Assgn2: Gene expression data analysis 2

Salvador Garcia, s1655274

3 February 2016

1 Description of the problem

Expression (concentration) of each of 50 genes was observed in the pancreas tissue of two groups of people: in a case group of 12 people who have pancreatic cancer, and in a control group of 10 people who do not have pancreatic cancer (the groups are approximately matched by age and gender). The study was performed in the same hospital.

The question is to determine whether there is a difference between the gene expression in the pancreas tissue between the groups of people with and without the cancer.

Statistical analysis.

Introduce the following random variables. Denote the logarithm of the gene expression of the kth gene for individual i in the case group by X_{ik}, and for the jth individual in the control group by Y_{jk}, $k = 1, 2, \ldots, N = 50$, $i = 1, \ldots, n = 12$, $j = 1, 2, \ldots, m = 10$.

2 Likelihood

The data have the following distribution:

$$X_{ik} \mid \mu_1, \sigma_1 \sim N(\mu_1, \sigma_1^2),\ i = 1, 2, \ldots, n \quad \textit{independently (given } \mu_1, \sigma_1^2),$$
$$Y_{jk} \mid \mu_2, \sigma_2 \sim N(\mu_2, \sigma_2^2),\ j = 1, 2, \ldots, m \quad \textit{independently (given } \mu_2, \sigma_2^2).$$

Also, X_{ik} and Y_{jk} are independent for all i, j, k. The observed data can be summarized as follows: $\bar{x} = 4.03$, $\bar{y} = 2.59$, $s_X = 0.29$ and $s_Y = 0.11$ where:

$$\bar{X} = \frac{1}{nN}\sum_{i=1}^{n}\sum_{k=1}^{N} X_{ik}, \quad \bar{Y} = \frac{1}{mN}\sum_{j=1}^{m}\sum_{k=1}^{N} Y_{jk},$$
$$s_X^2 = \frac{1}{nN}\sum_{i=1}^{n}\sum_{k=1}^{N} (X_{ik} - \bar{X})^2, \quad s_Y^2 = \frac{1}{mN}\sum_{j=1}^{m}\sum_{k=1}^{N} (Y_{jk} - \bar{Y})^2.$$

The purpose of this assignment is to determine if there is a difference between μ_1 and μ_2. In terms of the problem, to find if there is a relevant difference of gene expression in the two different tissues. In traditional statistics, this can be addressed by testing the hypothesis $\mu_1 = \mu_2$ or, equivalently, $\mu_1 - \mu_2 = 0$ given some α. In Bayesian statistics something similar can be done, but it is in terms of the posterior distribution of $\delta = \mu_1 - \mu_2$.

3 Prior distribution

3.1 Priors for μ_i

In this example we are working with normal distributions (that have two parameters) of two populations , μ_1 and σ_1 for the first one and μ_2 and σ_2 for the second one. In order to address the information of the μ_i parameters, two uninformative priors for μ_1 and μ_2 will be used: $\mu_i \sim N(0, A^2)$ with A very big. This way, the prior distribution of μ_1 and μ_2 will be as the fig. 1. It is easy to see that this distribution assigns almost evenly

the probabilities to all values from $(-\infty, \infty)$. Then, can be viewed as non-informative (The idea of apriori with a very big variance is important in this assignment).

Now, given the priors of μ_1 and μ_2, is necessary to think about the priors of σ_1 and σ_2. Is important to first give the priors μ_i because σ_i depends on its value.

3.2 Priors for σ_i

For σ_1 and σ_2 a gamma distribution with parameters *shape*(α) = *.001* and *rate* (β) = *.001* will be used. This distribution is shown in fig. 1. Remembering, the mean of the gamma distribution is $\frac{\alpha}{\beta}$, and the variance is $\frac{\alpha}{\beta^2}$. Then this distribution have $mean = 1$ and $variance = 1000$.

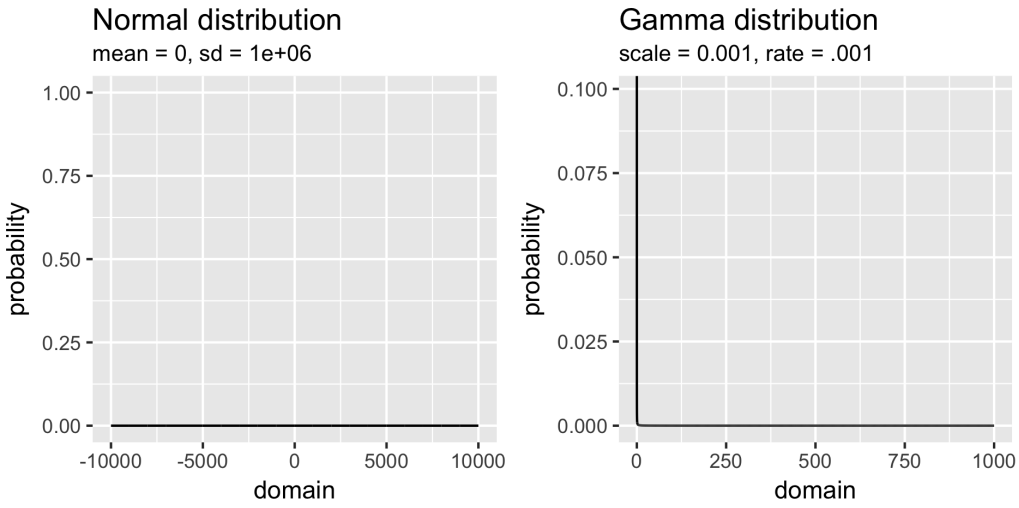

Figure 1: Uninformative normal and gamma distributions

Now, information from a previous study is used as a new apriori for σ_i. For the first one, the study says that the average precision (inverse of variance) is 11.11 for the cases and 100 for the control, but the variability is not known. This information given is presented below eq. (1).

$$\frac{\alpha_1}{\beta_1} = 11.11 \quad , \quad \frac{\alpha_2}{\beta_2} = 100 \tag{1}$$

Now, the variability of these values it is not known, so two assumptions to get two different scenarios will be made. The first one is that the prior variance is 10 and the other that the prior variance is 1000. Then, for both scenarios, this information can be interpreted as followseq. (2):

$$\frac{\alpha_i}{\beta_i^2} = 10 \quad , \quad \frac{\alpha_i}{\beta_i^2} = 1000 \tag{2}$$

With these formulas, a system of equations can be created to find α_1, α_2, β_1, and β_2. Using simple algebra, is easy to find that the parameters for the large variance scenario are eq. (3) and for the low variance scenario are eq. (4).

$$\alpha_1 = \frac{11.11^2}{1000} \quad , \quad \beta_1 = \frac{11.11}{1000} \quad , \quad \alpha_2 = \frac{100^2}{1000} \quad , \quad \beta_2 = \frac{100}{1000} \tag{3}$$

$$\alpha_1 = \frac{11.11^2}{10} \quad , \quad \beta_1 = \frac{11.11}{10} \quad , \quad \alpha_2 = \frac{100^2}{10} \quad , \quad \beta_2 = \frac{100}{10} \tag{4}$$

This two *apriori* distributions are used along with the non-informative distribution selected. As the hint says, it is important to distinguish the difference between precision and variance in the model. The one that is modelled though a gamma distribution is the precision. So the variability (variance) of 10 and 1000 shown in this section is the variability of the gamma distribution (the prior) of the precision.

