Bayesian Hierarchical/ Multilevel and Latent-Variable (Random-Effects) Modeling

1: Formulation of Bayesian models and fitting them with MCMC in WinBUGS

David Draper

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

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

National University of Ireland, Galway
1 Jun 2010

Continuous Outcomes

Case Study: *Measurement of physical constants*. What used to be called the National Bureau of Standards (NBS) in Washington, DC, conducts extremely high precision measurement of physical constants, such as the actual weight of so-called **check-weights** that are supposed to serve as reference standards (like the official kg).

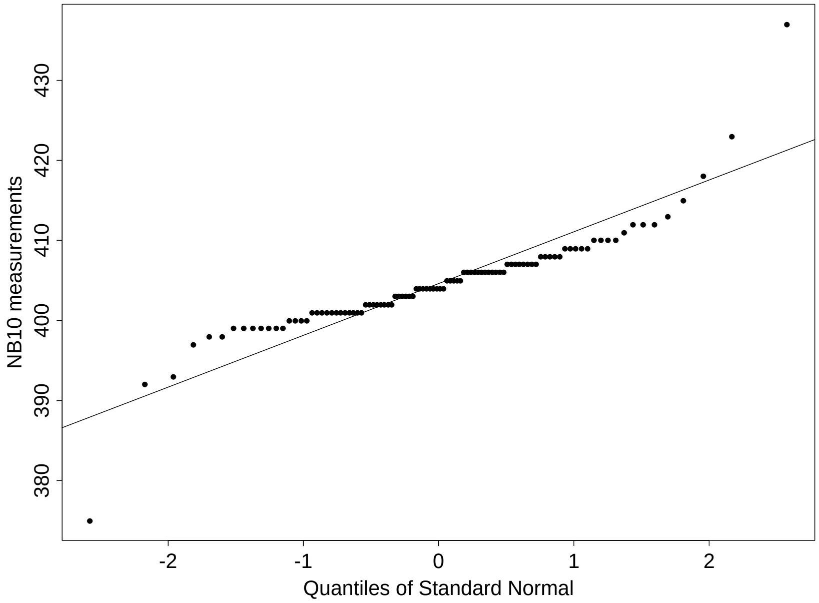
In 1962–63, for example, $n = 100$ weighings (listed below) of a block of metal called **NB10**, which was supposed to weigh exactly 10g, were made under conditions **as close to IID as possible** (Freedman et al., 1998).

Value	375	392	393	397	398	399	400	401
Frequency	1	1	1	1	2	7	4	12
Value	402	403	404	405	406	407	408	409
Frequency	8	6	9	5	12	8	5	5
Value	410	411	412	413	415	418	423	437
Frequency	4	1	3	1	1	1	1	1

Q: (a) How much does NB10 **really weigh**? (b) How certain are you given the data that the true weight of NB10 is **less than** (say) 405.25? And (c) How accurately can you **predict** the 101st measurement?

The graph below is a **normal qqplot** of the 100 measurements $y = (y_1, \ldots, y_n)$, which have a mean of $\bar{y} = 404.6$ (the units are **micrograms below 10g**) and an SD of $s = 6.5$.

NB10 Data



Evidently it's plausible in answering these questions to assume **symmetry** of the "underlying distribution" F in de Finetti's Theorem.

One standard choice, for instance, is the **Gaussian:**

$$\begin{aligned} (\mu, \sigma^2) & \sim & p(\mu, \sigma^2) \\ (Y_i | \mu, \sigma^2) & \overset{\text{IID}}{\sim} & N\big(\mu, \sigma^2\big) \, . \end{aligned} \qquad (1)$$

Here $N\big(\mu, \sigma^2\big)$ is the familiar **normal density**

$$p(y_i | \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2} \left(\frac{y_i - \mu}{\sigma} \right)^2 \right] . \qquad (2)$$

Gaussian Modeling

Even though you can see from the previous graph that (79) is **not a good model** for the NB10 data, I'm going to fit it to the data for practice in working with the normal distribution from a Bayesian point of view (later we'll **improve** upon the Gaussian).

(79) is more **complicated** than the models in the AMI and LOS case studies because the parameter θ here is a **vector**: $\theta = (\mu, \sigma^2)$.

To warm up for this new complexity let's first consider a **cut-down version of the model** in which we pretend that σ is known to be $\sigma_0 = 6.5$ (the sample SD).

This **simpler model** is then

$$\left\{ \begin{array}{ccc} \mu & \sim & p(\mu) \\ (Y_i|\mu) & \stackrel{\text{IID}}{\sim} & N\left(\mu, \sigma_0^2\right) \end{array} \right\}. \qquad (3)$$

The **likelihood function** in this model is

$$\begin{aligned} l(\mu|y) &= \prod_{i=1}^n \frac{1}{\sigma_0 \sqrt{2\pi}} \exp\left[-\frac{1}{2\sigma_0^2} (y_i - \mu)^2 \right] \\ &= c \exp\left[-\frac{1}{2\sigma_0^2} \sum_{i=1}^n (y_i - \mu)^2 \right] \qquad (4) \\ &= c \exp\left[-\frac{1}{2\sigma_0^2} \left(\sum_{i=1}^n y_i^2 - 2\,\mu \sum_{i=1}^n y_i + n\mu^2 \right) \right] \\ &= c \exp\left[-\frac{1}{2\left(\frac{\sigma_0^2}{n}\right)} (\mu - \bar{y})^2 \right]. \end{aligned}$$

Thus the likelihood function, when thought of as a **density** for μ, is a **normal distribution** with mean $\bar{y}$ and SD $\frac{\sigma_0}{\sqrt{n}}$.

Gaussian Modeling (continued)

Notice that this SD is the same as the frequentist **standard error** for $\bar{Y}$ based on an IID sample of size n from the $N\left(\mu, \sigma_0^2\right)$ distribution.

(82) also shows that the sample mean $\bar{y}$ is a **sufficient statistic** for μ in model (81).

In finding the conjugate prior for μ it would be nice if the **product of two normal distributions is another normal distribution**, because that would demonstrate that the conjugate prior is normal.

Suppose therefore, to see where it leads, that the **prior for** μ is (say) $p(\mu) = N\left(\mu_0, \sigma_\mu^2\right)$.

Then **Bayes' Theorem** would give

$$\begin{aligned} p(\mu|y) &= c\, p(\mu)\, l(\mu|y) \qquad (5) \\ &= c \exp\left[-\frac{1}{2\sigma_\mu^2}(\mu-\mu_0)^2\right] \exp\left[-\frac{n}{2\sigma_0^2}(\mu-\bar{y})^2\right] \\ &= c \exp\left\{-\frac{1}{2}\left[\frac{(\mu-\mu_0)^2}{\sigma_\mu^2} + \frac{n(\mu-\bar{y})^2}{\sigma_0^2}\right]\right\}, \end{aligned}$$

and we want this to **be of the form**

$$\begin{aligned} p(\mu|y) &= c \exp\left\{-\frac{1}{2}\left[A(\mu-B)^2 + C\right]\right\} \\ &= c \exp\left\{-\frac{1}{2}\left[A\mu^2 - 2AB\mu + (AB^2 + C)\right]\right\} \qquad (6) \end{aligned}$$

for some B, C, and $A > 0$.

`Maple` can help **see if this works**:

```
> collect( ( mu - mu0 )^2 / sigmamu^2 +
    n * ( mu - ybar )^2 / sigma0^2, mu );


                                                                2          2
/   1          n    \   2   /     mu0        n ybar \        mu0      n ybar
|-------- + -------| mu  + |-2 -------- - 2 -------| mu + -------- + -------
|       2         2|       |          2            2|            2          2
\sigmamu    sigma0 /       \   sigmamu      sigma0 /      sigmamu    sigma0
```

Gaussian Modeling

Matching coefficients for A and B (we don't really care about C) gives

$$A = \frac{1}{\sigma_\mu^2} + \frac{n}{\sigma_0^2} \quad \text{and} \quad B = \frac{\frac{\mu_0}{\sigma_\mu^2} + \frac{n\bar{y}}{\sigma_0^2}}{\frac{1}{\sigma_\mu^2} + \frac{n}{\sigma_0^2}}. \tag{7}$$

Since $A > 0$ this demonstrates two things: (1) the **conjugate prior** for μ in model (81) is **normal**, and (2) the **conjugate updating rule** (when σ_0 is assumed known) is

$$\left\{ \begin{array}{c} \mu \sim N\left(\mu_0, \sigma_\mu^2\right) \\ (Y_i|\mu) \overset{\text{IID}}{\sim} N\left(\mu, \sigma_0^2\right), \\ i = 1, \ldots, n \end{array} \right\} \rightarrow (\mu|y) = (\mu|\bar{y}) = N\left(\mu_*, \sigma_*^2\right), \tag{8}$$

where the **posterior mean and variance** are given by

$$\mu_* = B = \frac{\left(\frac{1}{\sigma_\mu^2}\right)\mu_0 + \left(\frac{n}{\sigma_0^2}\right)\bar{y}}{\frac{1}{\sigma_\mu^2} + \frac{n}{\sigma_0^2}} \quad \text{and} \quad \sigma_*^2 = A^{-1} = \frac{1}{\frac{1}{\sigma_\mu^2} + \frac{n}{\sigma_0^2}}. \tag{9}$$

It becomes useful in understanding the meaning of these expressions to define the $\boxed{\textbf{precision}}$ of a distribution, which is just the **reciprocal** of its variance: whereas the variance and SD scales measure **uncertainty**, the precision scale quantifies **information** about an unknown.

With this convention (87) has a series of **intuitive interpretations**, as follows:

- The **prior**, considered as an **information source**, is Gaussian with mean μ_0, variance σ_μ^2, and **precision** $\frac{1}{\sigma_\mu^2}$, and when viewed as a data set consists of n_0 (to be determined below) observations;

- The **likelihood**, considered as an **information source**, is Gaussian with mean $\bar{y}$, variance $\frac{\sigma_0^2}{n}$, and **precision** $\frac{n}{\sigma_0^2}$, and when viewed as a data set consists of n observations;

Gaussian Modeling (continued)

• The **posterior**, considered as an **information source**, is Gaussian, and the posterior mean is a **weighted average** of the prior mean and data mean, with weights given by the **prior** and **data precisions**;

• The **posterior precision** (the reciprocal of the posterior variance) is just the **sum** of the prior and data precisions (this is why people invented the idea of precision—on this scale **knowledge** about μ in model (81) is **additive**); and

• **Rewriting** μ_* as

$$\mu_* = \frac{\left(\frac{1}{\sigma_\mu^2}\right)\mu_0 + \left(\frac{n}{\sigma_0^2}\right)\bar{y}}{\frac{1}{\sigma_\mu^2} + \frac{n}{\sigma_0^2}} = \frac{\left(\frac{\sigma_0^2}{\sigma_\mu^2}\right)\mu_0 + n\bar{y}}{\frac{\sigma_0^2}{\sigma_\mu^2} + n}, \qquad (10)$$

you can see that the **prior sample size** is

$$n_0 = \frac{\sigma_0^2}{\sigma_\mu^2} = \frac{1}{\left(\frac{\sigma_\mu}{\sigma_0}\right)^2}, \qquad (11)$$

which makes sense: the **bigger** σ_μ is in relation to σ_0, the **less prior information** is being incorporated in the conjugate updating (86).

Bayesian inference with multivariate θ. Returning now to (79) with σ^2 unknown, (as mentioned above) this model has a ($p = 2$)-dimensional **parameter vector** $\theta = (\mu, \sigma^2)$.

