the person was in the experimental group and 0 if he/she was in the control group:

i	1	2	$\cdots$	287	288	289	$\cdots$	572
x_i	0	0	$\cdots$	0	1	1	$\cdots$	1
y_i	1	0	$\cdots$	2	0	3	$\cdots$	1

Then for $i = 1, \ldots, n = 572$ the **multiplicative** model can be written

$$\begin{array}{rcl} (y_i \,|\lambda_i) & \overset{\text{indep}}{\sim} & \text{Poisson}(\lambda_i) \\ \log(\lambda_i) & = & \gamma_0 + \gamma_1 x_i \\ (\gamma_0, \gamma_1) & \sim & \text{diffuse} \end{array} \tag{38}$$

In this model the **control** people have

$$\log(\lambda_i) = \gamma_0 + \gamma_1(0) = \gamma_0, \quad \text{i.e.,} \quad \lambda_C = e^{\gamma_0}, \tag{39}$$

and the **experimental** people have

$$\begin{array}{rcl} \log(\lambda_i) & = & \gamma_0 + \gamma_1(1) = \gamma_0 + \gamma_1, \quad \text{i.e.,} \\ \lambda_E & = & e^{\gamma_0+\gamma_1} = e^{\gamma_0} e^{\gamma_1} = \lambda_C e^{\gamma_1}. \end{array} \tag{40}$$

Now you may remember from basic **Taylor series** that for γ_1 not too far from 0

$$e^{\gamma_1} \doteq 1 + \gamma_1, \tag{41}$$

A Multiplicative Poisson Model

so that finally (for γ_1 **fairly near 0**)

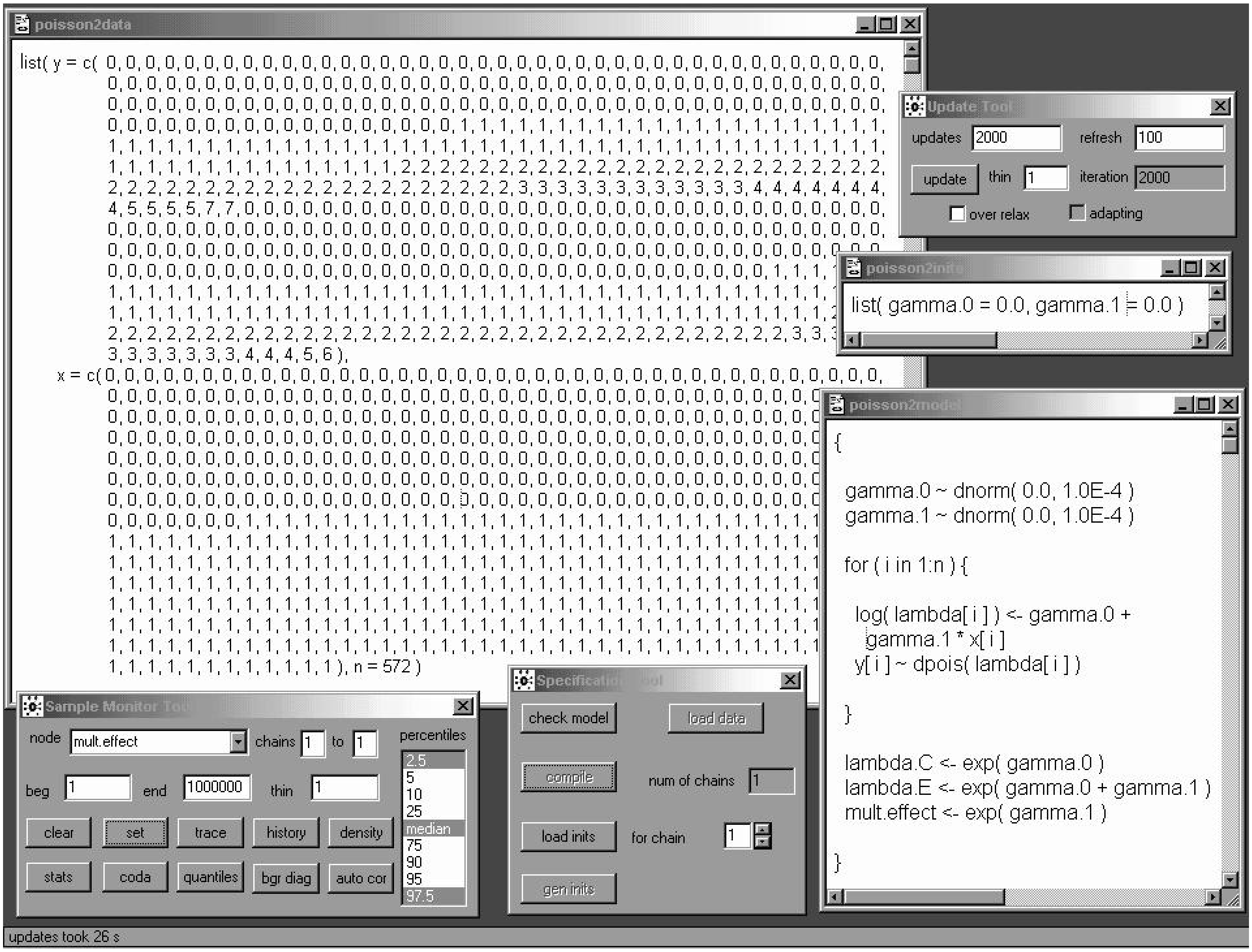
$$\lambda_E \doteq (1 + \gamma_1)\,\lambda_C, \tag{42}$$

which is a way of expressing equation (3) in **Poisson language**.