4 Posterior inference

In this section the posterior analysis of (σ_1, σ_2) will be given. This is done with the relations $\sigma_1 = \frac{1}{\tau_1}$ and $\sigma_2 = \frac{1}{\tau_2}$. For this analysis, two chains will be used with different initial points (due to the convergence property, at the end both chains will end in the same value).

4.1 Posterior distributions

4.1.1 non informative prior

Table 1: Summary of analysis with non informative prior

parameters	mean	sd	2.5%	50%	97.5%
μ_1	4.03	0.0119	4.007	4.03	4.053
μ_2	2.59	0.005	2.58	2.59	2.6
δ	1.44	0.0129	1.415	1.44	1.465
σ_1	0.2907	0.0084	0.2749	0.2905	0.3081
σ_2	0.1105	0.0035	0.1038	0.1104	0.1175

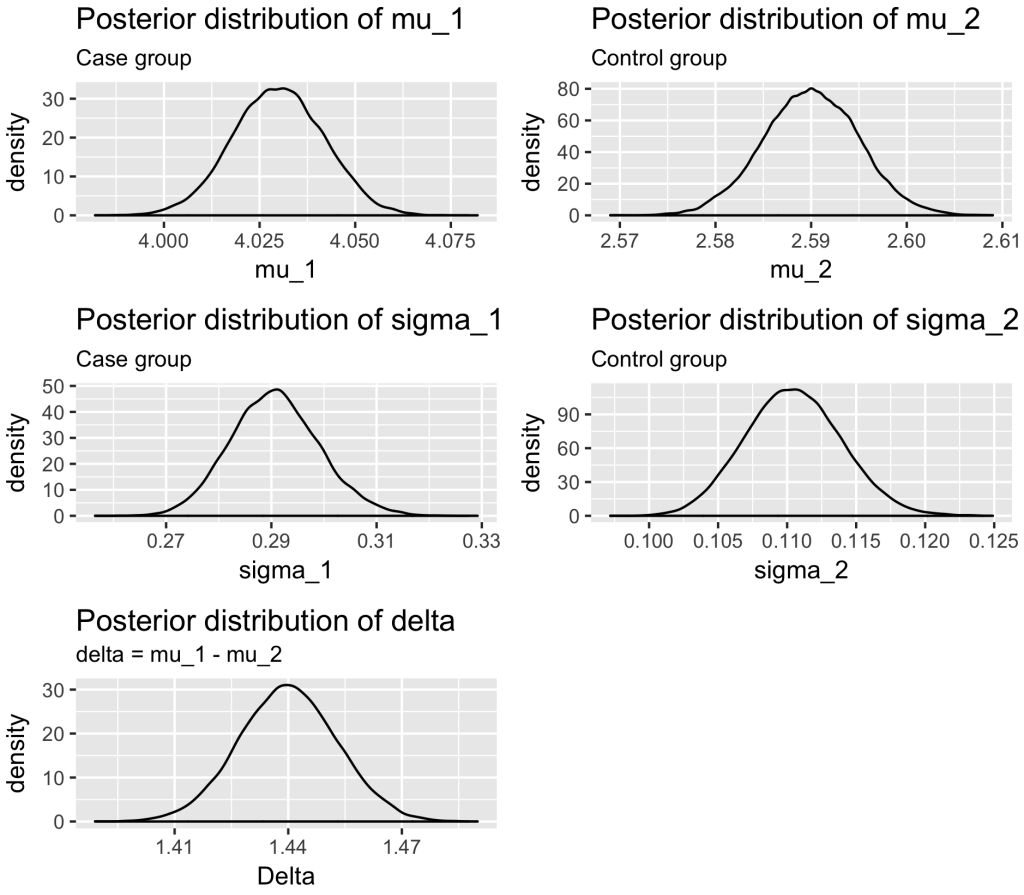

Figure 2: Posterior distributions with uninformative prior

In this example, as the fig. 2 and table 1 show, there is an important difference between μ_1 and μ_2. In fact, the credibility intervals (95%) of each are $(4.007, 4.053)$ and $(2.58, 2.6)$ and equivalently for δ: $(1.415, 1.465)$. With this prior, we can conclude that there is a difference between μ_1 and μ_2.

4.1.2 Informative prior with low variance

Table 2: Summary of analysis with informative prior (low variance)

parameters	mean	sd	2.5%	50%	97.5%
μ_1	4.0299	0.0119	4.007	4.03	4.053
μ_2	2.59	0.0046	2.581	2.59	2.599
δ	1.4399	0.0128	1.415	1.44	1.465
σ_1	0.291	0.0082	0.2755	0.2909	0.3081
σ_2	0.1021	0.0014	0.0994	0.1021	0.105

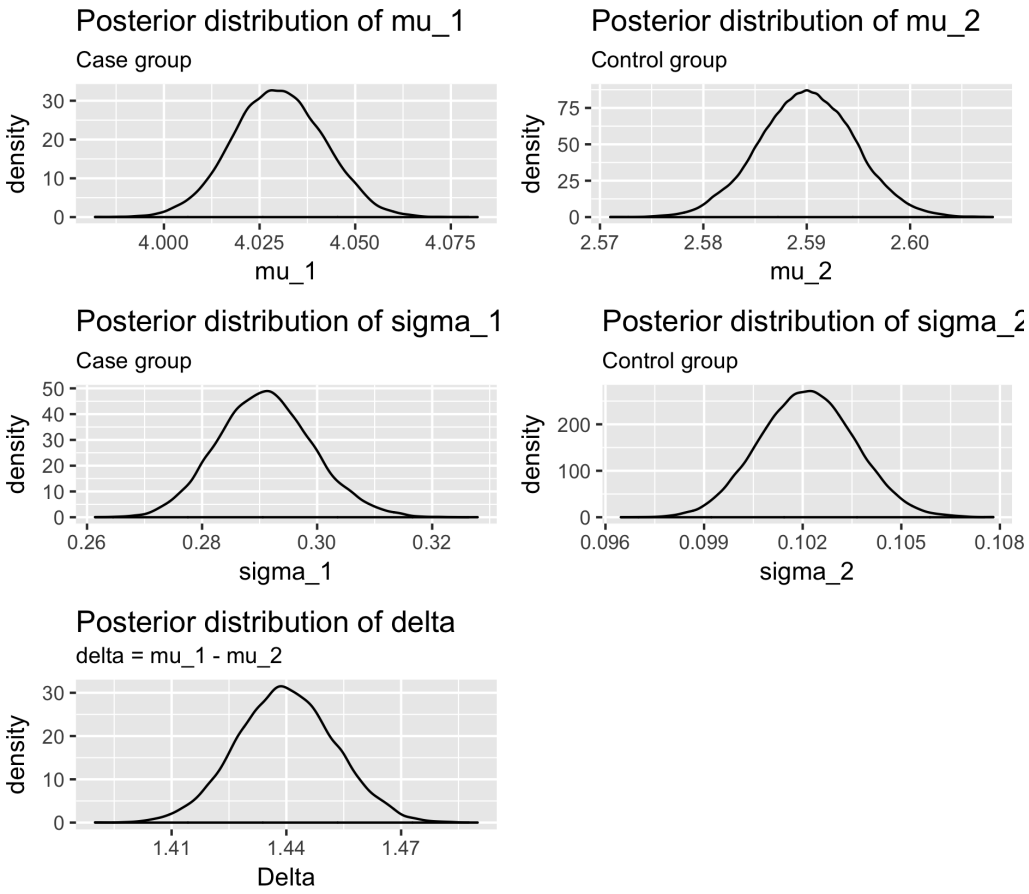

Figure 3: Posterior distributions with informative prior (low variance)