When $p > 1$ you can still use Bayes' Theorem directly to obtain the **joint posterior distribution**,

$$\begin{aligned} p(\theta|y) &= p(\mu, \sigma^2|y) = c\,p(\theta)\,l(\theta|y) \\ &= c\,p(\mu, \sigma^2)\,l(\mu, \sigma^2|y), \end{aligned} \qquad (12)$$

Multivariate Unknown θ

where $y = (y_1, \ldots, y_n)$, although making this calculation directly requires a p-dimensional **integration** to evaluate the normalizing constant c; for example, in this case

$$\begin{aligned} c &= [p(y)]^{-1} = \left(\iint p(\mu, \sigma^2, y) \, d\mu \, d\sigma^2 \right)^{-1} \\ &= \left(\iint p(\mu, \sigma^2) \, l(\mu, \sigma^2 | y) \, d\mu \, d\sigma^2 \right)^{-1} . \qquad (13) \end{aligned}$$

Usually, however, you'll be more interested in the **marginal posterior distributions**, in this case $p(\mu|y)$ and $p(\sigma^2|y)$.

Obtaining these requires p integrations, each of dimension $(p-1)$, a process that people refer to as **marginalization** or **integrating out the nuisance parameters**—for example,

$$p(\mu|y) = \int_0^\infty p(\mu, \sigma^2 | y) \, d\sigma^2 \ . \qquad (14)$$

Predictive distributions also involve a p-dimensional integration: for example, with $y = (y_1, \ldots, y_n)$,

$$\begin{aligned} p(y_{n+1}|y) &= \iint p(y_{n+1}, \mu, \sigma^2 | y) \, d\mu \, d\sigma^2 \qquad (15) \\ &= \iint p(y_{n+1} | \mu, \sigma^2) \, p(\mu, \sigma^2 | y) \, d\mu \, d\sigma^2 . \end{aligned}$$

And, finally, if you're interested in a **function of the parameters**, you have some more hard integrations ahead of you.

For instance, suppose you wanted the posterior distribution for the **coefficient of variation** $\lambda = g_1(\mu, \sigma^2) = \frac{\sqrt{\sigma^2}}{\mu}$ in model (79).

Multivariate Unknown θ

Then one fairly direct way to get this posterior (e.g., Bernardo and Smith, 1994) is to (a) introduce a **second function** of the parameters, say $\eta = g_2(\mu, \sigma^2)$, such that the mapping $f = (g_1, g_2)$ from (μ, σ^2) to (λ, η) is **invertible**; (b) compute the joint posterior for (λ, η) through the usual **change-of-variables formula**

$$p(\lambda, \eta|y) = p_{\mu,\sigma^2}\big[f^{-1}(\lambda, \eta)|y\big] \, |J_{f^{-1}}(\lambda, \eta)| \, , \qquad (16)$$

where $p_{\mu,\sigma^2}(\cdot, \cdot|y)$ is the joint posterior for μ and σ^2 and $|J_{f^{-1}}|$ is the **determinant** of the **Jacobian** of the inverse transformation; and (c) **marginalize** in λ by integrating out η in $p(\lambda, \eta|y)$, in a manner analogous to (92).

Here, for instance, $\eta = g_2(\mu, \sigma^2) = \mu$ would create an invertible f, with **inverse** defined by $(\mu = \eta, \sigma^2 = \lambda^2 \eta^2)$; the **Jacobian determinant** comes out $2\lambda\eta^2$ and (94) becomes $p(\lambda, \eta|y) = 2\lambda\eta^2 \, p_{\mu,\sigma^2}(\eta, \lambda^2\eta^2|y)$.

This process involves **two integrations**, one (of dimension p) to get the normalizing constant that defines (94) and one (of dimension $(p-1)$) to get rid of η.

You can see that when p is a lot bigger than 2 all these integrals may create **severe computational problems**—this has been the **big stumbling block** for applied Bayesian work for a long time.

More than 200 years ago **Laplace** (1774)—perhaps the second applied Bayesian in history (after Bayes himself)—developed, as one avenue of solution to this problem, what people now call **Laplace approximations** to high-dimensional integrals of the type arising in Bayesian calculations (see, e.g., Tierney and Kadane, 1986).

Starting in the next case study after this one, we'll use another, computationally intensive, **simulation-based** approach: **Markov chain Monte Carlo** (MCMC).

Gaussian Modeling

Back to model (79). The conjugate prior for $\theta = \left(\mu, \sigma^2 \right)$ in this model (e.g., Gelman et al., 2003) turns out to be most simply described **hierarchically**:

$$\begin{aligned} \sigma^2 & \sim \text{SI-}\chi^2(\nu_0, \sigma_0^2) \\ (\mu | \sigma^2) & \sim N\left(\mu_0, \frac{\sigma^2}{\kappa_0} \right). \end{aligned} \tag{17}$$

Here saying that $\sigma^2 \sim \text{SI-}\chi^2(\nu_0, \sigma_0^2)$, where SI stands for **scaled inverse**, amounts to saying that the precision $\tau = \frac{1}{\sigma^2}$ follows a **scaled** χ^2 distribution with parameters ν_0 and σ_0^2.

The scaling is chosen so that σ_0^2 can be interpreted as a **prior estimate** of σ^2, with ν_0 the **prior sample size** of this estimate (i.e., **think of a prior data set with** ν_0 **observations and sample SD** σ_0).

Since χ^2 is a special case of the Gamma distribution, SI-χ^2 must be a special case of the **inverse Gamma** family—its **density** (see Gelman et al., 2003, Appendix A) is

$$\sigma^2 \sim \text{SI-}\chi^2(\nu_0, \sigma_0^2) \leftrightarrow \tag{18}$$

$$p(\sigma^2) = \frac{\left(\frac{1}{2} \nu_0 \right)^{\frac{1}{2} \nu_0}}{\Gamma\left(\frac{1}{2} \nu_0 \right)} \left(\sigma_0^2 \right)^{\frac{1}{2} \nu_0} \left(\sigma^2 \right)^{-\left(1 + \frac{1}{2} \nu_0 \right)} \exp\left(\frac{-\nu_0 \, \sigma_0^2}{2 \sigma^2} \right).$$

As may be verified with `Maple`, this distribution has **mean** (provided that $\nu_0 > 2$) and **variance** (provided that $\nu_0 > 4$) given by

$$E\left(\sigma^2 \right) = \frac{\nu_0}{\nu_0 - 2} \sigma_0^2 \quad \text{and} \quad V\left(\sigma^2 \right) = \frac{2 \nu_0^2}{(\nu_0 - 2)^2 (\nu_0 - 4)} \sigma_0^4. \tag{19}$$

Gaussian Modeling (continued)

The parameters μ_0 and κ_0 in the second level of the prior model (95), $(\mu|\sigma^2) \sim N\Big(\mu_0, \frac{\sigma^2}{\kappa_0}\Big)$, have **simple parallel interpretations** to those of σ_0^2 and ν_0: μ_0 is the **prior estimate** of μ, and κ_0 is the **prior effective sample size** of this estimate.

The **likelihood function** in model (79), with **both** μ and σ^2 **unknown**, is

$$
\begin{aligned}
l(\mu, \sigma^2|y) &= c \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(y_i - \mu)^2\right] \\
&= c \left(\sigma^2\right)^{-\frac{1}{2}n} \exp\left[-\frac{1}{2\sigma^2} \sum_{i=1}^{n}(y_i - \mu)^2\right] \qquad (20) \\
&= c \left(\sigma^2\right)^{-\frac{1}{2}n} \exp\left[-\frac{1}{2\sigma^2} \left(\sum_{i=1}^{n} y_i^2 - 2\,\mu \sum_{i=1}^{n} y_i + n\mu^2\right)\right].
\end{aligned}
$$

The **expression in brackets** in the last line of (98) is

$$
\begin{aligned}
[\ \cdot\ ] &= -\frac{1}{2\sigma^2}\left[\sum_{i=1}^{n} y_i^2 + n(\mu - \bar{y})^2 - n\bar{y}^2\right] \qquad (21) \\
&= -\frac{1}{2\sigma^2}\left[n(\mu - \bar{y})^2 + (n-1)s^2\right],
\end{aligned}
$$

where $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2$ is the **sample variance**. Thus

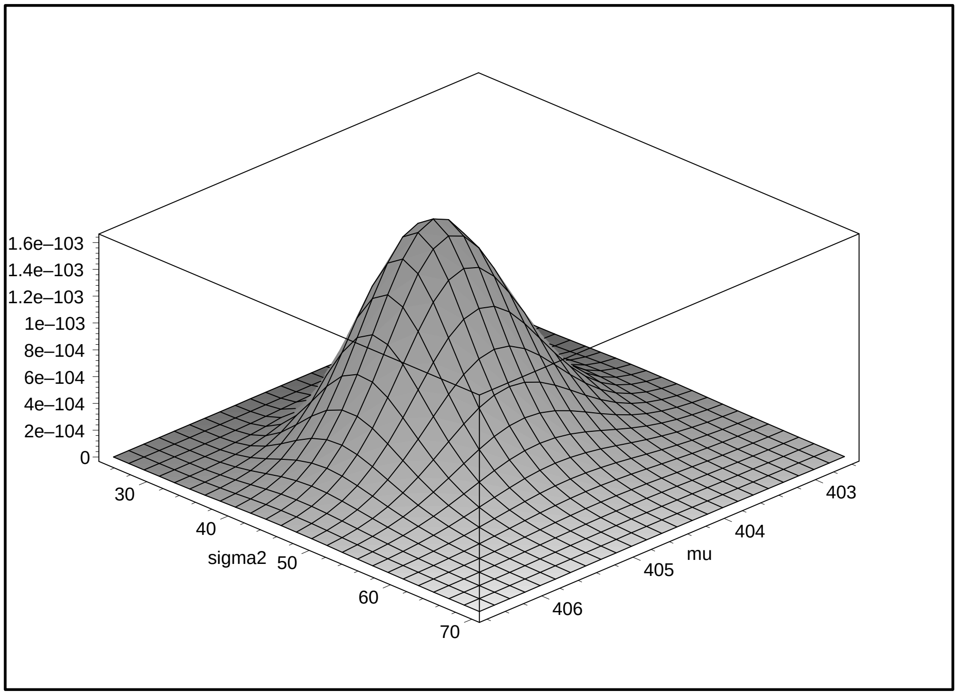
$$
l(\mu, \sigma^2|y) = c \left(\sigma^2\right)^{-\frac{1}{2}n} \exp\left\{-\frac{1}{2\sigma^2}\left[n(\mu - \bar{y})^2 + (n-1)s^2\right]\right\},
$$

and it's clear that the **vector** $\left(\bar{y}, s^2\right)$ is **sufficient** for $\theta = \left(\mu, \sigma^2\right)$ in this model, i.e., $l(\mu, \sigma^2|y) = l(\mu, \sigma^2|\bar{y}, s^2)$.