Fitting this model in BUGS is easy:

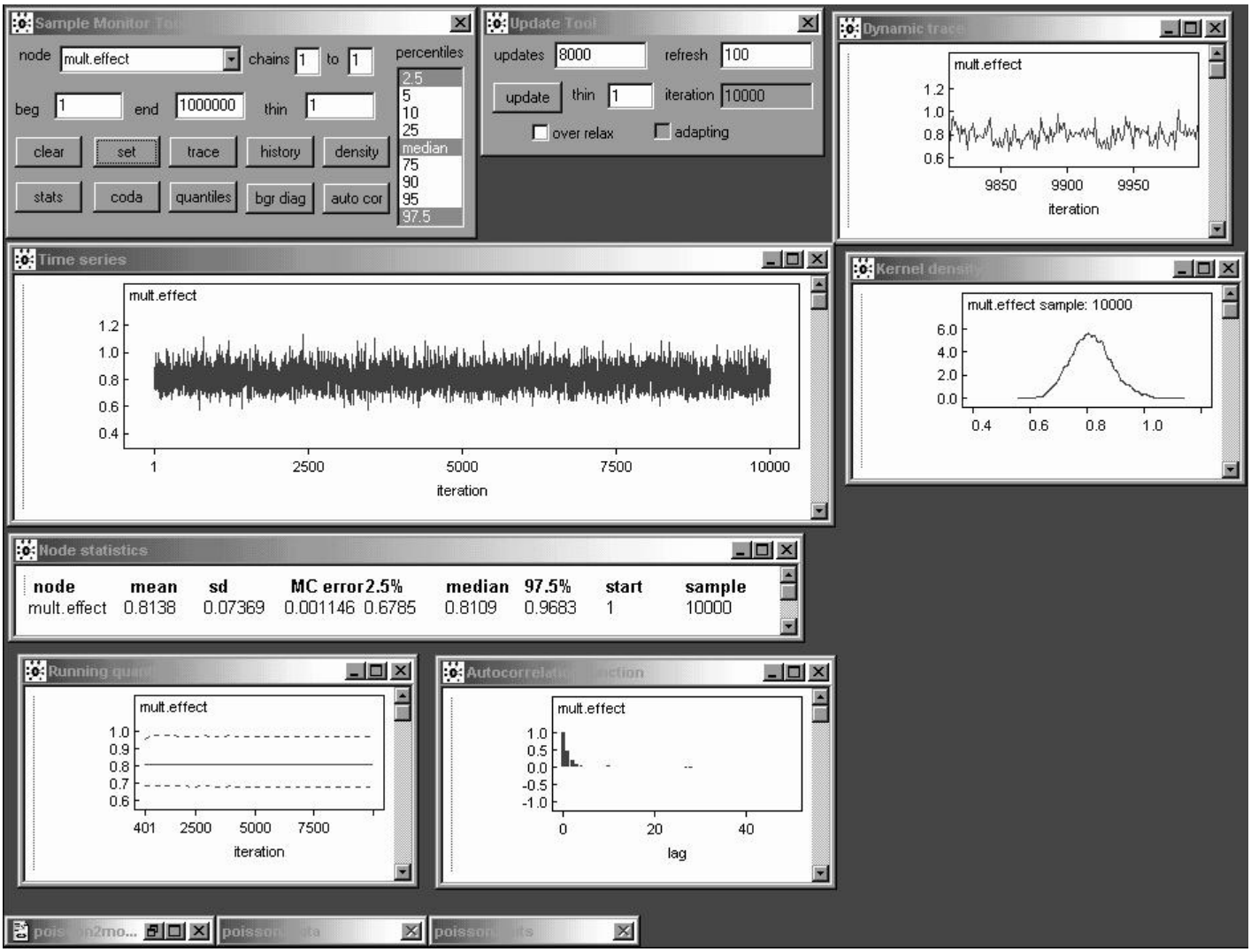
```
model poisson2;

const

  n = 572;

var

  gamma.0, gamma.1, lambda[ n ], x[ n ], y[ n ], lambda.C,
  lambda.E, mult.effect;

data x in "poisson-x.dat", y in "poisson-y.dat";
inits in "poisson2.in";

{
  gamma.0 ~ dnorm( 0.0, 1.0E-4 );    # flat priors for
  gamma.1 ~ dnorm( 0.0, 1.0E-4 );    # gamma.0 and gamma.1

  for ( i in 1:n ) {

    log( lambda[ i ] ) <- gamma.0 + gamma.1 * x[ i ];
    y[ i ] ~ dpois( lambda[ i ] );

  }

  lambda.C <- exp( gamma.0 );
  lambda.E <- exp( gamma.0 + gamma.1 );
  mult.effect <- exp( gamma.1 );
}
```

WinBUGS Implementation (continued)



The **multiplicative Poisson model** (11) takes longer to run—2,000 burn-in iterations now take about **4 seconds at 2.0 PC GHz**—but still exhibits **fairly good mixing**, as we'll see below.

WinBUGS Implementation (continued)



A total of **10,000 iterations** (the chain started essentially in equilibrium, so the burn-in can be absorbed into the monitoring run) reveals that the **multiplicative effect parameter** e^{γ_1} in model (11) behaves like an AR_1 series with $\widehat{\rho}_1 \doteq 0.5$, but the Monte Carlo standard error for the posterior mean is still only about **0.001** with a run of this length.

Additive versus Multiplicative Fit

A burn-in of 2,000 and a monitoring run of 8,000 again yields **good MCMC diagnostics** and permits a comparison between the additive and multiplicative Poisson models, as in Table 4.4.

Table 4.4. Comparison of inferential conclusions from the additive and multiplicative Poisson models.

$\boxed{\lambda_C}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
additive	0.943	0.0577	$(0.832, 1.06)$
multiplicative	0.945	0.0574	$(0.837, 1.06)$

$\boxed{\lambda_E}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
additive	0.769	0.0521	$(0.671, 0.875)$
multiplicative	0.768	0.0518	$(0.671, 0.872)$

$\boxed{\text{effect}}$ Model	Posterior Mean	Posterior SD	Central 95% Interval
additive	-0.174	0.0774	$(-0.325, -0.024)$
multiplicative	-0.184	0.0743	$(-0.324, -0.033)$

With this model it is as if the experimental people's average underlying rates of hospitalization have been **multiplied by 0.82**, give or take about 0.07.

The additive and multiplicative effects are **similar** here, because both are not too far from zero.

Extra-Poisson Variability

However, none of this has verified that the **Poisson model is reasonable** for these data—the histograms show that the Gaussian model is clearly unreasonable, but the diagnostic plots in WinBUGS and CODA only check on the adequacy of the **MCMC** sampling, not the model.

In fact we had a good clue that the data are **not** Poisson back on page 2: as noted in part 2, the Poisson(λ) distribution has mean λ and also variance λ—in other words, the **variance-to-mean-ratio** (VTMR) for the Poisson is 1. But

```
> var( C ) / mean( C )
[1] 1.62599
> var( E ) / mean( E )
[1] 1.322979
```

i.e., the data exhibit **extra-Poisson variability** (VTMR > 1).