In this example, as the fig. 3 and table 2 show, there is an important difference between μ_1 and μ_2. In fact, the credibility intervals (95%) of each are $(4.007, 4.053)$ and $(2.581, 2.599)$ and equivalently for δ: $(1.415, 1.465)$. The conclusions are the same than the previous example (with the non informative prior). But is important to distinguish that the posterior distribution of σ_1 and σ_2 are a little bit different. (0.2907 vs 0.291 and 0.1105 vs 0.1021).

4.1.3 Informative prior with large variance

Table 3: Summary of analysis with informative prior (large variance)

parameters	mean	sd	2.5%	50%	97.5%
μ_1	4.0299	0.0119	4.007	4.03	4.053
μ_2	2.59	0.005	2.58	2.59	2.6
δ	1.4399	0.0129	1.415	1.44	1.465
σ_1	0.2907	0.0084	0.2748	0.2905	0.3081
σ_2	0.1099	0.0034	0.1034	0.1099	0.1168

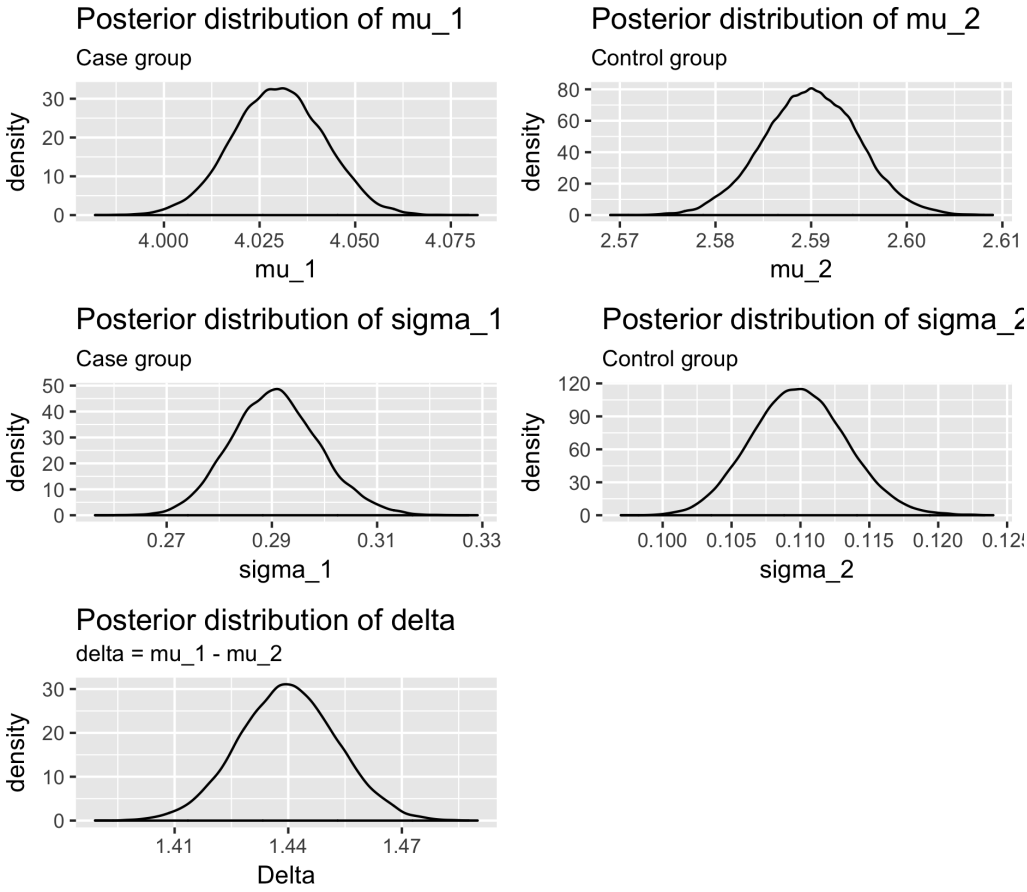

Figure 4: Posterior distributions with informative prior (large variance)

At last, the a informative a priori with large variance is used. The conclusions are the same than the previous two examples, but the estimations of the parameters and the standards deviation of them are almost the same with the first example (non informative). The differences are of the magnitude of 0.0001.

4.2 Posterior convergence

For the posterior distributions showed in the last subsections, two chains were used with the initial values:

$$\begin{aligned} \mu_1 = 10 \quad , \quad \mu_2 = 0 \quad , \quad \tau_1 = 0.1 \quad , \quad \tau_2 = 0.5 \\ \mu_1 = -10 \quad , \quad \mu_2 = -10 \quad , \quad \tau_1 = 0.2 \quad , \quad \tau_2 = 0.2 \end{aligned} \tag{5}$$

4.2.1 non informative prior

In order to determine if the chain have converged, The Gelman-Rubin statistics was tried. For the four parameters there is a good convergence, but for σ_2 maybe more iterations would be better. (There are little differences between the chains, but it is a very good convergence). Another analysis made was to check the autocorrelations coefficients. The first blue line is 1 because the lag is 0, so is the correlation between the same element. In other parts of the chain seems that there are no important correlation between the consecutive points (at least the shown).