Gaussian Analysis

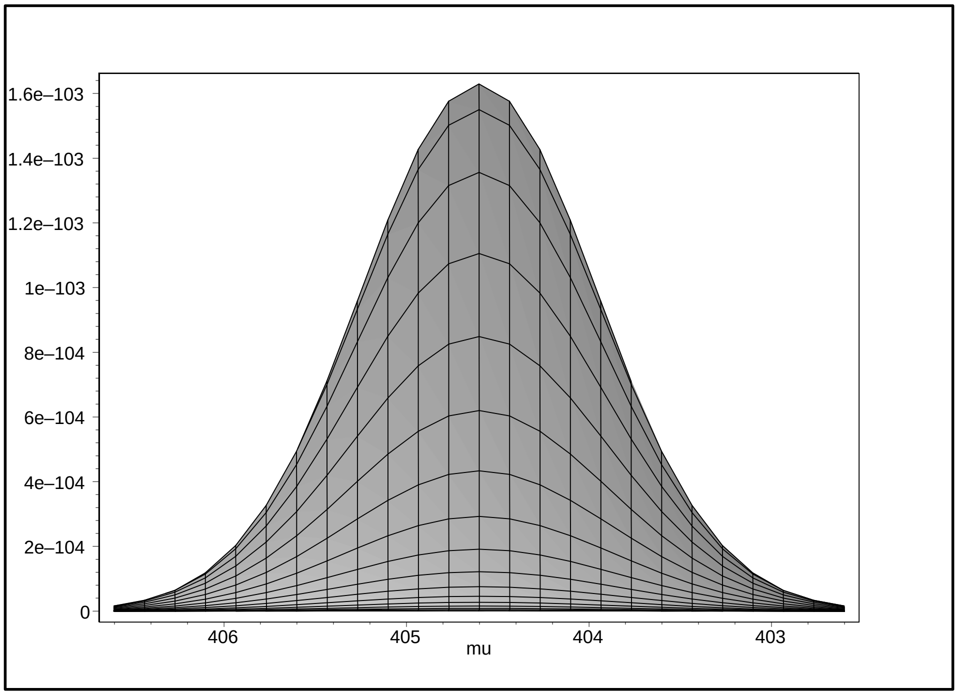
`Maple` can be used to make **3D** and **contour plots** of this likelihood function with the NB10 data:

```
> l := ( mu, sigma2, ybar, s2, n ) -> sigma2^( - n / 2 ) *
    exp( - ( n * ( mu - ybar )^2 + ( n - 1 ) * s2 ) / ( 2 * sigma2 ) );

l := (mu, sigma2, ybar, s2, n) ->

                                                2
          (- 1/2 n)           n (mu - ybar)  + (n - 1) s2
    sigma2          exp(- 1/2 ---------------------------)
                                           sigma2

> plotsetup( x11 );

> plot3d( l( mu, sigma2, 404.6, 42.25, 100 ), mu = 402.6 .. 406.6,
    sigma2 = 25 .. 70 );
```

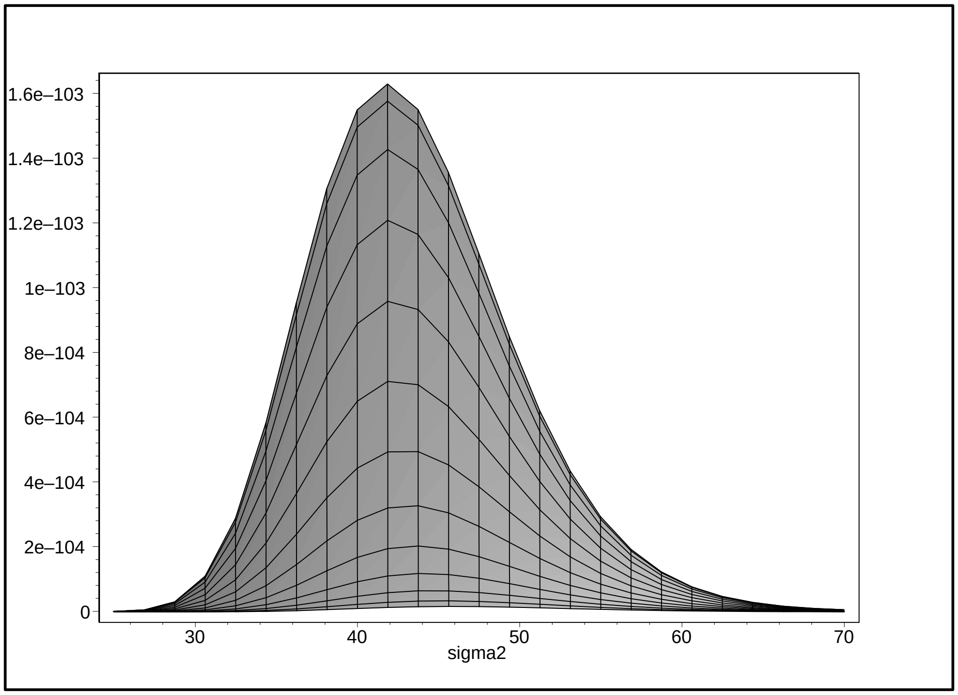


You can use the mouse to **rotate** 3D plots and get **other useful views** of them:

Gaussian Analysis



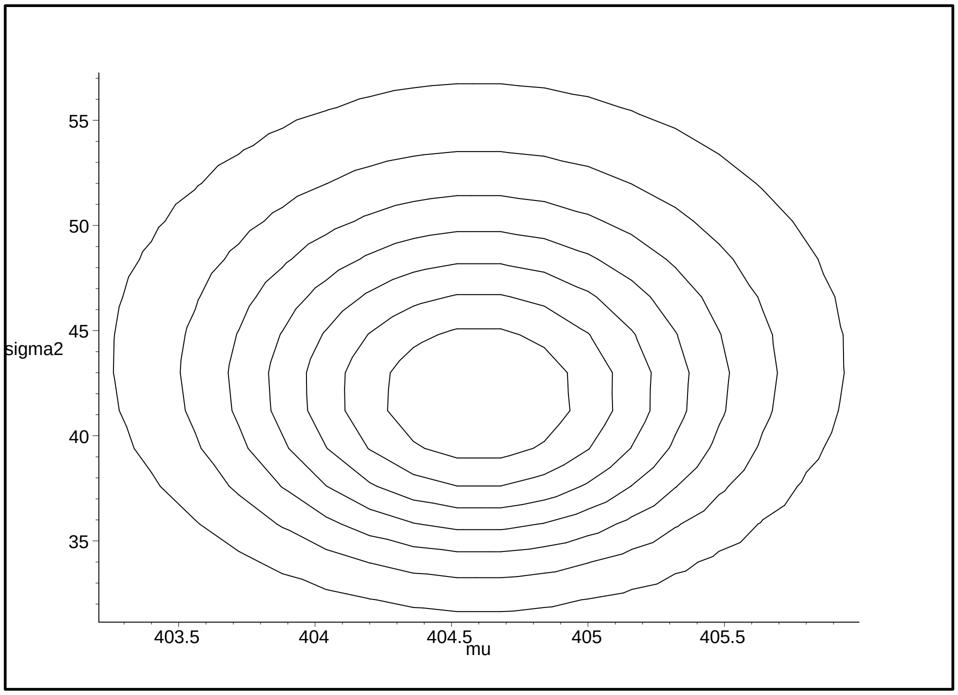
The **projection** or **shadow plot** of μ looks a lot like a **normal** (or maybe a t) distribution.



And the shadow plot of σ^2 looks a lot like a **Gamma** (or maybe an **inverse Gamma**) distribution.

Gaussian Analysis

```
> plots[ contourplot ]( 10^100 * l( mu, sigma2, 404.6, 42.25, 100 ),
    mu = 402.6 .. 406.6, sigma2 = 25 .. 70, color = black );
```



The **contour plot** shows that μ and σ^2 are **uncorrelated** in the likelihood distribution, and the **skewness** of the marginal distribution of σ^2 is also evident.

Posterior analysis. Having adopted the **conjugate prior** (95), what I'd like next is simple expressions for the **marginal posterior distributions** $p(\mu|y)$ and $p(\sigma^2|y)$ and for **predictive distributions** like $p(y_{n+1}|y)$.

Fortunately, in model (79) all of the **integrations** (such as (92) and (93)) may be done **analytically** (see, e.g., Bernardo and Smith 1994), yielding the following results:

$$\begin{aligned}
(\sigma^2|y,\mathcal{G}) &\sim \text{SI-}\chi^2(\nu_n,\sigma_n^2), \\
(\mu|y,\mathcal{G}) &\sim t_{\nu_n}\left(\mu_n,\frac{\sigma_n^2}{\kappa_n}\right), \quad \text{and} \qquad (22)\\
(y_{n+1}|y,\mathcal{G}) &\sim t_{\nu_n}\left(\mu_n,\frac{\kappa_n+1}{\kappa_n}\sigma_n^2\right).
\end{aligned}$$

NB10 Gaussian Analysis

In the above **expressions**

$$\begin{aligned} \nu_n &= \nu_0 + n, \\ \sigma_n^2 &= \frac{1}{\nu_n}\left[\nu_0 \sigma_0^2 + (n-1)s^2 + \frac{\kappa_0 n}{\kappa_0 + n}(\bar{y} - \mu_0)^2\right], \qquad (23) \\ \mu_n &= \frac{\kappa_0}{\kappa_0 + n}\,\mu_0 + \frac{n}{\kappa_0 + n}\,\bar{y}, \quad \text{and} \\ \kappa_n &= \kappa_0 + n, \end{aligned}$$

$\bar{y}$ and s^2 are the usual **sample mean** and **variance** of y, and $\mathcal{G}$ denotes the assumption of the **Gaussian model**.

Here $t_\nu(\mu, \sigma^2)$ is a **scaled** version of the usual t_ν distribution, i.e., $W \sim t_\nu(\mu, \sigma^2) \iff \frac{W-\mu}{\sigma} \sim t_\nu$.

The scaled t distribution (see, e.g., Gelman et al., 2003, Appendix A) has **density**

$$\eta \sim t_\nu(\mu, \sigma^2) \leftrightarrow p(\eta) = \frac{\Gamma\left[\frac{1}{2}(\nu+1)\right]}{\Gamma\left(\frac{1}{2}\nu\right)\sqrt{\nu \pi \sigma^2}}\left[1 + \frac{1}{\nu \sigma^2}(\eta - \mu)^2\right]^{-\frac{1}{2}(\nu+1)}. \qquad (24)$$

This distribution has **mean** μ (as long as $\nu > 1$) and **variance** $\frac{\nu}{\nu-2}\sigma^2$ (as long as $\nu > 2$).

Notice that, as with all previous conjugate examples, the posterior mean is again a **weighted average** of the prior mean and data mean, with weights determined by the **prior sample size** and the **data sample size**:

$$\mu_n = \frac{\kappa_0}{\kappa_0 + n}\,\mu_0 + \frac{n}{\kappa_0 + n}\,\bar{y}. \qquad (25)$$

NB10 Gaussian Analysis (continued)

NB10 Gaussian Analysis. *Question (a):* I don't know anything about what NB10 is supposed to weigh (down to the nearest microgram) or about the accuracy of the NBS's measurement process, so I want to use a **diffuse prior** for μ and σ^2.

Considering the meaning of the **hyperparameters**, to provide little prior information I want to choose both ν_0 and κ_0 **close to 0**.

Making them exactly 0 would produce an **improper** prior distribution (which doesn't integrate to 1), but choosing positive values as close to 0 as you like yields a **proper and highly diffuse prior**.