This actually **makes good sense** if you think about it, as follows.

The Poisson model assumes that everybody in the control group has the **same underlying rate** λ_C of hospitalization, and similarly everybody in the experimental group has the **same rate** λ_E.

Unobserved Predictor Variables

In reality it's far more reasonable to think that each person has his/her **own** underlying rate of hospitalization that depends on **baseline health status**, **age**, and various other things.

Now Hendriksen **forgot to measure** (or at least to report on) these other variables (he may have hoped that the randomization would balance them between C and E)—the only predictor we have is x, the **experimental status dummy variable**—so the best we can do is to lump all of these other **unobserved** predictor variables together into a kind of **"error" term** e.

This amounts to **expanding** the second Poisson model (11) above: for $i = 1, \ldots, n = 572$ the new model is

$$\begin{array}{rcll} (y_i \,|\lambda_i) & \overset{\text{indep}}{\sim} & \text{Poisson}(\lambda_i) & \\ \log(\lambda_i) & = & \gamma_0 + \gamma_1 x_i + e_i & (43) \\ e_i & \overset{\text{IID}}{\sim} & N\left(0, \sigma_e^2\right) & \\ \left(\gamma_0, \gamma_1, \sigma_e^2\right) & \sim & \text{diffuse.} & \end{array}$$

Random-
Effects Poisson Regression

The **Gaussian** choice for the error distribution is **conventional**, not dictated by the science of the problem (although if there were a lot of unobserved predictors hidden inside the e_i their weighted sum would be close to normal by the **Central Limit Theorem**).

Model (16) is an **expansion** of the earlier model (11) because you can obtain model (11) from (16) by setting $\sigma_e^2 = 0$, whereas with (16) we're letting σ_e^2 vary and **learning about it from the data**.

The addition of the **random effects** e_i to the model is one way to address the extra-Poisson variability: this model would be called a **lognormal mixture of Poisson distributions** (or a **random effects Poisson regression** (REPR) model) because it's as if each person's λ is drawn from a lognormal distribution and then his/her number of hospitalizations y is drawn from a Poisson distribution with his/her λ, and this mixing process will make the variance of y **bigger than its mean**.

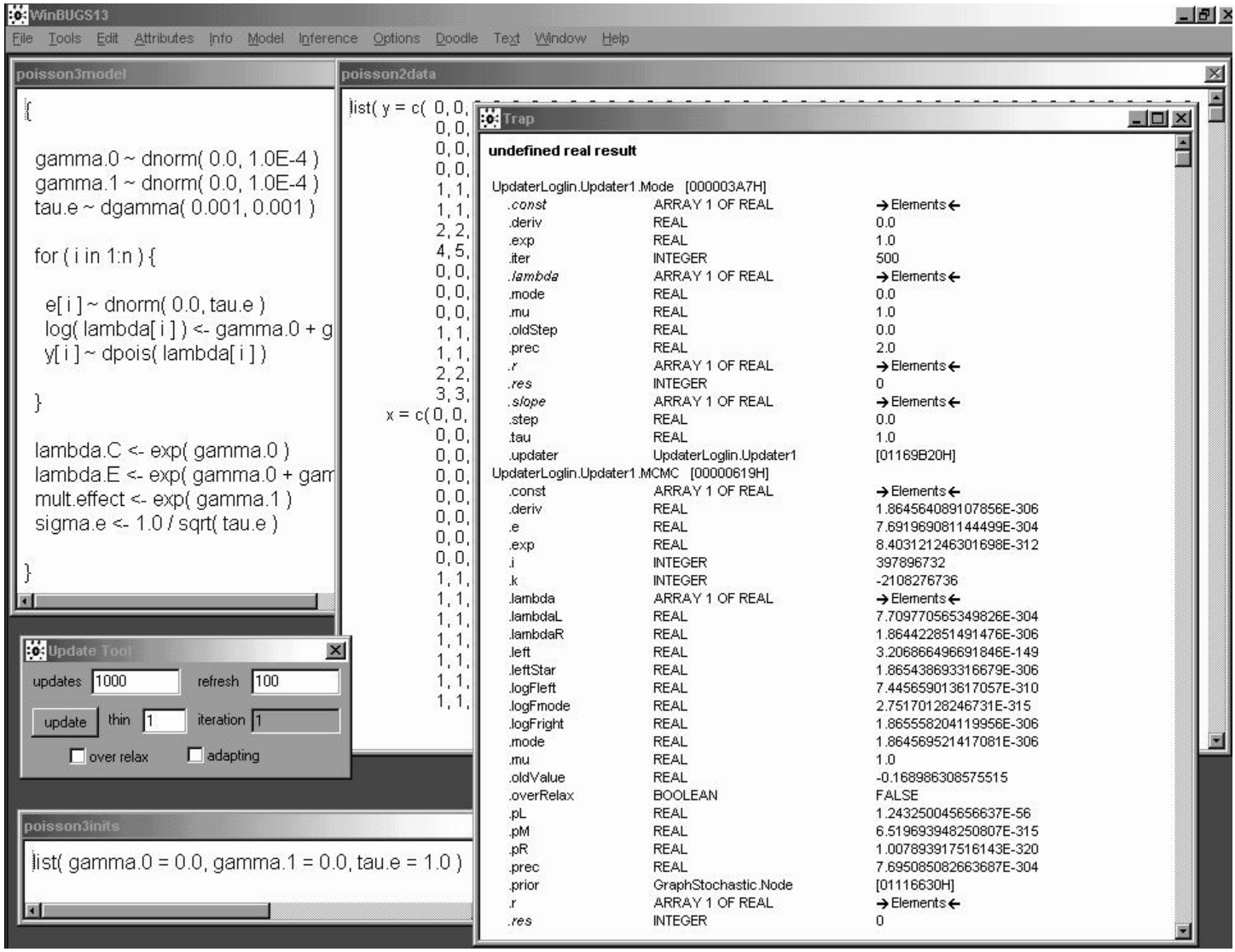
WinBUGS Implementation

The new WinBUGS model is

```
{

  gamma.0 ~ dnorm( 0.0, 1.0E-4 )
  gamma.1 ~ dnorm( 0.0, 1.0E-4 )
  tau.e ~ dgamma( 0.001, 0.001 )

  for ( i in 1:n ) {

    e[ i ] ~ dnorm( 0.0, tau.e )
    log( lambda[ i ] ) <- gamma.0 + gamma.1 * x[ i ] +
      e[ i ]
    y[ i ] ~ dpois( lambda[ i ] )

  }

  lambda.C <- exp( gamma.0 )
  lambda.E <- exp( gamma.0 + gamma.1 )
  mult.effect <- exp( gamma.1 )
  sigma.e <- 1.0 / sqrt( tau.e )

}
```

I again use a **diffuse** $\Gamma(\epsilon, \epsilon)$ prior (with $\epsilon = 0.001$) for the **precision** τ_e of the random effects.