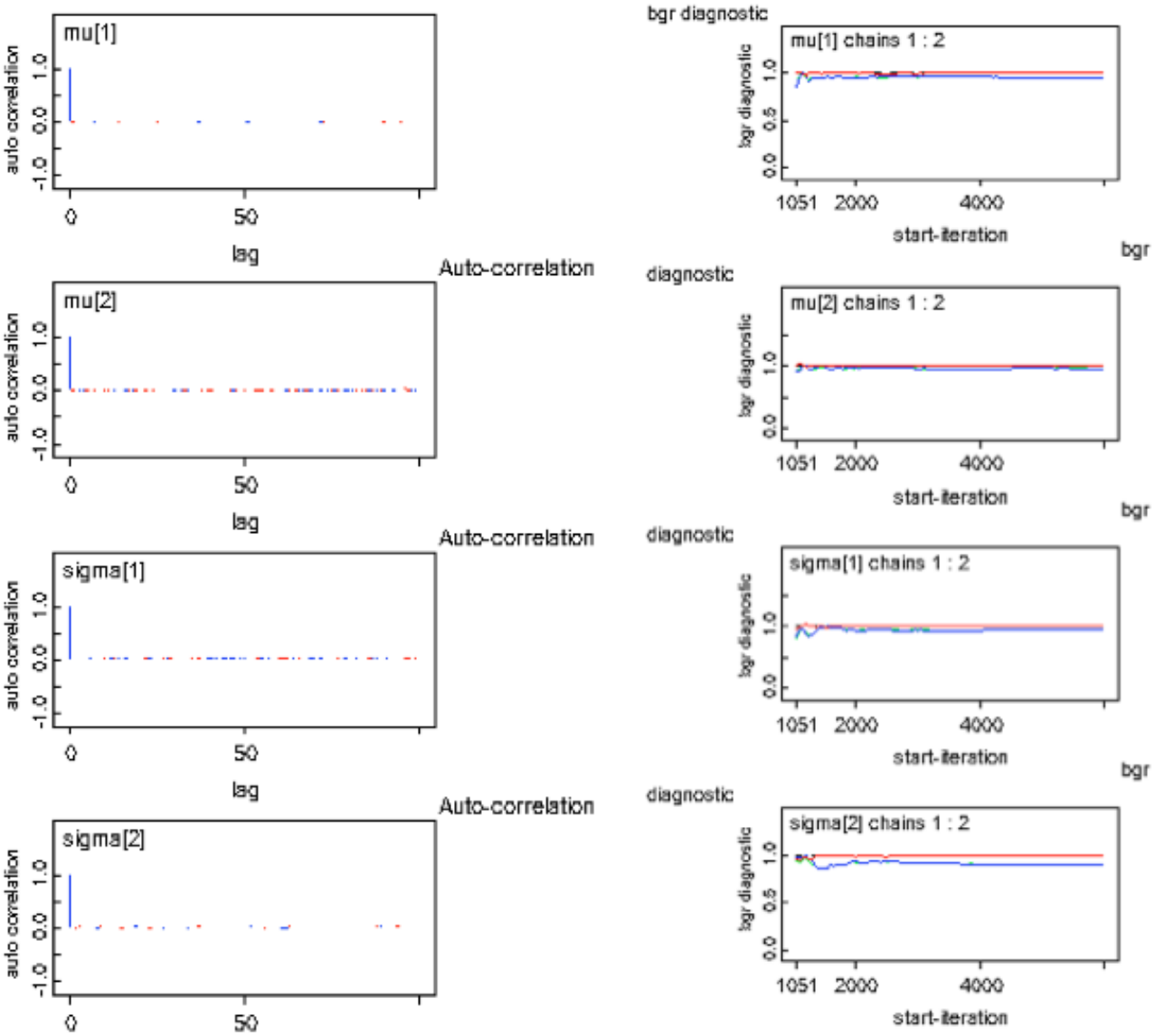

Figure 5: Autocorrelation and Gelman-Rubin statistic (non informative prior)

Now, the history of the parameters is shown in fig. 6. Is easy to see that the chain is exploring the same area and there are almost any point outside the central area of the distributions. So, it looks very good the convergence of the chain. The burning iterations were 1500. Before that number, the chain moves trying to explore the parameters domain (different initial points should converge to the same point). This is an indication that the optimum is still not achieved, and if these iterations are considered, the analysis will be biased. The autocorrelation analysis is very important. As the MCMC name states, is a Markov Chain, so the position on the next iterations depends directly on the position in the current iteration. It is important to thin in order to de-correlate each iteration (although some authors says that most of the times it is not necessary to thinning). The MC error is good when checking the fig. 7. According with the BIAS project, the MC error compares with the SD of the posterior distribution in a fraction $\frac{SD}{\sqrt{(iterations)}}$ also that the thumb error is that should be less than 1% of the posterior SD is a good value. In all the estimates this is achieved.

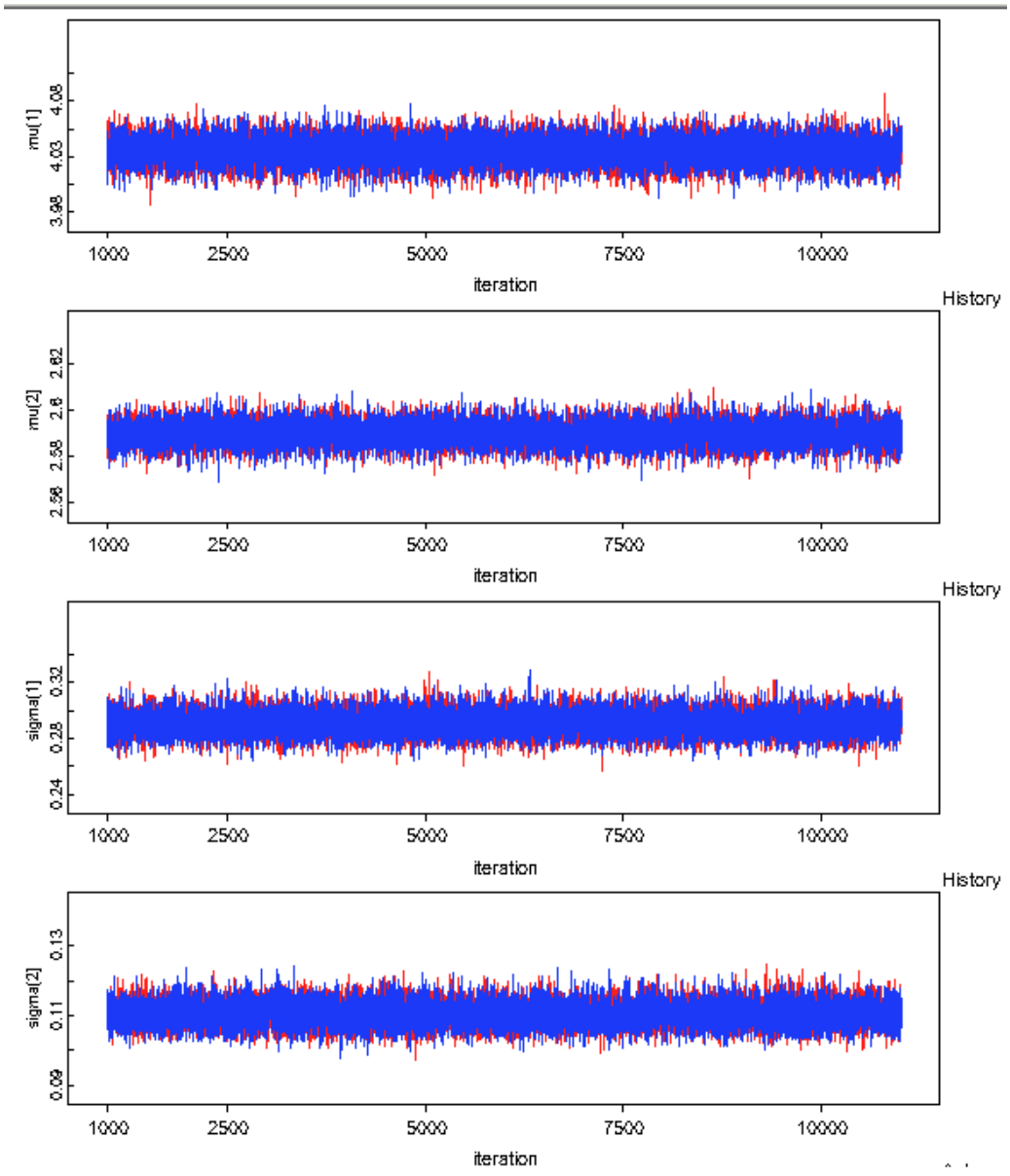

Figure 6: History of parameters (non informative)