You can see from (100, 101) that the result is then

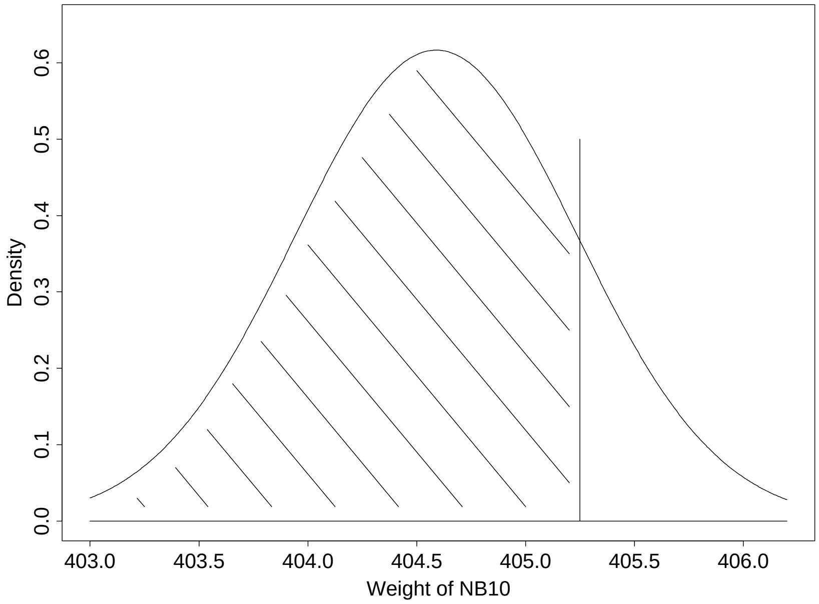
$$(\mu | y, \mathcal{G}) \sim t_n \left[\bar{y}, \frac{(n-1)s^2}{n^2} \right] \doteq N\left(\bar{y}, \frac{s^2}{n} \right), \qquad (26)$$

i.e., with diffuse prior information (as with the Bernoulli model in the AMI case study) the 95% central Bayesian interval **virtually coincides** with the usual frequentist 95% confidence interval
$\bar{y} \pm t_{n-1}^{.975} \frac{s}{\sqrt{n}} = 404.6 \pm (1.98)(0.647) = (403.3, 405.9)$.

Thus both {frequentists who assume $\mathcal{G}$} and {Bayesians who assume $\mathcal{G}$ with a diffuse prior} conclude that **NB10 weighs about** 404.6μg **below 10g, give or take about** 0.65μg.

Question (b). If interest focuses on whether NB10 weighs **less than some value** like 405.25, when reasoning in a Bayesian way you can answer this question directly: the posterior distribution for μ is shown below, and $P_B(\mu < 405.25 | y, \mathcal{G}, \text{diffuse prior}) \doteq .85$, i.e., your **betting odds** in favor of the proposition that $\mu < 405.25$ are about 5.5 to 1.

NB10 Gaussian Analysis (continued)



When reasoning in a frequentist way $P_F(\mu < 405.25)$ is **undefined**; about the best you can do is to test $H_0 \colon \mu < 405.25$, for which the p-value would (approximately) be $p = P_{F,\mu=405.25}(\bar{y} > 405.59) = 1 - .85 = .15$, i.e., **insufficient evidence to reject** H_0 at the usual significance levels (note the **connection** between the p-value and the posterior probability, which arises in this example because the null hypothesis is **one-sided**).

NB The significance test tries to answer a **different question**: in Bayesian language it looks at $P(\bar{y}|\mu)$ instead of $P(\mu|\bar{y})$.

Many people find the latter quantity **more interpretable**.

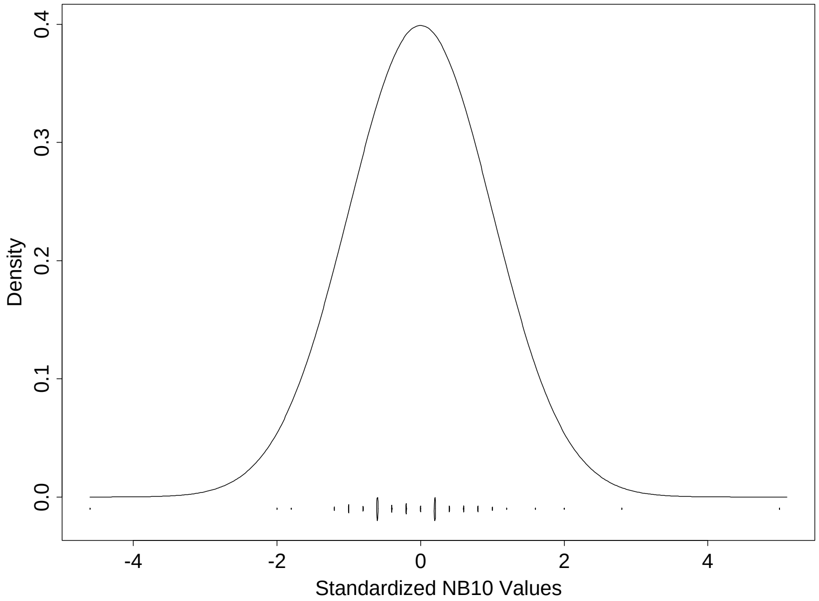
Question (c). We saw earlier that **in this model**

$$(y_{n+1}|y, \mathcal{G}) \sim t_{\nu_n}\left[\mu_n, \frac{\kappa_n + 1}{\kappa_n}\sigma_n^2\right], \qquad (27)$$

and for n large and ν_0 and κ_0 close to 0 this is $(y_{n+1}|y, \mathcal{G}) \dot{\sim} N(\bar{y}, s^2)$, i.e., a **95% posterior predictive interval** for y_{n+1} is $(392, 418)$.

Model Expansion

A **standardized version** of this predictive distribution is plotted below, with the standardized NB10 data values **superimposed**.



It's evident from this plot (and also from the normal qqplot given earlier) that the Gaussian model provides a **poor fit** for these data—the three most extreme points in the data set in standard units are $-4.6, 2.8$, and 5.0.

With the **symmetric heavy tails** indicated in these plots, in fact, the empirical CDF looks quite a bit like that of a t distribution with a rather small number of **degrees of freedom**.

This suggests revising the previous model by **expanding** it: **embedding** the Gaussian in the t family and adding a parameter k for **tail-weight**.

Unfortunately there's no standard **closed-form conjugate** choice for the prior on k.

A more **flexible** approach to computing is evidently needed—this is where **Markov chain Monte Carlo** methods come in.

t Sampling Distribution

$\boxed{\textbf{Example:}}$ the **NB10 Data**. Recall from the posterior predictive plot toward the end of part 2 of the lecture notes that the Gaussian model for the NB10 data was inadequate: the tails of the data distribution are **too heavy** for the Gaussian.

It was also clear from the normal qqplot that the data are **symmetric**.

This suggests thinking of the NB10 data values y_i as like draws from a t **distribution** with fairly small degrees of freedom ν.

One way to write this model is

$$\begin{aligned} (\mu, \sigma^2, \nu) & \sim p(\mu, \sigma^2, \nu) \\ (y_i | \mu, \sigma^2, \nu) & \stackrel{\text{IID}}{\sim} t_\nu(\mu, \sigma^2), \end{aligned} \tag{28}$$

where $t_\nu(\mu, \sigma^2)$ denotes the **scaled t-distribution** with mean μ, scale parameter σ^2, and shape parameter ν.

This distribution has variance $\sigma^2 \left(\frac{\nu}{\nu - 2} \right)$ for $\nu > 2$ (so that shape and scale are mixed up, or **confounded** in $t_\nu(\mu, \sigma^2)$) and may be thought of as the distribution of the quantity $\mu + \sigma \, e$, where e is a draw from the **standard** t distribution that is tabled at the back of all introductory statistics books.

However, a **better way** to think about model (28) is as follows.

It's a fact from **basic distribution theory**, probably of more interest to Bayesians than frequentists, that the t distribution is an $\boxed{\textbf{Inverse Gamma mixture of Gaussians}}$.

This just means that to generate a t random quantity you can first draw from an Inverse Gamma distribution and then draw from a Gaussian **conditional** on what you got from the Inverse Gamma.

t Sampling Distribution

($\lambda \sim \Gamma^{-1}(\alpha, \beta)$ just means that $\lambda^{-1} = \frac{1}{\lambda} \sim \Gamma(\alpha, \beta)$).

In more detail, $(y|\mu, \sigma^2, \nu) \sim t_\nu(\mu, \sigma^2)$ is the same as the **hierarchical model**

$$\begin{array}{rcl} (\lambda|\nu) & \sim & \Gamma^{-1}\left(\frac{\nu}{2}, \frac{\nu}{2}\right) \\ (y|\mu, \sigma^2, \lambda) & \sim & N\left(\mu, \lambda\, \sigma^2\right). \end{array} \qquad (29)$$

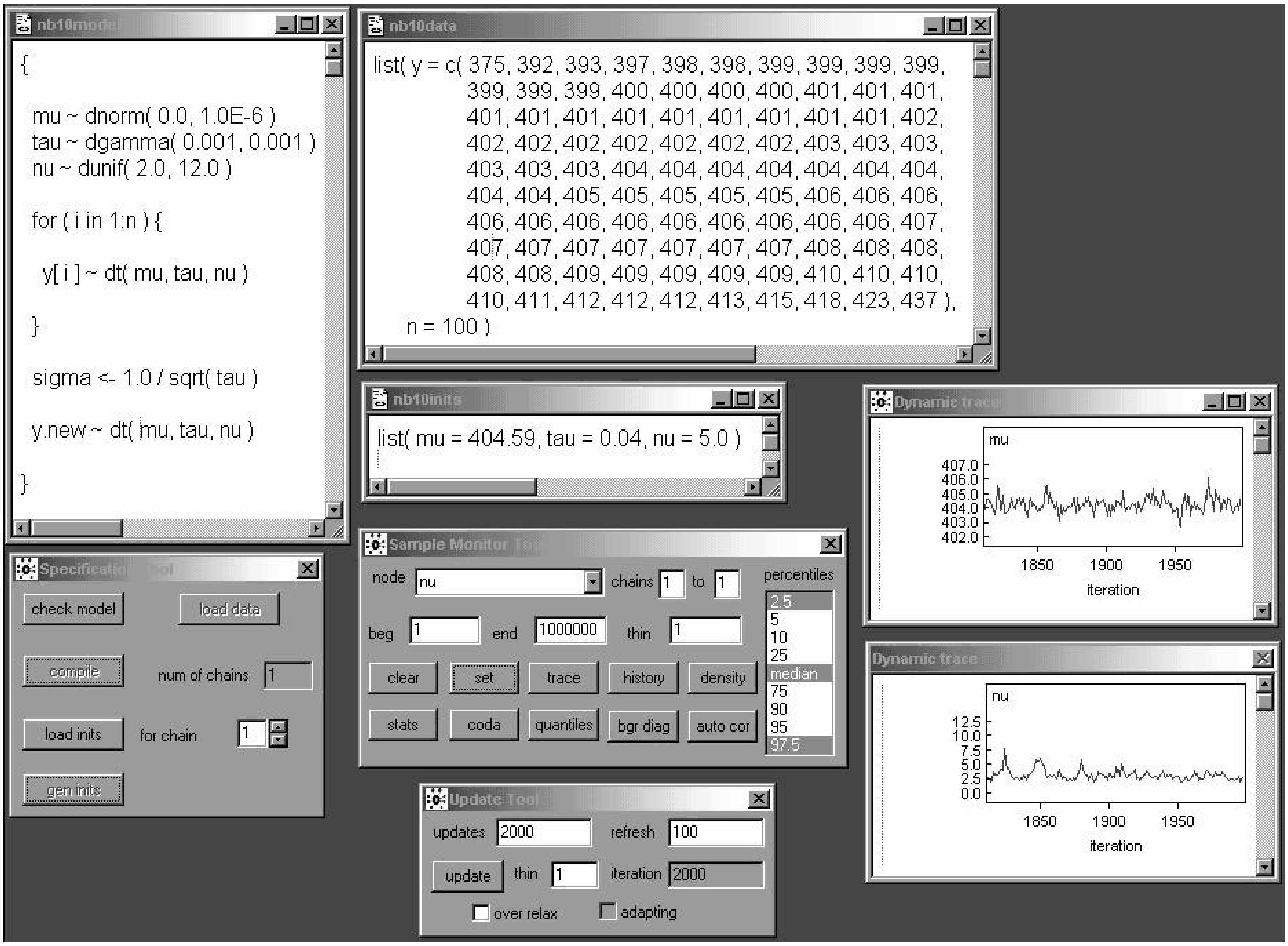
Putting this together with the **conjugate prior** for μ and σ^2 we looked at earlier in the Gaussian model gives the following HM for the NB10 data:

$$\begin{array}{rcl} \nu & \sim & p(\nu) \\ \sigma^2 & \sim & \text{SI-}\chi^2\left(\nu_0, \sigma_0^2\right) \\ \left(\mu|\sigma^2\right) & \sim & N\left(\mu_0, \frac{\sigma^2}{\kappa_0}\right) \\ (\lambda_i|\nu) & \overset{\text{IID}}{\sim} & \Gamma^{-1}\left(\frac{\nu}{2}, \frac{\nu}{2}\right) \\ \left(y_i|\mu, \sigma^2, \lambda_i\right) & \overset{\text{indep}}{\sim} & N\left(\mu, \lambda_i\, \sigma^2\right). \end{array} \qquad (30)$$