WinBUGS Implementation (continued)



With a model like that in equation (16), there are n **random effects** e_i that need to be sampled as nodes in the graph (the e_i play the role of **auxiliary variables** in the MCMC) along with the fixed effects (γ_0, γ_1) and the variance parameter σ_e^2.

In earlier releases of the software, at least, this made it more crucial to give WinBUGS **good starting values**.

Here WinBUGS release **1.3** has figured out that random draws like $\boxed{1.66 \cdot 10^{-316}}$ result from the **generic** (and quite poor) initial values $(\gamma_0, \gamma_1, \tau_e) = (0.0, 0.0, 1.0)$ and has **refused to continue sampling**.

Sensitivity to Initial Values

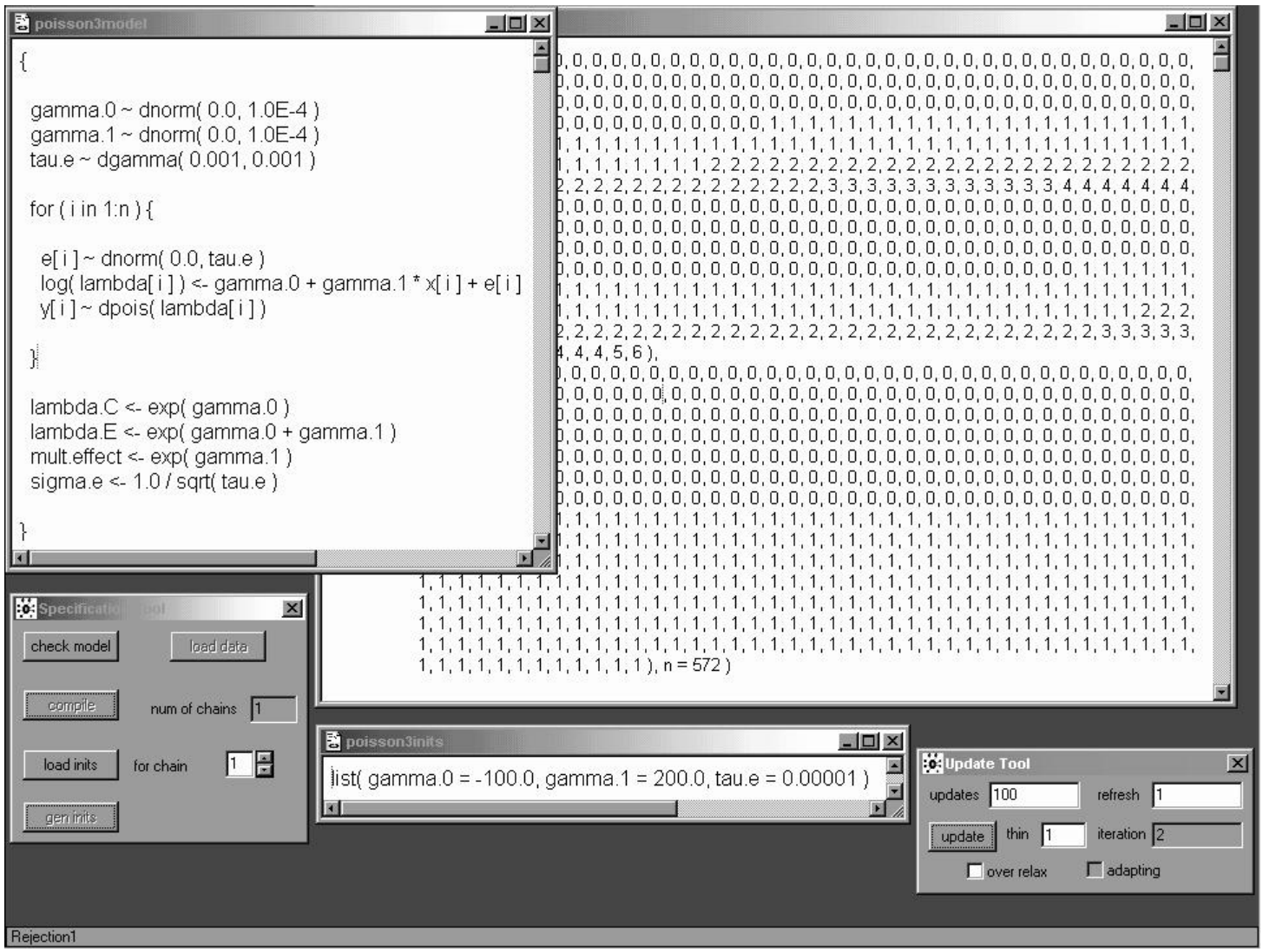
Warning `WinBUGS` can fail, particularly in random-effects models, when you give it initial values that are not very close to the final posterior means; an example in release 1.3 is the REPR model (16) on the IHGA data with the **"generic"** starting values $(\gamma_0, \gamma_1, \tau_e) = (0.0, 0.0, 1.0)$.

When this problem arises there are two ways out in `WinBUGS`: **trial and error**, or a calculation (see below).

NB `MLwiN` does not have this problem—it gets its starting values from **maximum likelihood** (the mode of the likelihood function is often a decent approximation to the mean or mode of the posterior).