Node statistics

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
delta	1.44	0.0129	9.063E-5	1.415	1.44	1.465	1001	20000
mu[1]	4.03	0.01186	8.497E-5	4.007	4.03	4.053	1001	20000
mu[2]	2.59	0.004976	3.549E-5	2.58	2.59	2.6	1001	20000
sigma[1]	0.2907	0.008403	6.414E-5	0.2748	0.2905	0.3081	1001	20000
sigma[2]	0.1105	0.0035	2.34E-5	0.1038	0.1104	0.1175	1001	20000
tau[1]	11.86	0.6841	0.005221	10.53	11.85	13.24	1001	20000
tau[2]	82.19	5.201	0.03486	72.41	82.03	92.75	1001	20000

Figure 7: Summary of model (non informative)

4.3 Informative prior with low variance

The Gelman-Rubin statistics was tried for this example too. For the four parameters there is a good convergence, but again, σ_2 could have better performance if more iterations are added. (There are little differences between the chains, but it is a very good convergence). The autocorrelation coefficients also were calculated. The conclusions are the same than the previous example.

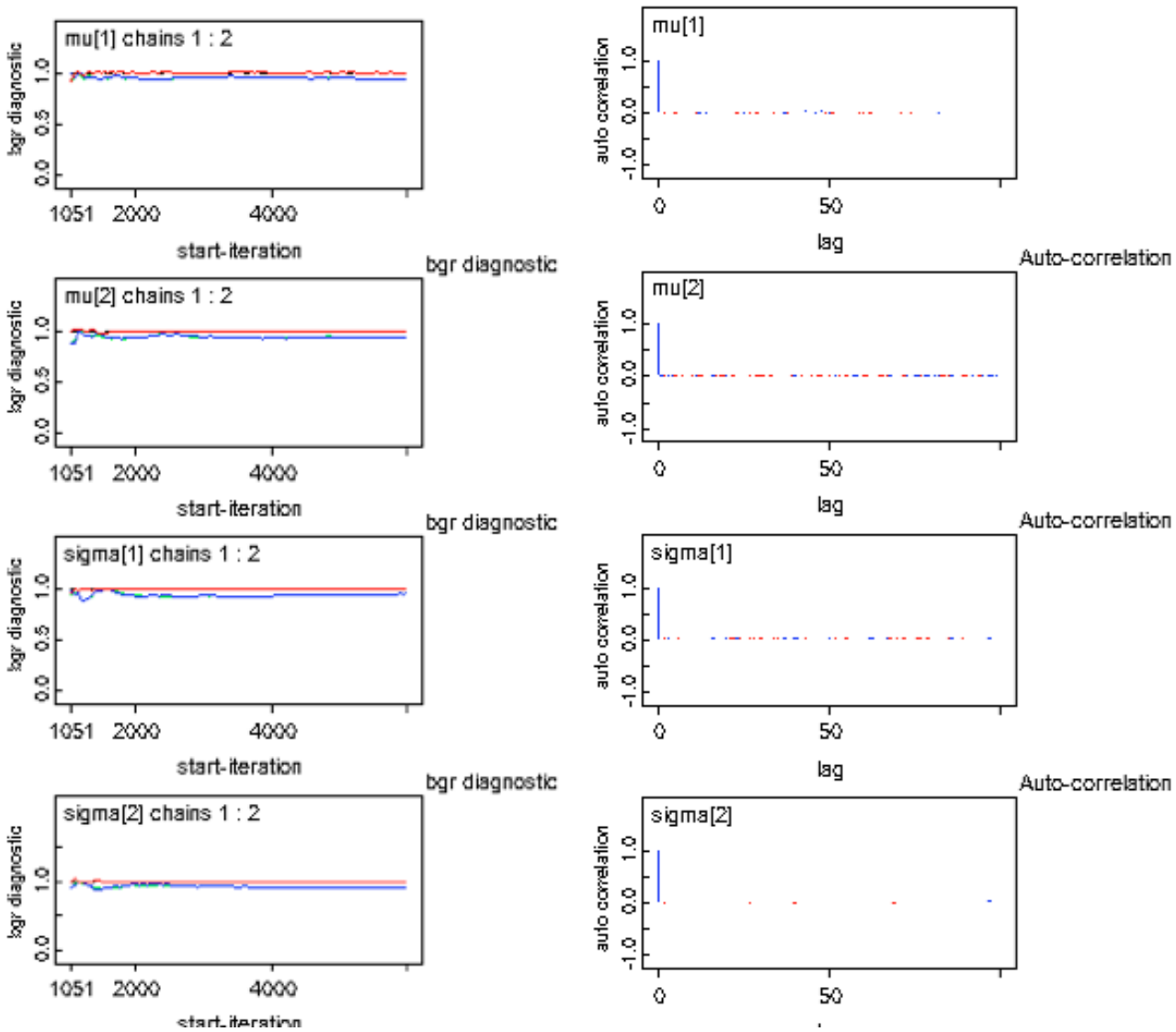

Figure 8: Autocorrelation and Gelman-Rubin statistic (low variance prior)

Again, the history of the parameters is shown in fig. 9. The chains seems to have converged and there is no problem associated with convergence. In this example the burning iterations were 1500 too. The values of the MC error are also below the 1% thumb rule.