Remembering also from introductory statistics that the Gaussian distribution is the **limit** of the t family as $\nu \rightarrow \infty$, you can see that the idea here has been to **expand** the Gaussian model by embedding it in the richer t family, of which it's a special case with $\nu = \infty$.

Model expansion is often the best way to deal with **uncertainty in the modeling process**: when you find deficiencies of the current model, **embed it in a richer class**, with the model expansion in directions suggested by the deficiencies (we'll also see this method in action again later).

WinBUGS Implementation



nb10model

```
{

  mu ~ dnorm( 0.0, 1.0E-6 )
  tau ~ dgamma( 0.001, 0.001 )
  nu ~ dunif( 2.0, 12.0 )

  for ( i in 1:n ) {

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

  }

  sigma <- 1.0 / sqrt( tau )

  y.new ~ dt( mu, tau, nu )

}
```

nb10data

```
list( y = c( 375, 392, 393, 397, 398, 398, 399, 399, 399, 399,
             399, 399, 399, 400, 400, 400, 400, 401, 401, 401,
             401, 401, 401, 401, 401, 401, 401, 401, 401, 402,
             402, 402, 402, 402, 402, 402, 402, 403, 403, 403,
             403, 403, 403, 404, 404, 404, 404, 404, 404, 404,
             404, 404, 405, 405, 405, 405, 405, 406, 406, 406,
             406, 406, 406, 406, 406, 406, 406, 406, 406, 407,
             407, 407, 407, 407, 407, 407, 407, 408, 408, 408,
             408, 408, 409, 409, 409, 409, 409, 410, 410, 410,
             410, 411, 412, 412, 412, 413, 415, 418, 423, 437 ),
      n = 100 )
```

nb10inits

```
list( mu = 404.59, tau = 0.04, nu = 5.0 )
```

Specification Tool: num of chains 1; for chain 1

Sample Monitor Tool: node nu; chains 1 to 1; beg 1; end 1000000; thin 1

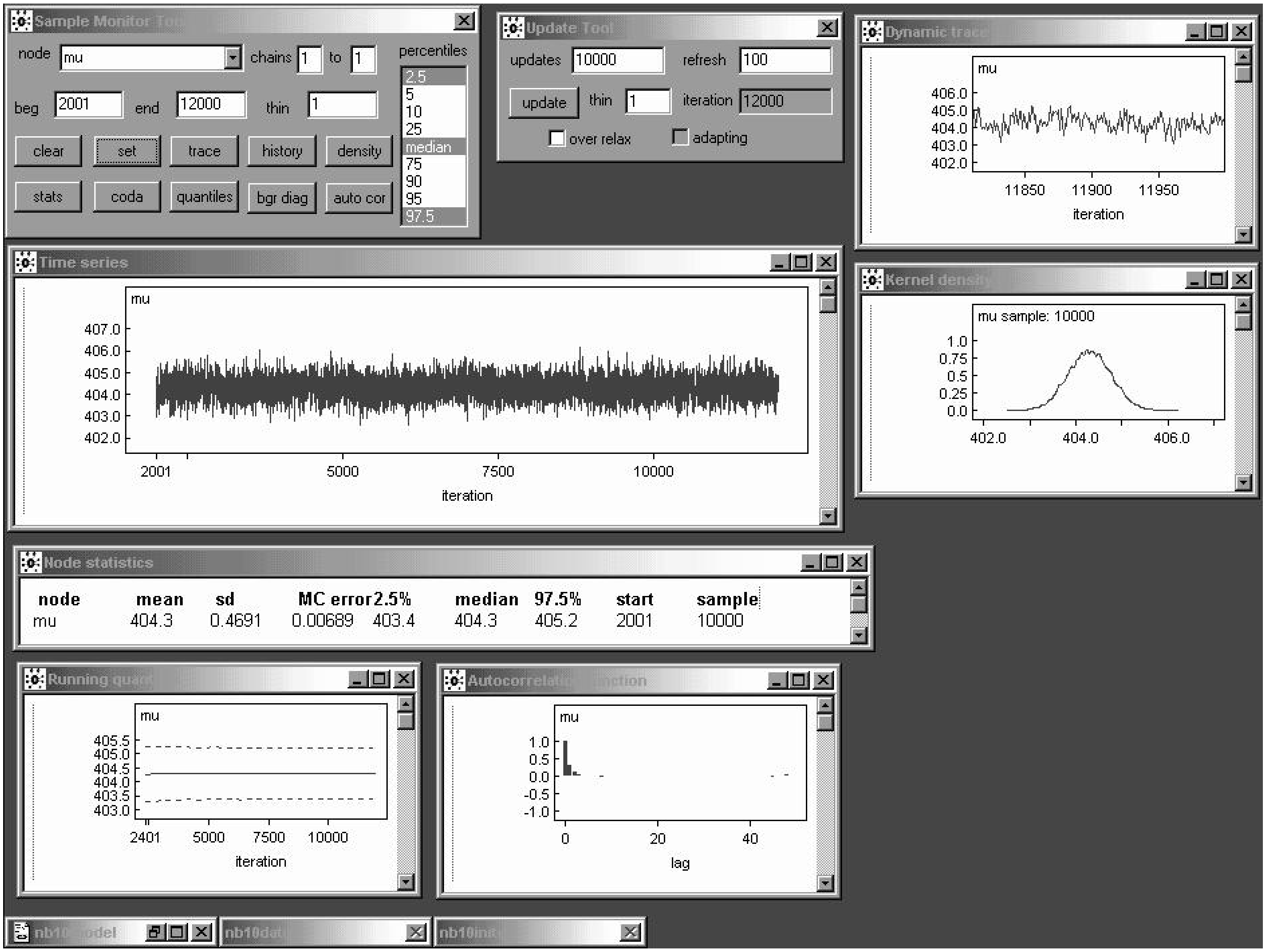
Update Tool: updates 2000; refresh 100; thin 1; iteration 2000; □ over relax; □ adapting

I read in three files—the **model**, the **data**, and the **initial values**—and used the `Specification Tool` from the `Model` menu to `check` the model, `load` the data, `compile` the model, `load` the initial values, and `generate` additional initial values for uninitialized nodes in the graph.

I then used the `Sample Monitor Tool` from the `Inference` menu to `set` the `mu`, `sigma`, `nu`, and `y.new` nodes, and clicked on `Dynamic Trace` **plots** for `mu` and `nu`.

Then choosing the `Update Tool` from the `Model` menu, specifying 2000 in the `updates` box, and clicking `update` permitted a **burn-in** of 2,000 iterations to occur with the **time series traces** of the two parameters displayed in **real time**.

WinBUGS Implementation (continued)



Sample Monitor Tool

node mu chains 1 to 1

beg 2001 end 12000 thin 1

Update Tool

updates 10000 refresh 100

thin 1 iteration 12000

□ over relax □ adapting

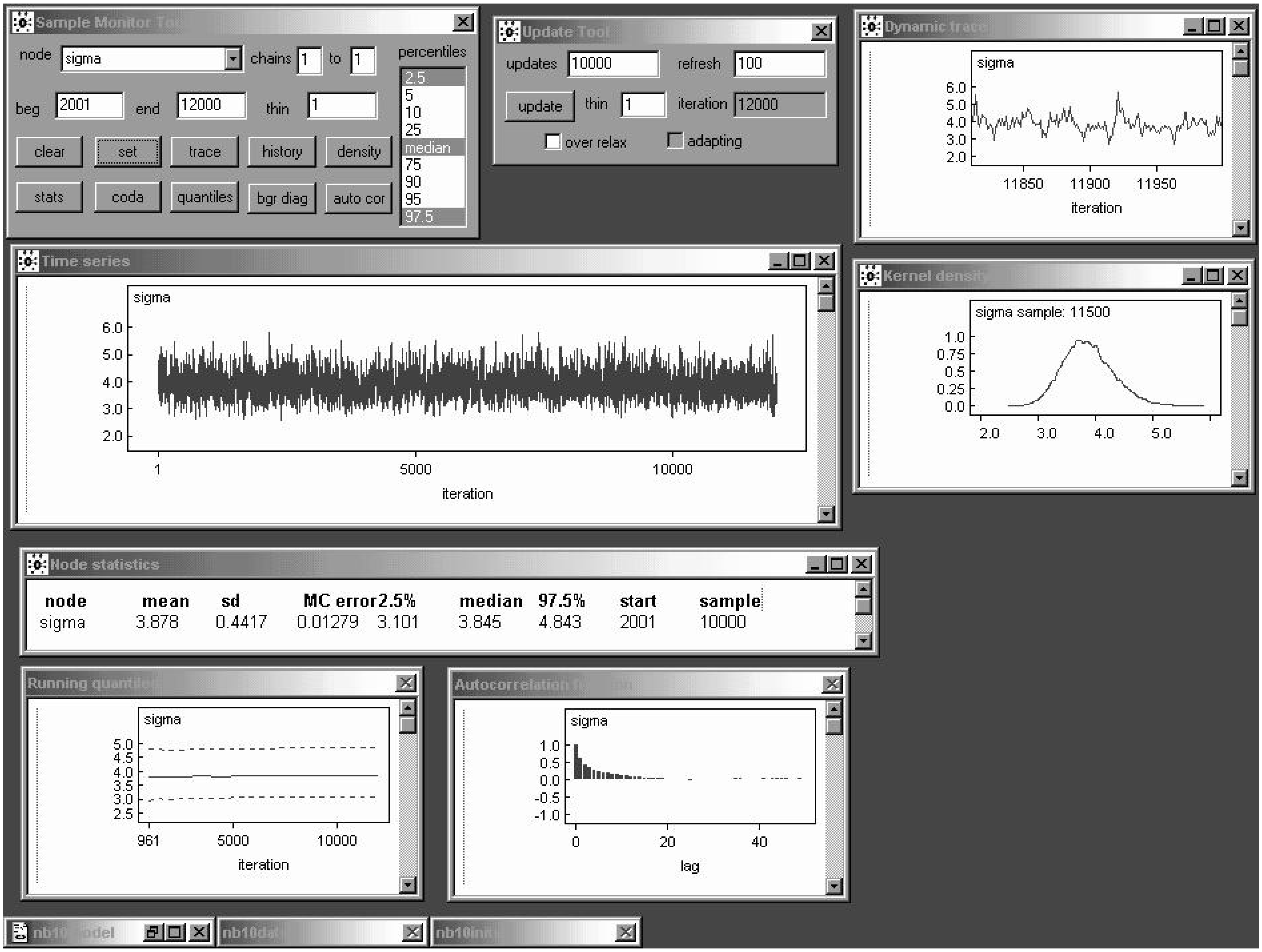
Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
mu	404.3	0.4691	0.00689	403.4	404.3	405.2	2001	10000

After **minimizing** the model, data, and inits windows and **killing** the Specification Tool (which are no longer needed until the model is respecified), I typed 10000 in the updates box of the Update Tool and clicked update to generate a **monitoring run** of 10,000 iterations (you can watch the updating of mu and nu dynamically to get an idea of the **mixing**, but this slows down the sampling).