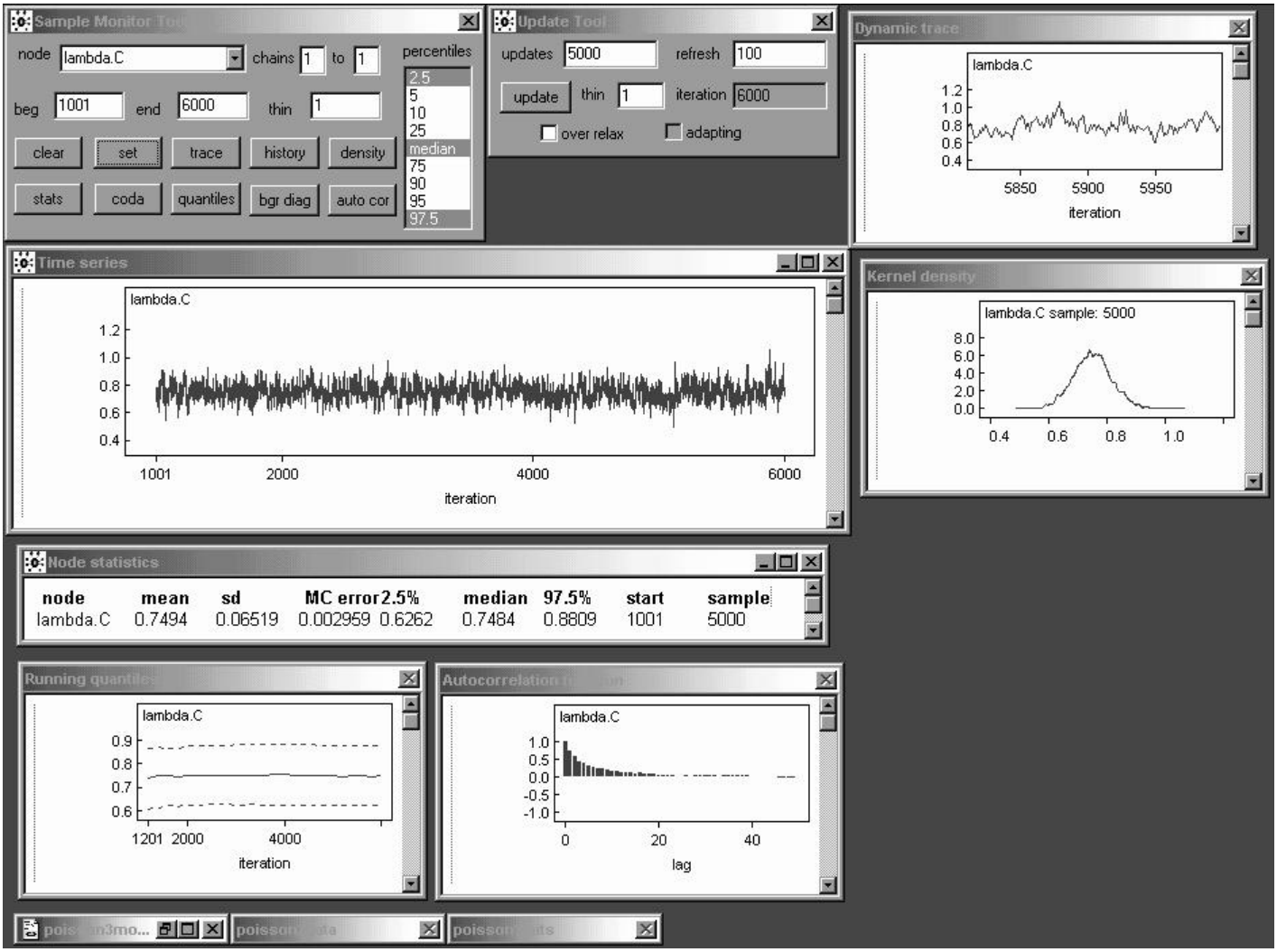
Technical note. To get a decent starting value for τ_e in model (16) you can calculate as follows: renaming the random effects η_i to avoid confusion with the number e, (1) $V(y_i) = V[E(y_i\,|\eta_i)] + E[V(y_i\,|\eta_i)]$, where (2) $(y_i\,|\eta_i) \sim$ Poisson $\left(e^{\gamma_0+\gamma_1 x_i+\eta_i}\right)$, so $E(y_i\,|\eta_i) = V(y_i\,|\eta_i) = e^{\gamma_0+\gamma_1 x_i+\eta_i}$. Then (3) $V[E(y_i\,|\eta_i)] = V\left(e^{\gamma_0+\gamma_1 x_i+\eta_i}\right) = e^{2(\gamma_0+\gamma_1 x_i)} V(e^{\eta_i})$ and $E[V(y_i\,|\eta_i)] = E\left(e^{\gamma_0+\gamma_1 x_i+\eta_i}\right) = e^{\gamma_0+\gamma_1 x_i} E(e^{\eta_i})$. Now (4) e^{η_i} is lognormal with mean 0 and variance σ_e^2 on the log scale, so $E(e^{\eta_i}) = e^{\frac{1}{2}\sigma_e^2}$ and $V(e^{\eta_i}) = e^{\sigma_e^2}\left(e^{\sigma_e^2} - 1\right)$, yielding finally $V(y_i) = e^{2(\gamma_0+\gamma_1 x_i)+\frac{1}{2}\sigma_e^2} + e^{\gamma_0+\gamma_1 x_i+\sigma_e^2}\left(e^{\sigma_e^2} - 1\right)$. (5) Plugging in $x_i = 0$ for the C group, whose sample variance is 1.54, and using the value $\gamma_0 = -0.29$ from runs with previous models, gives an equation for σ_e^2 that can be solved numerically, yielding $\sigma_e^2 \doteq 0.5$ and $\tau_e \doteq 2$.

WinBUGS Implementation (continued)



Interestingly, WinBUGS release 1.4 is able to **sample successfully** with the generic starting values $(\gamma_0, \gamma_1, \tau_e) = (0.0, 0.0, 1.0)$, although of course a **longer burn-in period** would be needed when they're used; you have to try **truly absurd** initial values to get it to fall over, and when it does so the **error message** ("Rejection1") in the lower left corner is **more discreet**.