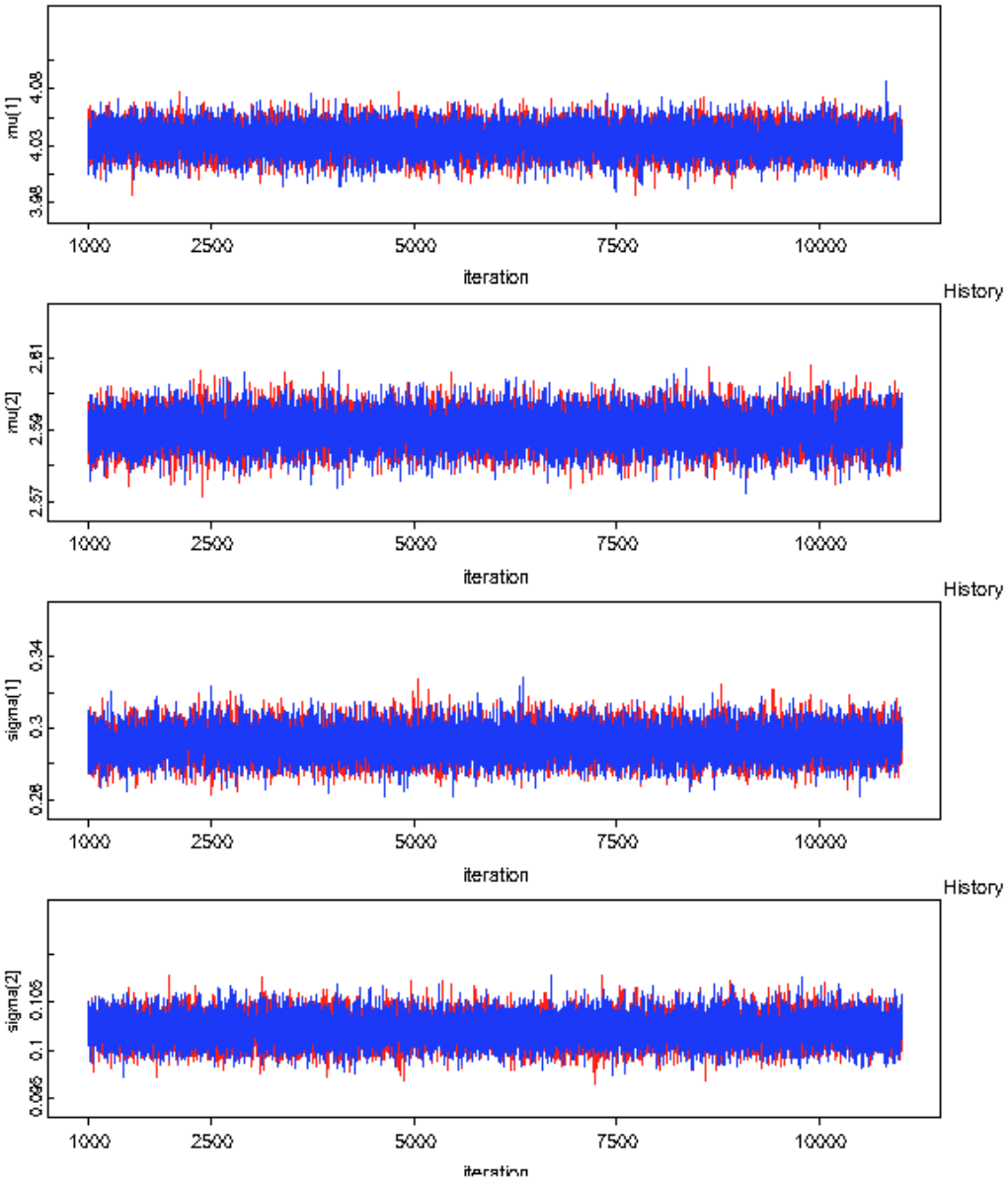

Figure 9: History of parameters (low variance prior)

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
delta	1.44	0.01274	8.634E-5	1.415	1.44	1.465	1001	20000
mu[1]	4.03	0.01184	7.799E-5	4.007	4.03	4.053	1001	20000
mu[2]	2.59	0.004595	3.354E-5	2.581	2.59	2.599	1001	20000
sigma[1]	0.291	0.008238	5.756E-5	0.2755	0.2909	0.308	1001	20000
sigma[2]	0.1021	0.001446	9.963E-6	0.09937	0.1021	0.105	1001	20000
tau[1]	11.84	0.6684	0.004674	10.54	11.82	13.18	1001	20000
tau[2]	95.91	2.715	0.0187	90.73	95.86	101.3	1001	20000

Figure 10: Summary of model (low variance prior)

4.4 Informative prior with large variance

The Gelman-Rubin statistics was tried for this example too. For the four parameters there is a good convergence, but again, σ_2 could have better performance if more iterations are added. (There are little differences between the chains, but it is a very good convergence). The autocorrelation coefficients also were calculated. The conclusions are the same than the previous two examples.