After **killing** the Dynamic Trace window for nu (to concentrate on mu for now), in the Sample Monitor Tool I selected mu from the pull-down menu, set the beg and end boxes to 2001 and 12000, respectively (to summarize only the **monitoring** part of the run), and clicked on history to get the **time series trace** of the monitoring run, density to get a **kernel density trace** of the 10,000 iterations, stats to get **numerical summaries** of the monitored iterations, quantiles to get a trace of the **cumulative estimates** of the 2.5%, 50% and 97.5% points in the estimated posterior, and autoC to get the **autocorrelation function**.

WinBUGS Implementation (continued)



Sample Monitor Tool

node sigma chains 1 to 1

beg 2001 end 12000 thin 1

Update Tool

updates 10000 refresh 100

thin 1 iteration 12000

□ over relax □ adapting

Node statistics

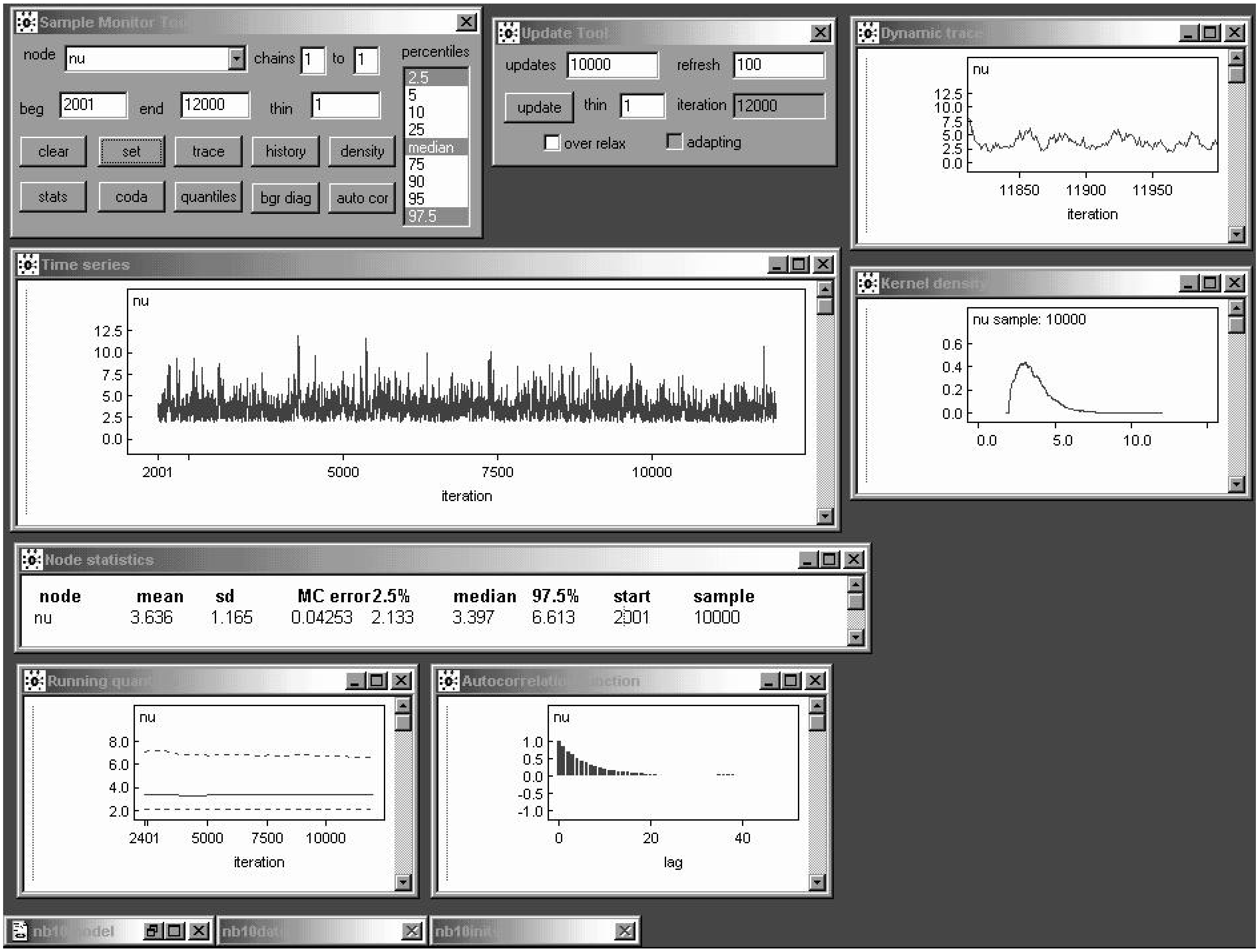
node	mean	sd	MC error	2.5%	median	97.5%	start	sample
sigma	3.878	0.4417	0.01279	3.101	3.845	4.843	2001	10000

You can see that the output for μ is **mixing fairly well**—the ACF looks like that of an AR_1 series with first-order **serial correlation** of only about **0.3**.

σ is mixing less well: its ACF looks like that of an AR_1 series with first-order **serial correlation** of about **0.6**.

This means that a monitoring run of 10,000 would probably **not be enough** to satisfy **minimal Monte Carlo accuracy goals**—for example, from the Node statistics window the estimated posterior mean is **3.878** with an estimated MC error of **0.0128**, meaning that we've not yet achieved **three-significant-figure accuracy** in this posterior summary.

`WinBUGS` Implementation (continued)



node: nu; chains 1 to 1; beg 2001; end 12000; thin 1

updates 10000; refresh 100; thin 1; iteration 12000

□ over relax □ adapting

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
nu	3.636	1.165	0.04253	2.133	3.397	6.613	2001	10000

And ν's mixing is the worst of the three: its ACF looks like that of an AR_1 series with first-order **serial correlation** of a bit less than **+0.9**.

`WinBUGS` has a somewhat complicated provision for printing out the autocorrelations; alternately, you can **approximately infer** $\hat{\rho}_1$ from an equation like (51) above: assuming that the `WinBUGS` people are taking the output of any MCMC chain as (**at least approximately**) AR_1 and using the formula

$$\widehat{SE}\left(\bar{\theta}^*\right) = \frac{\hat{\sigma}_\theta}{\sqrt{m}} \sqrt{\frac{1+\hat{\rho}_1}{1-\hat{\rho}_1}}, \tag{31}$$

you can **solve** this equation for $\hat{\rho}_1$ to get

$$\hat{\rho}_1 = \frac{m\left[\widehat{SE}\left(\bar{\theta}^*\right)\right]^2 - \hat{\sigma}_\theta^2}{m\left[\widehat{SE}\left(\bar{\theta}^*\right)\right]^2 + \hat{\sigma}_\theta^2}. \tag{32}$$

`WinBUGS` Implementation (continued)

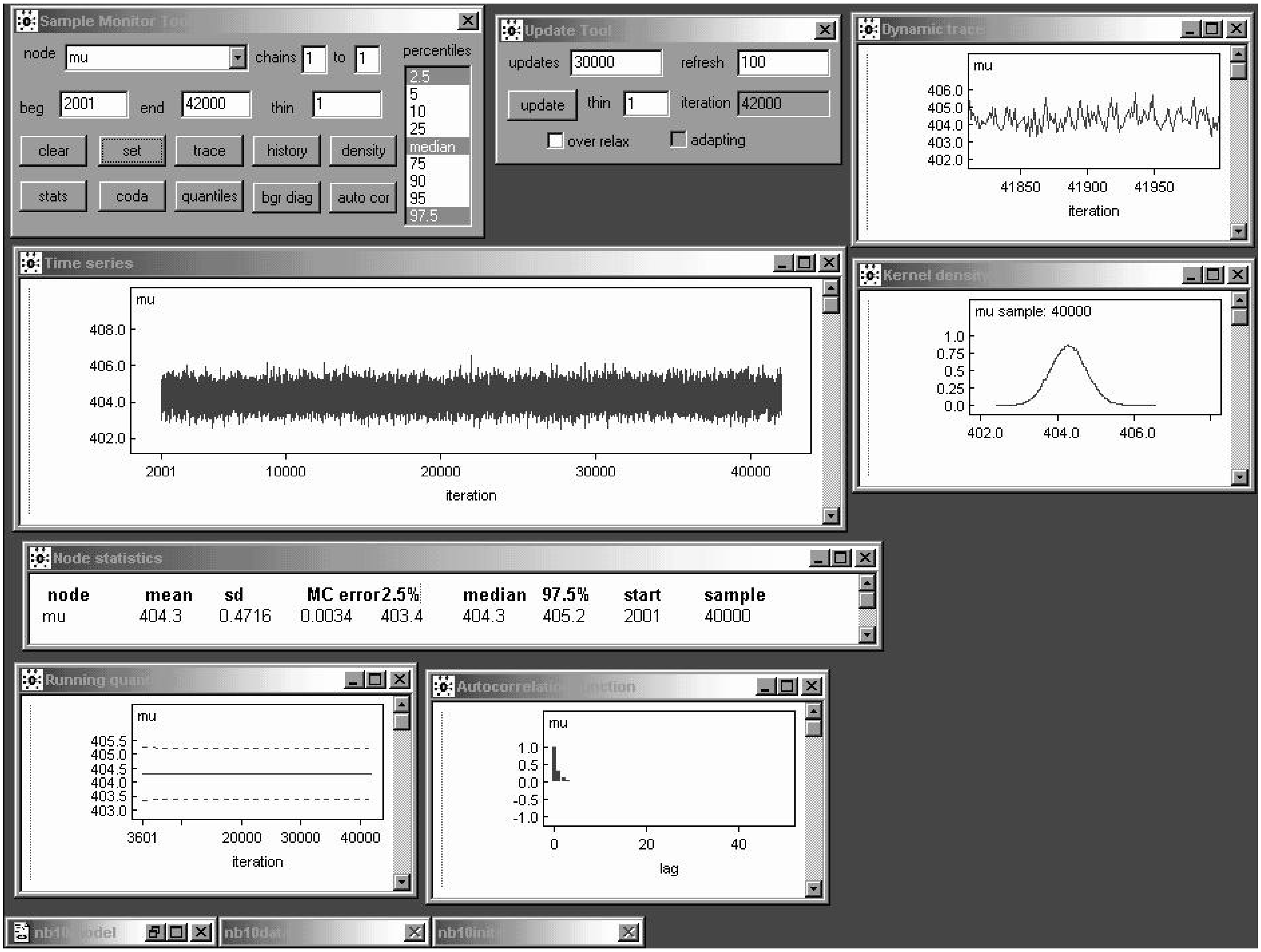
Plugging in the relevant values here gives

$$\widehat{\rho}_1 = \frac{(10,000)(0.04253)^2 - (1.165)^2}{(10,000)(0.04253)^2 + (1.165)^2} \doteq 0.860, \qquad (33)$$

which is smaller than the corresponding value of **0.972** generated by the `classicBUGS` sampling method (from `CODA`, page 67).