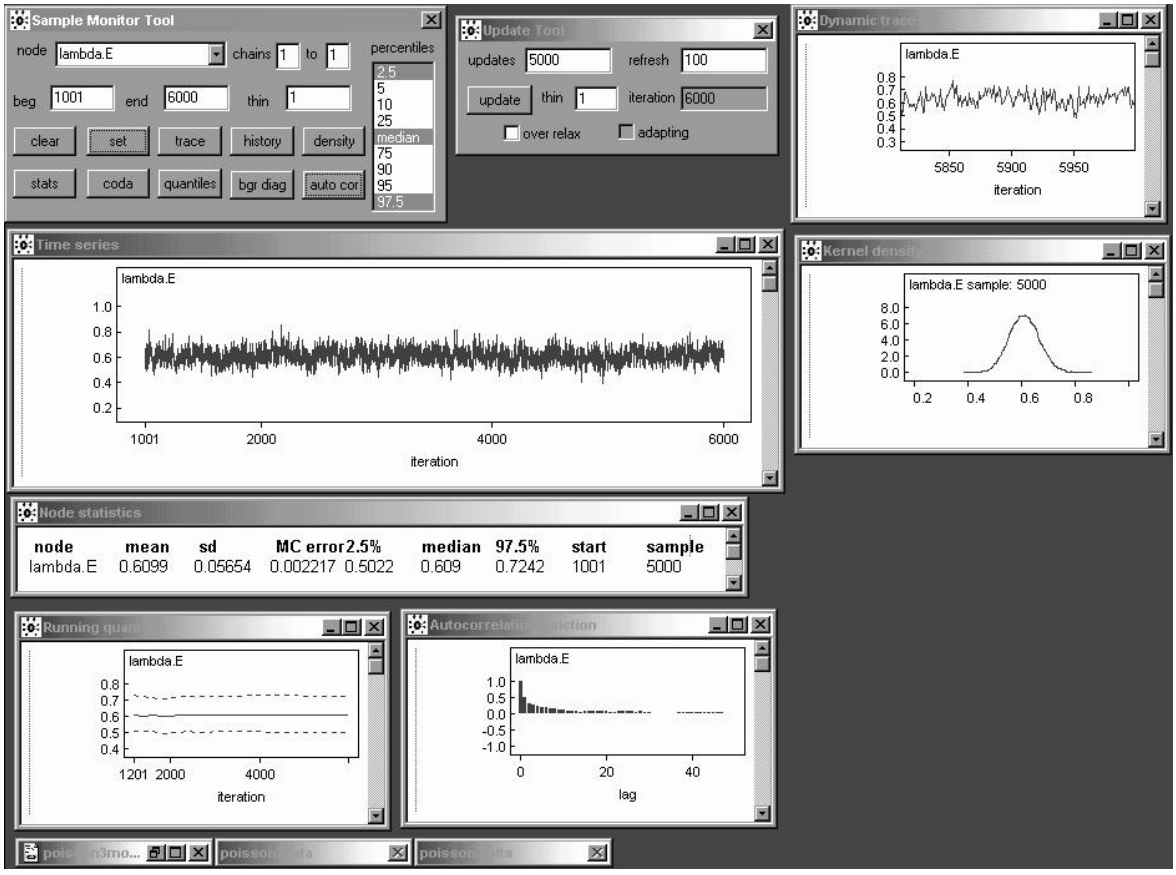
WinBUGS Implementation (continued)

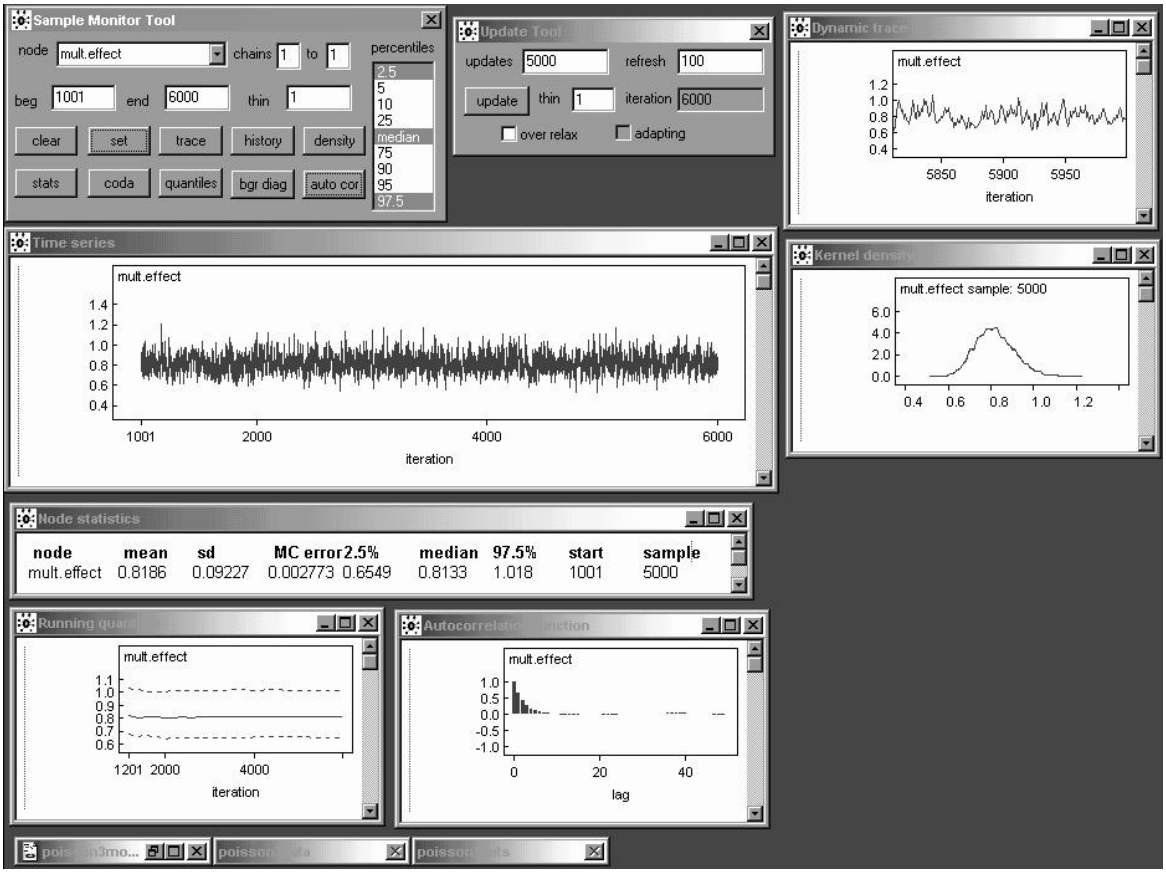


With a **better set of initial values**—$(\gamma_0, \gamma_1, \tau_e) = (-0.058, -0.21, 2.0)$, obtained from (a) the earlier Poisson models (in the case of the regression parameters γ_j) and (b) either a calculation like the one on the bottom of page 29 or trial and error—WinBUGS is able to make progress, although this model takes **a fairly long time to fit** in release 1.4: a burn-in of 1,000 takes 11 seconds at 1.0 PC GHz (the code runs about twice as fast in release 1.3 for some reason).