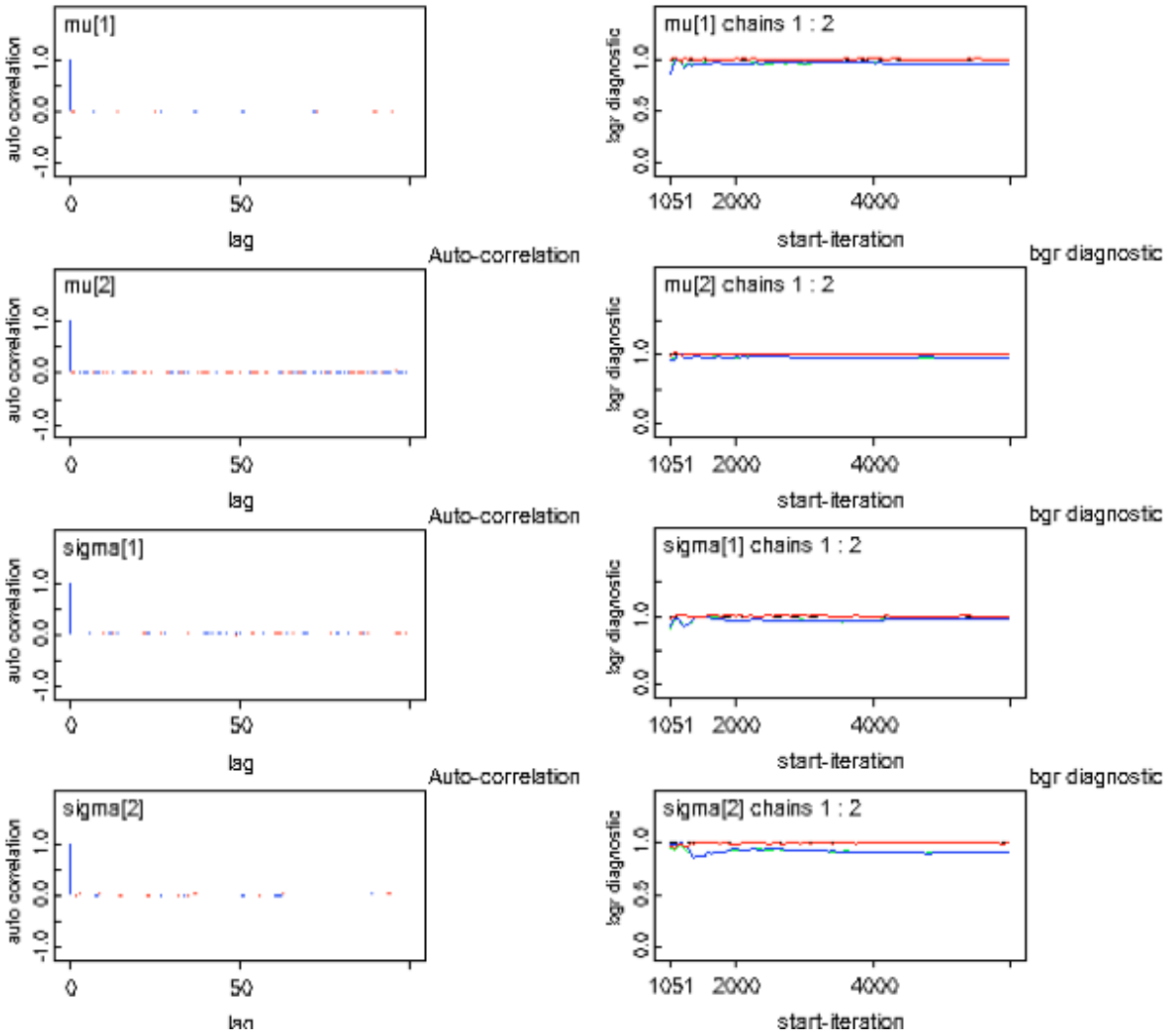

Figure 11: Autocorrelation and Gelman-Rubin statistic (large variance prior)

Again, the history of the parameters is shown in fig. 12. The chains seems to have converged and there is no problem associated with convergence. In this example the burning iterations were 1500 too. The values of the MC error are also below the 1% thumb rule.

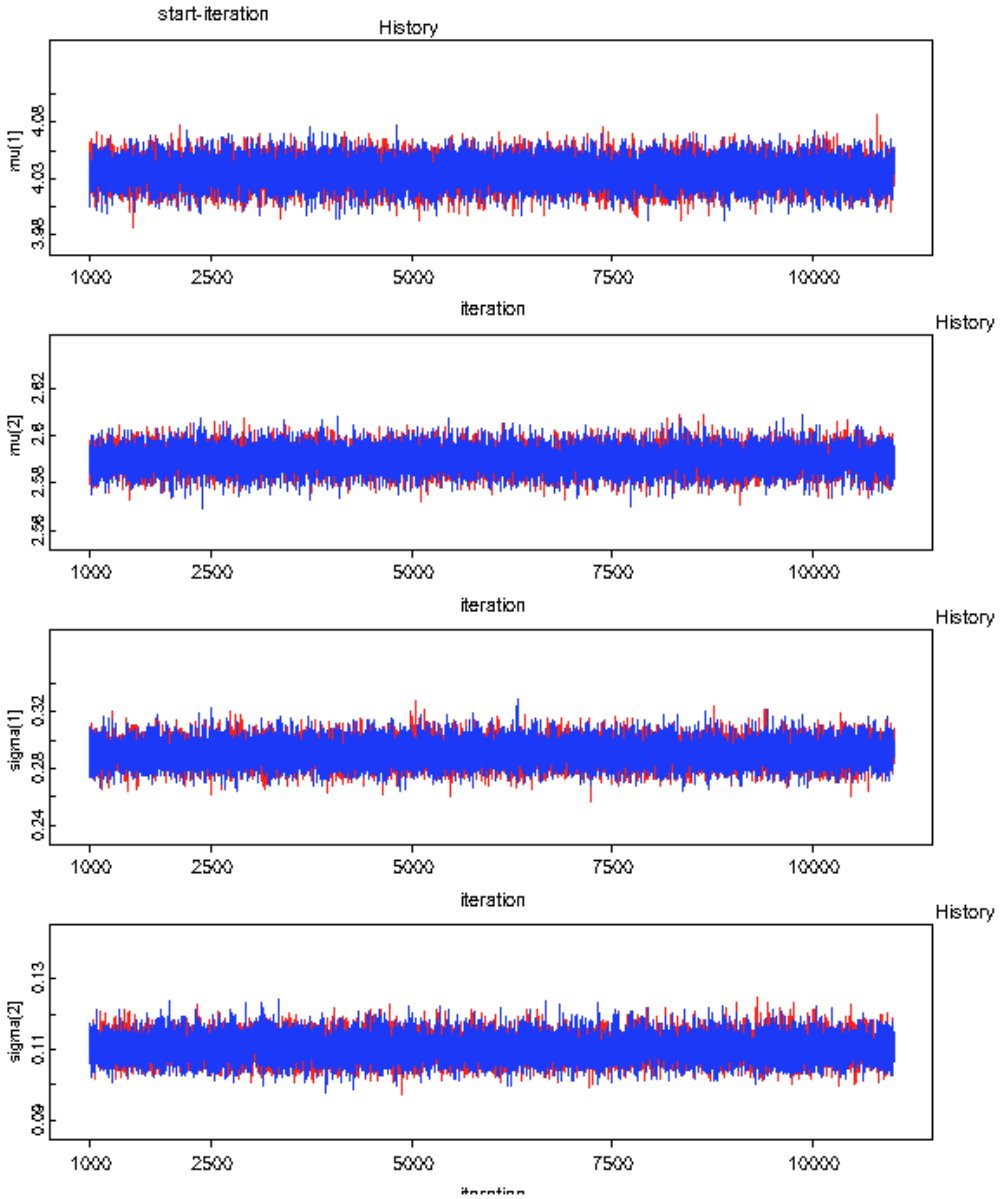

Figure 12: History of parameters (large variance prior)

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
delta	1.44	0.0129	9.063E-5	1.415	1.44	1.465	1001	20000
mu[1]	4.03	0.01186	8.497E-5	4.007	4.03	4.053	1001	20000
mu[2]	2.59	0.004976	3.549E-5	2.58	2.59	2.6	1001	20000
sigma[1]	0.2907	0.008403	6.414E-5	0.2748	0.2905	0.3081	1001	20000
sigma[2]	0.1105	0.0035	2.34E-5	0.1038	0.1104	0.1175	1001	20000
tau[1]	11.86	0.6841	0.005221	10.53	11.85	13.24	1001	20000
tau[2]	82.19	5.201	0.03486	72.41	82.03	92.75	1001	20000

Figure 13: Summary of model (large variance prior)

5 Model comparison

In this assignment three prior distributions were used: the first was an uninformative prior, while the second and third were informative ones. For these two, one was a distribution with low variance and the other one was a distribution with large variance. The intuition behind the apriori non informative is that this has a very large variance, so the model does not restricts the parameter into some space in the domain of the parameter. This way, a normal distribution with a very big variance can be used as a non informative prior. Now, the informative distribution with large variance is expected to behave similar than the non informative, as can be see in the fig. 14. Also, in this graph is easy to how is similar to the informative distribution with small variance (except in the σ_2 parameter).