To match the `classicBUGS` strategy outlined above (page 71) I typed 30000 in the `updates` window in the `Update Tool` and hit `update`, yielding a **total monitoring run** of 40,000.

Remembering to type **42000** in the `end` box in the `Sample Monitoring Tool` window before going any further, to get a **monitoring** run of 40,000 after the initial **burn-in** of 2,000, the summaries below for μ are **satisfactory in every way**.



Sample Monitor Tool

node mu chains 1 to 1

beg 2001 end 42000 thin 1

percentiles: 2.5, 5, 10, 25, median, 75, 90, 95, 97.5

Update Tool

updates 30000 refresh 100

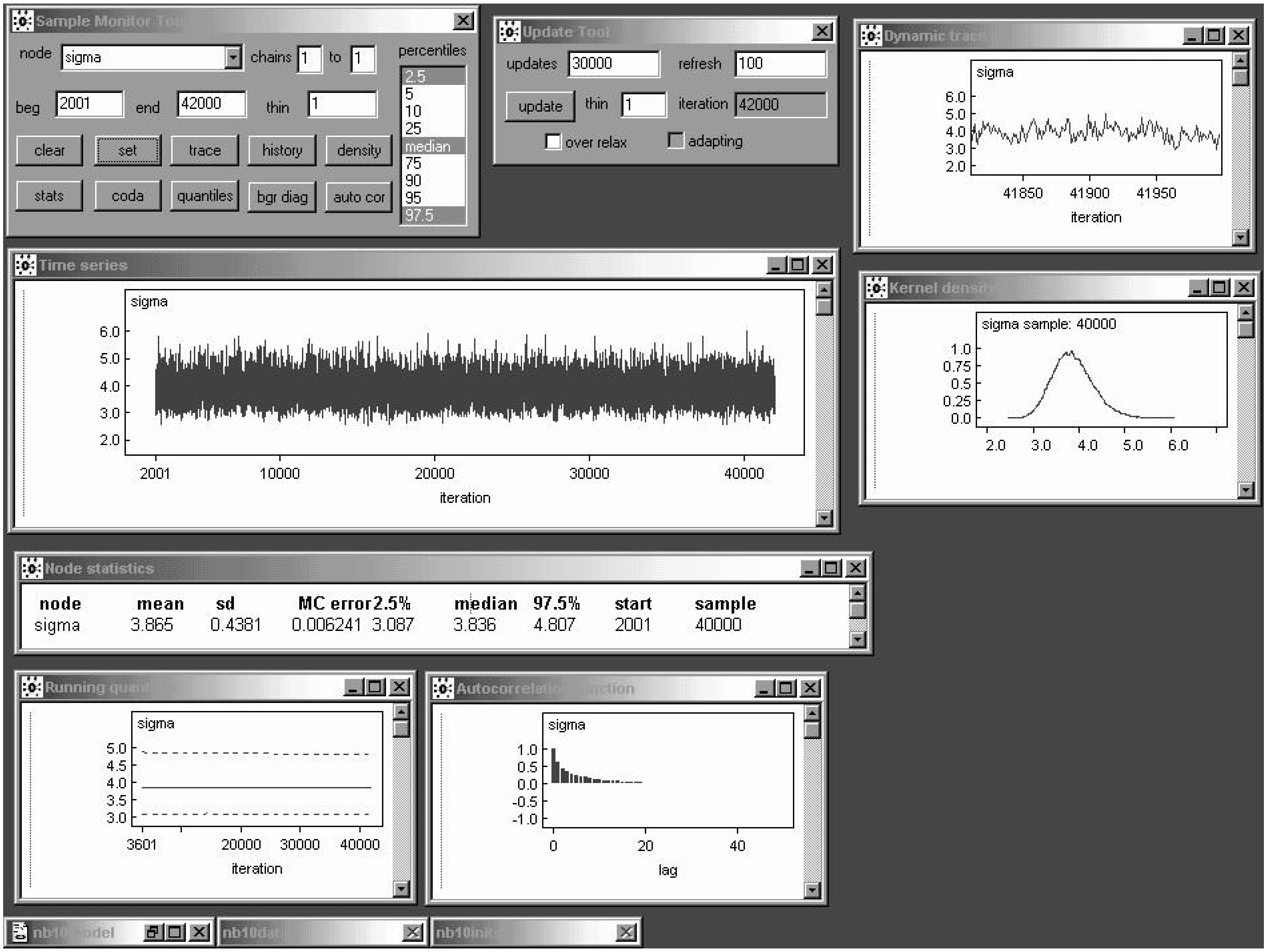
thin 1 iteration 42000

□ over relax □ adapting

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
mu	404.3	0.4716	0.0034	403.4	404.3	405.2	2001	40000

WinBUGS Implementation (continued)



node sigma, chains 1 to 1

beg 2001, end 42000, thin 1

updates 30000, refresh 100

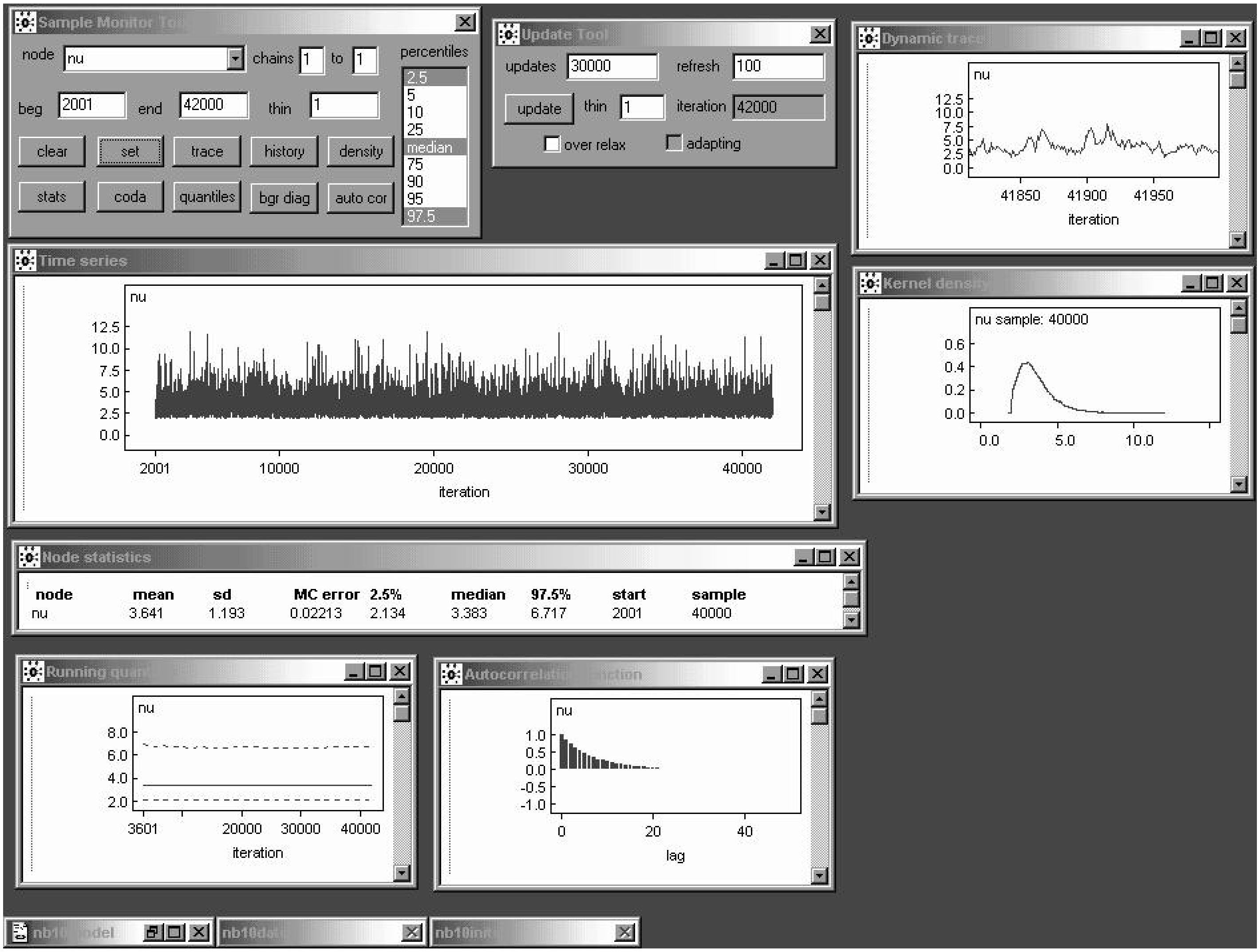
thin 1, iteration 42000

□ over relax □ adapting

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
sigma	3.865	0.4381	0.006241	3.087	3.836	4.807	2001	40000

A monitoring run of **40,000** also looks good for σ: on this basis, and **conditional on this model and prior**, I think σ is around **3.87** (posterior mean, with an **MCSE** of **0.006**), give or take about **0.44** (posterior SD), and my 95% central posterior interval for σ runs from about **3.09** to about **4.81** (the distribution has a bit of **skewness** to the right, which makes sense given that σ is a **scale parameter**).

WinBUGS Implementation (continued)