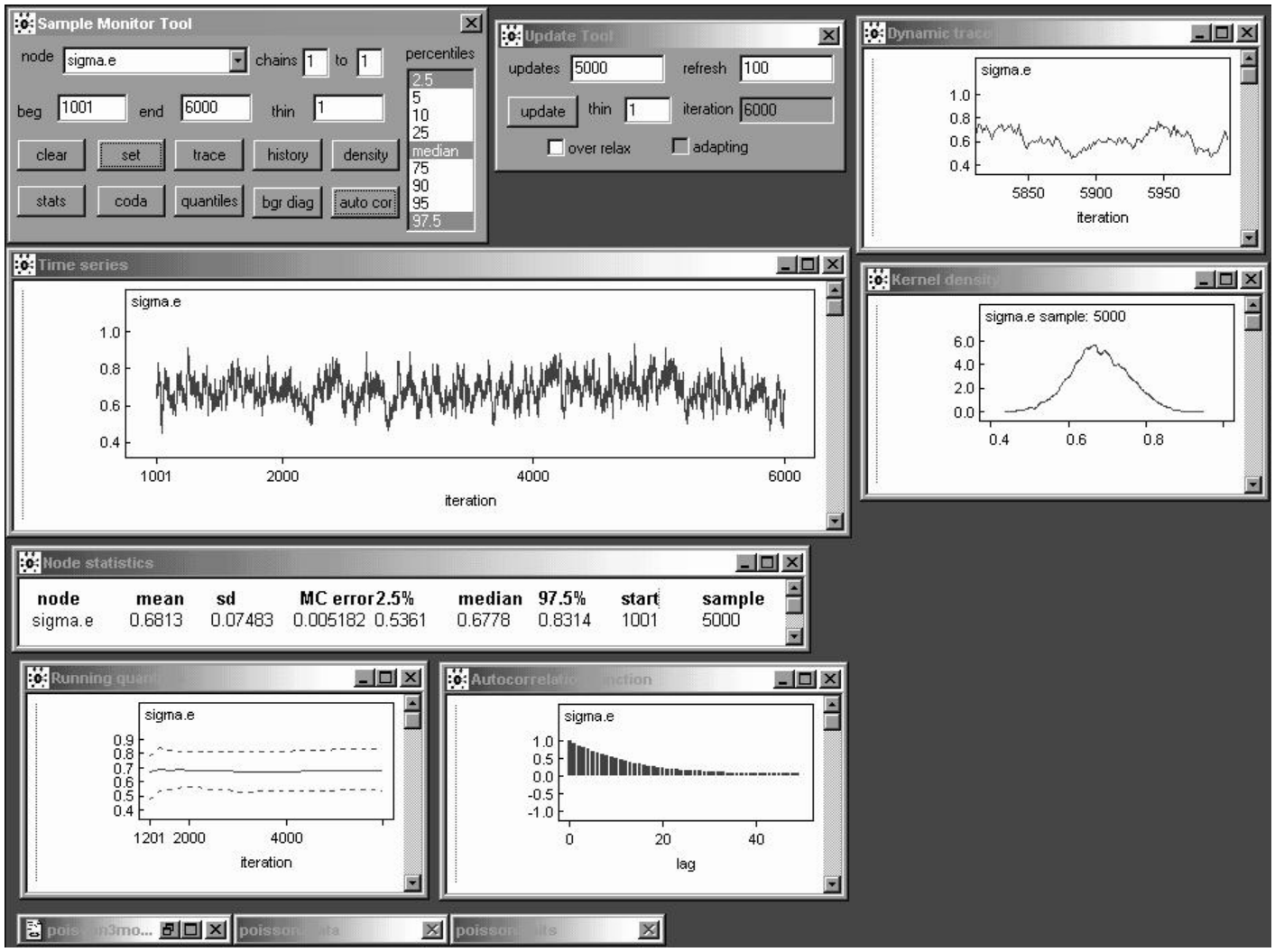
A monitoring run of **5,000** iterations reveals that the random effects make everything **mix more slowly**: λ_C (this page) and λ_E and the multiplicative effect (next page) all behave like AR_1 series with $\widehat{\rho}_1 \doteq 0.7$, 0.5, and 0.6, respectively.

WinBUGS Implementation (continued)