Sample Monitor Tool

node: nu; chains: 1 to 1; beg: 2001; end: 42000; thin: 1

Update Tool

updates: 30000; refresh: 100; thin: 1; iteration: 42000

□ over relax □ adapting

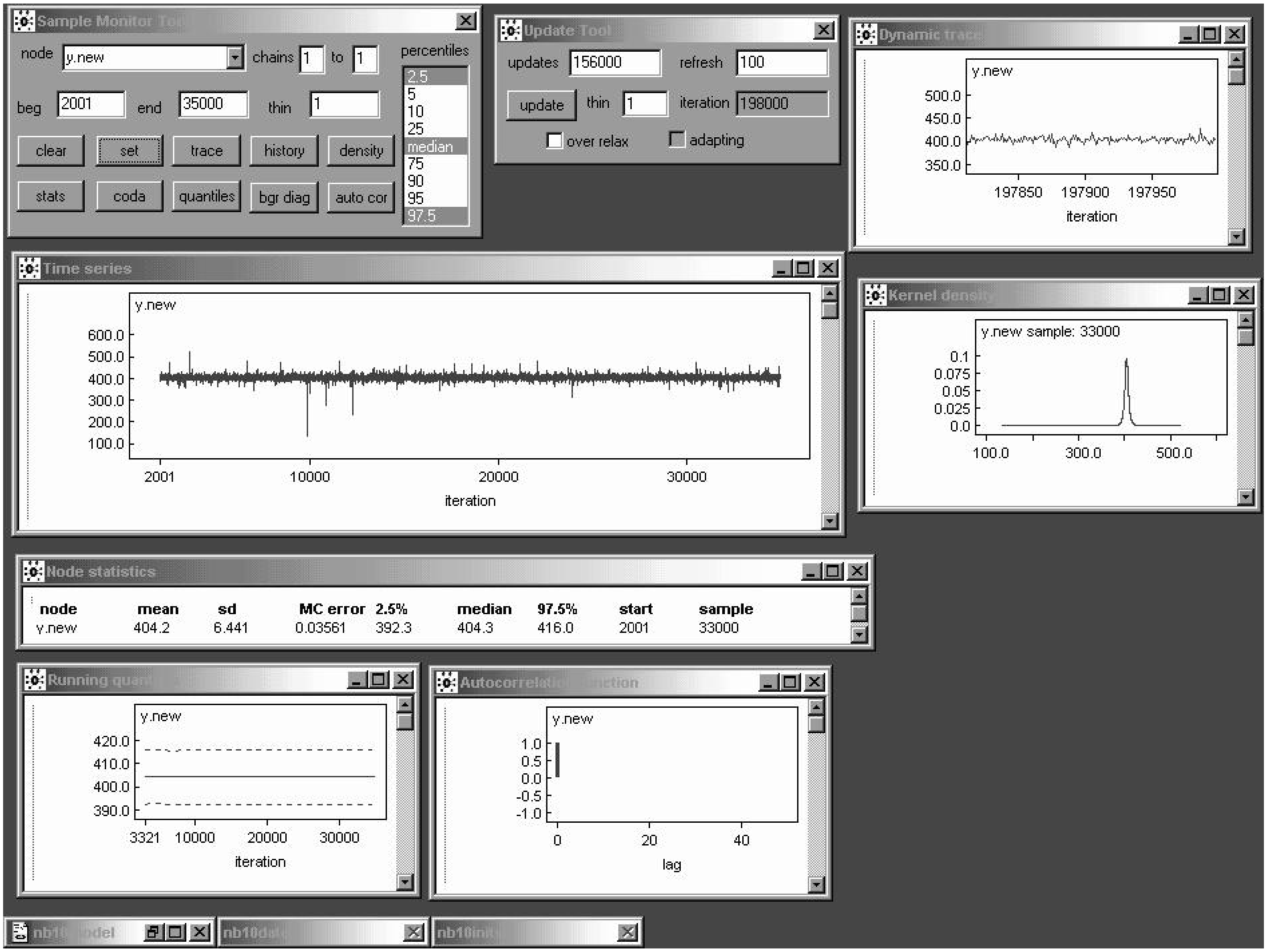
Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
nu	3.641	1.193	0.02213	2.134	3.383	6.717	2001	40000

If the **real goal** were ν I would use a **longer monitoring run**, but the main point here is μ, and we saw back on p. 67 that μ and ν are **close to uncorrelated in the posterior**, so this is good enough.

If you wanted to report the **posterior mean** of ν with an MCSE of **0.01** (to come close to 3-sigfig accuracy) you'd have to increase the length of the monitoring run by a **multiplicative factor** of $\left(\frac{0.02213}{0.01}\right)^2 \doteq 4.9$, which would yield a **recommended length** of monitoring run of about **196,000** iterations (the entire monitoring phase would take about **3 minutes** at **2.0 (PC) GHz**).

WinBUGS Implementation (continued)



node y.new chains 1 to 1

beg 2001 end 35000 thin 1

updates 156000 refresh 100

thin 1 iteration 198000

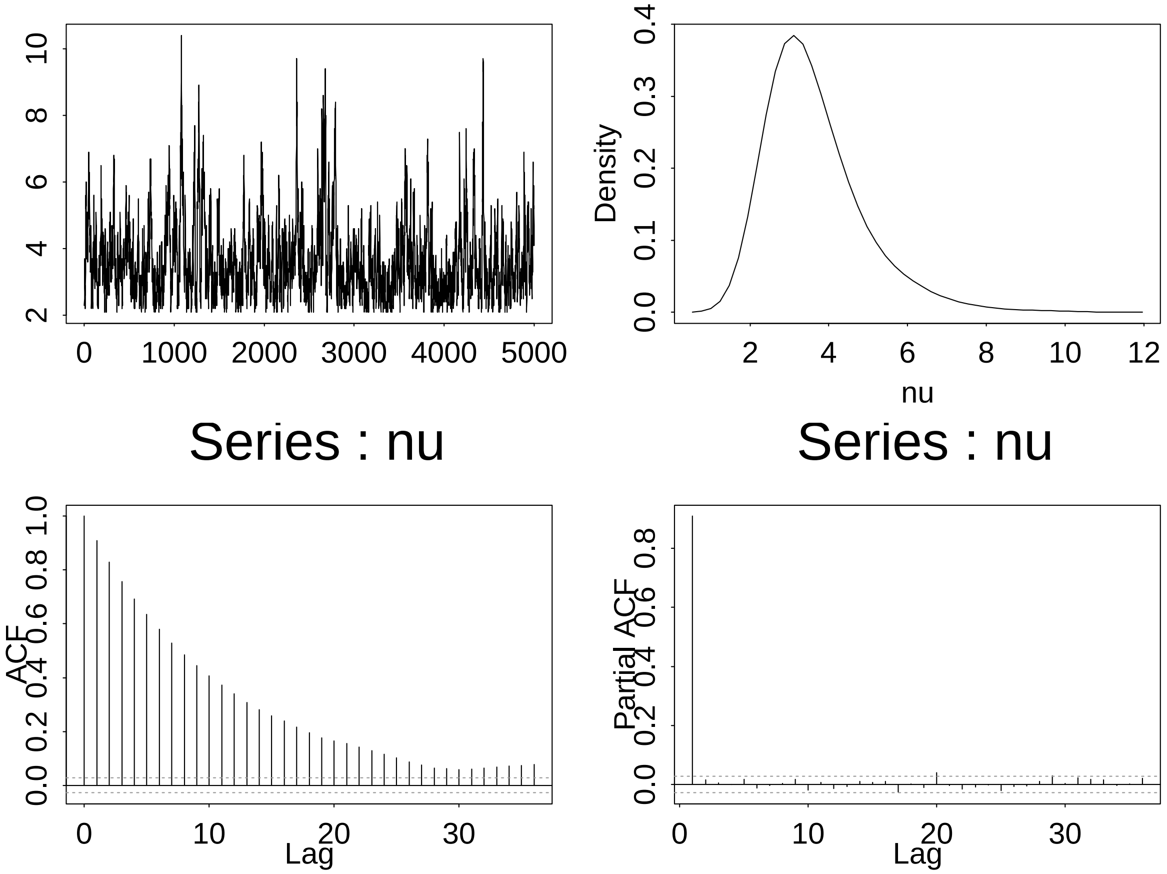
☐ over relax ☐ adapting

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
y.new	404.2	6.441	0.03561	392.3	404.3	416.0	2001	33000

The **posterior predictive distribution** for y_{n+1} given $(y_1, \ldots, y_n)$ is interesting in the t model: the predictive mean and SD of 404.3 and 6.44 are **not far** from the sample mean and SD (404.6 and 6.5, respectively), but the predictive distribution has **very heavy tails**, consistent with the degrees of freedom parameter ν in the t distribution being so small (the time series trace has a few simulated values less than **300** and greater than **500**, **much farther** from the center of the observed data than the most outlying actual observations).

Gaussian Comparison

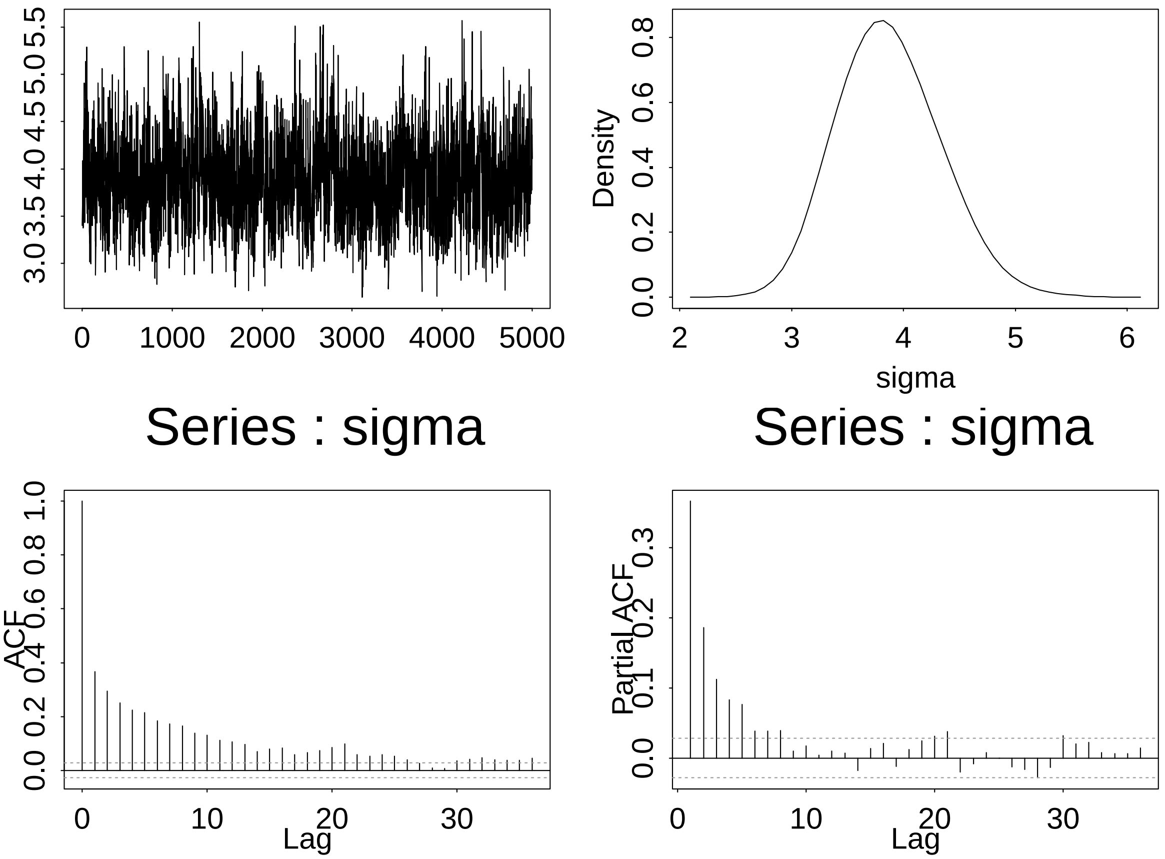
The posterior SD for μ, the only parameter directly comparable across the Gaussian and t models for the NB10 data, came out $\boxed{\textbf{0.47}}$ from the t modeling, versus $\boxed{0.65}$ with the Gaussian, i.e., the interval estimate for μ from the (incorrect) Gaussian model is about **40% wider** that that from the (much better-fitting) t model.



A Model Uncertainty Anomaly?

$\boxed{\textbf{NB}}$ Moving from the Gaussian to the t model involves a net increase in **model uncertainty**, because when you assume the Gaussian you're in effect saying that you know the t degrees of freedom are ∞, whereas with the t model you're treating ν as unknown. And yet, even though there's been an increase in model uncertainty, the inferential uncertainty about μ has **gone down**.

This is relatively rare—**usually when model uncertainty increases so does inferential uncertainty** (Draper 2004)—and arises in this case because of two things: (a) the t model **fits better** than the Gaussian, and (b) the Gaussian is actually a **conservative** model to assume as far as inferential accuracy for location parameters is concerned.


Density
sigma
Series : sigma
ACF
Lag
Series : sigma
Partial ACF
Lag