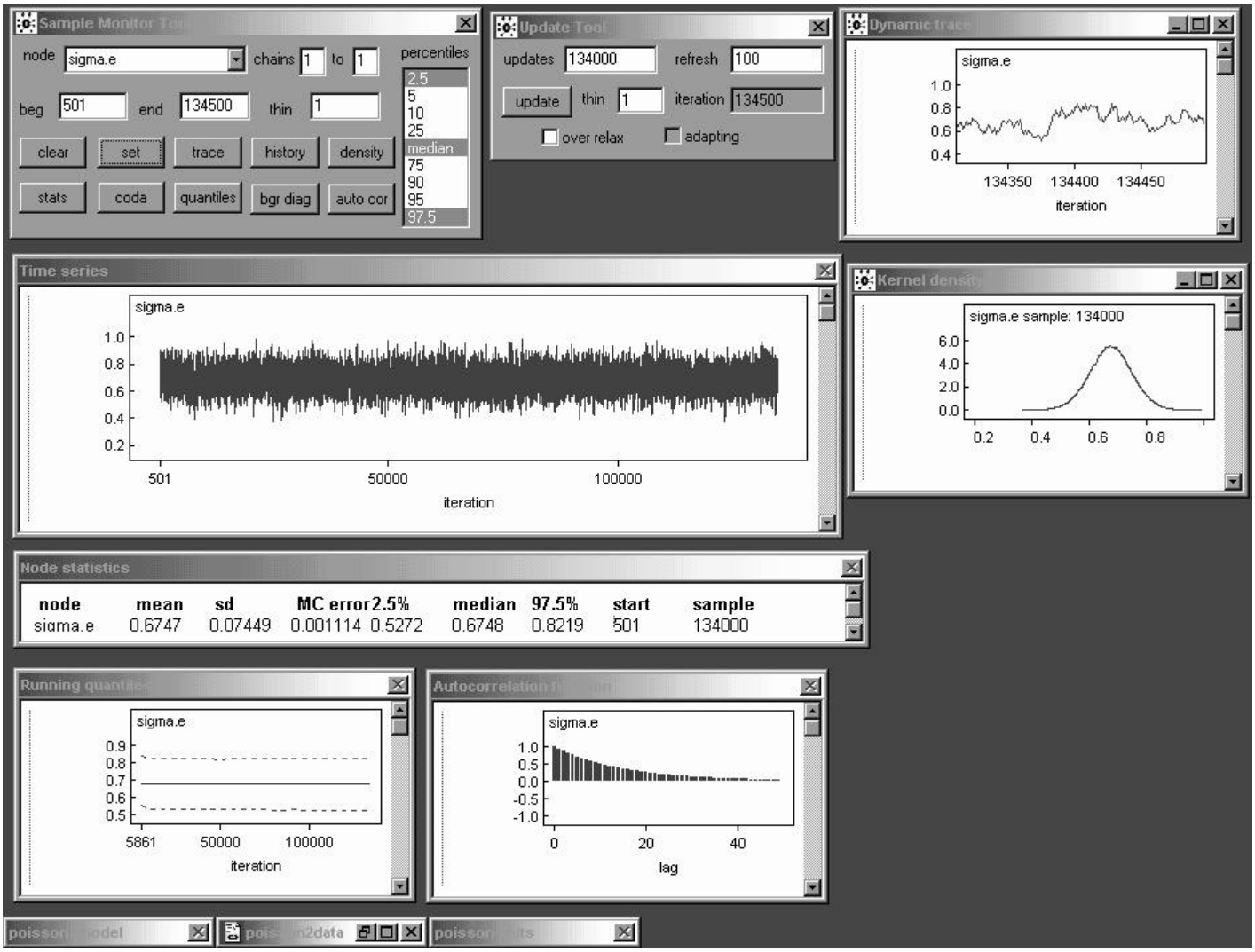
WinBUGS Implementation (continued)



Learning about σ_e in this model is **slow**: its autocorrelation function is that of an AR_1 with a **high value** of $\widehat{\rho}_1$ (equation (55) on page 76 of part 3 of the lecture notes gives $\widehat{\rho}_1 \doteq \mathbf{0.92}$).

The MCSE of the posterior mean for σ_e based on 5,000 draws is **0.005182**—to get this down to (say) **0.001** I need to increase the length of the monitoring run by a factor of $\left(\frac{0.005182}{0.001}\right)^2 \doteq \mathbf{26.9}$, meaning a total run of about $(26.9)(5,000) \doteq 134,000$ iterations (this takes about **half an hour** at 1 PC GHz).

WinBUGS Implementation (continued)



There is **clear evidence** that σ_e is **far from 0**—its posterior mean and SD are estimated as **0.675** (with an MCSE of about **0.001** after 134,000 iterations) and **0.074**, respectively—meaning that the **model expansion** from (11) to (16) was **amply justified**.

REPR Model Results

(Another way to achieve the goal of describing the extra-Poisson variability would be to fit different **negative binomial** distributions to the observed counts in the C and E groups—the negative binomial is a **gamma mixture of Poissons**, and the gamma and lognormal distributions often fit long-tailed data about equally well, so you would not be surprised to find that the two approaches give **similar results**.)

Table 4.5. Comparison of inferential conclusions about the multiplicative effect parameter e^{γ_1} from the fixed-effects and random-effects Poisson regression models.

Model	Posterior Mean	Posterior SD	Central 95% Interval
FEPR	0.816	0.0735	(0.683, 0.969)
REPR	0.830	0.0921	(0.665, 1.02)

Table 4.5 compares the REPR model inferential results with those from model (11), which could also be called a **fixed-effects Poisson regression** (FEPR) model.

The "error" SD σ_e has posterior mean **0.68**, give or take about 0.07 (on the $\log(\lambda)$ scale), corresponding to substantial extra-Poisson variability, which translates into **increased uncertainty** about the multiplicative effect parameter e^{γ_1}.

I'll argue later that the REPR model **fits the data well**, so the conclusion I'd publish from these data is that IHGA reduces the average number of hospitalizations per two years by about $100\,(1 - 0.083)\% = \boxed{\mathbf{17\%}}$ give or take about 9% (ironically this conclusion is similar to that from the Gaussian model, but this is **coincidence**).

References

Bryk AS, Raudenbush SW (1992). *Hierarchical Linear Models: Applications and Data Analysis Methods*. London: Sage.

Carlin BP, Louis TA (1996). *Bayes and Empirical Bayes Methods for Data Analysis*. London: Chapman & Hall.

Draper D, Gaver D, Goel P, Greenhouse J, Hedges L, Morris C, Tucker J, Waternaux C (1993a). *Combining Information: Statistical Issues and Opportunities for Research*. Contemporary Statistics Series, No. 1. American Statistical Association, Alexandria VA.

Draper D, Hodges JS, Mallows CL, Pregibon D (1993b). Exchangeability and data analysis (with discussion). *Journal of the Royal Statistical Society, Series A*, **156**, 9–37.

Draper D (1995). Inference and hierarchical modeling in the social sciences (with discussion). *Journal of Educational and Behavioral Statistics*, **20**, 115–147, 233–239.

Hedges LV, Olkin I (1985). *Statistical Methods for Meta-Analysis*. New York: Academic Press.

Morris CN (1983). Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association*, **78**, 47–59.

Morris CN (1988). Determining the accuracy of Bayesian empirical Bayes estimators in the familiar exponential families. In *Proceedings of the Fourth Purdue Symposium on Statistical Decision Theory and Related Topics IV, part 1.*, SS Gupta, JO Berger, eds. New York: Springer-Verlag, 251–263.

Raudenbush SW (1984). Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 19 experiments. *Journal of Educational Psychology*, **76**, 85–97.