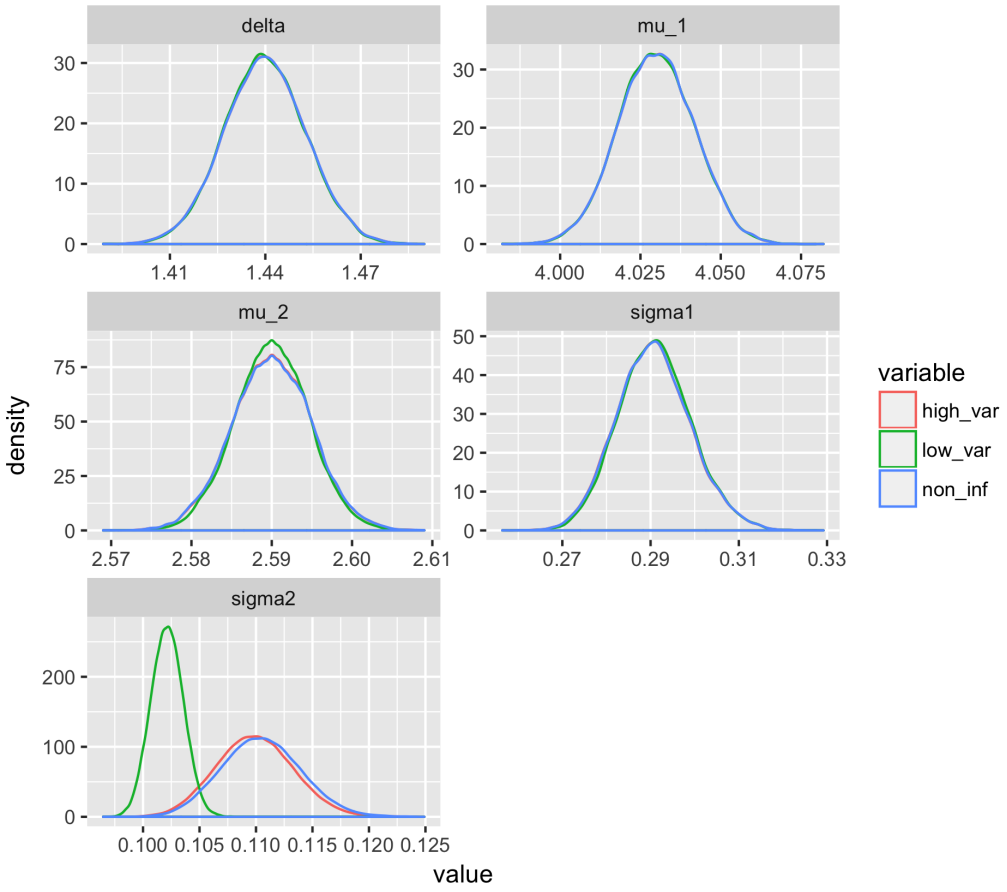

Figure 14: Comparison of posterior distributions with different apriori

Although there is a difference in σ_2, when the scale is observed, then this can be negligible (compared with the original a priori which has a mean of 100).

In summary, the specification of a prior far from the likelihood and also with small variance, can affect the posterior distribution. But in this example it just change the mean of the distributions just by a small amount. It is important to say that the modifications of the variance can have more important changes in the posterior distribution.

A R graphs code

For the plots in R, the package tidyverse and R2OpenBugs were used. Also, the package wine was used. BUGS models were sent via email

```
#Load the OpenBUGS Package - make sure XQuartz is running
library(tidyverse)
library(R2OpenBUGS)

#define the model
nummodel <- function(){
    # Model
    # distribution of the sample means
    barx ~ dnorm(mu[1], tau.mean[1])
    bary ~ dnorm(mu[2], tau.mean[2])
    # distribution of the sample variances
    s2x  ~ dgamma(n1, tau1.2)
    s2y  ~ dgamma(n2, tau2.2)
    # Priors
    for (j in 1 : 2){
      mu[j] ~ dnorm(0, 0.000001)
      # standard deviation of error distribution
      sigma[j] <- sqrt(1 / tau[j])}
    tau[1] ~ dgamma(0.01,0.01)
    tau[2] ~ dgamma(0.01,0.01)
    # introduce the difference between the means (cases-controls)
    delta <- mu[1] - mu[2]
    # introduce the precisions of the means
    tau.mean[1] <- tau[1]*n*N
    tau.mean[2] <- tau[2]*m*N
    # introduce the scale of the gamma distribution (chi-square)
    tau1.2 <- n*N* tau[1]/2
    tau2.2 <- m*N*tau[2]/2
    # introduce the shape of the gamma distribution (chi-square)
    n1 <- (n*N-1)/2
    n2 <- (m*N-1)/2
}

# write the model code out to a file
write.model(nummodel, "nummodel.txt")
model.file1 = paste(getwd(),"nummodel.txt", sep="/")
## and let's take a look:
file.show("nummodel.txt")

#prepare the data for input into OpenBUGS
n = 12
m=10
N = 50
barx = 4.03
bary = 2.59
s2x = 0.0841
s2y = 0.0121
data <- list ("n", "m", "N", "barx", "bary", "s2x", "s2y")
```

```
# initialization of variables
inits <- function(){
  list (chain1 <- c("mu[1]=10", "mu[2] = 0"", "tau[1] = 0.1", "tau[2] = 0.5"),
        chain2 <- c("mu[1]=-10", "mu[2] = -10", "tau[1] = 0.2", "tau[2] = 0.2"))}

#set the WINE working directory and the directory to OpenBUGS - change the OpenBUGS.exe location as
    necessary
WINE="/usr/local/bin/wine"
WINEPATH="/usr/local/bin/winepath"
OpenBUGS.pgm="/Users/salvadorgarcia/.wine/drive_c/Program
    Files/OpenBUGS/OpenBUGS323/OpenBUGS.exe"

#these are the parameters to save
parameters = c("mu", "tau", "delta", "sigma")

#run the model
schools.sim <- bugs(data, inits,
                    model.file = model.file1,
                    parameters=parameters,
                    n.chains = 2,
                    n.iter = 11000,
                    OpenBUGS.pgm=OpenBUGS.pgm,
                    WINE=WINE,
                    WINEPATH=WINEPATH,
                    useWINE=T,
                    n.burnin = 1500)

schools.sim$summary %>% round(4) %>% write.table(pipe("pbcopy"))

modelo_noninf <- schools.sim
p1 <- schools.sim$sims.list$delta %>%
  data.frame() %>% setNames(c("Delta")) %>%
  ggplot(aes(x= Delta))+
  geom_density() +
  labs( title = "Posterior distribution of delta", subtitle = "delta = mu_1 - mu_2")

p2 <- schools.sim$sims.list$mu[,1] %>%
  data.frame() %>% setNames(c("mu_1")) %>%
  ggplot(aes(x= mu_1))+
  geom_density() +
  labs( title = "Posterior distribution of mu_1", subtitle = "Case group")

p3 <- schools.sim$sims.list$mu[,2] %>%
  data.frame() %>% setNames(c("mu_2")) %>%
  ggplot(aes(x= mu_2))+
  geom_density() +
  labs( title = "Posterior distribution of mu_2", subtitle = "Control group")

p4 <- schools.sim$sims.list$sigma[,1] %>%
  data.frame() %>% setNames(c("sigma_1")) %>%
  ggplot(aes(x= sigma_1))+
```

```
  geom_density() +
  labs( title  = "Posterior distribution  of sigma_1", subtitle  = "Case group")

p5 <- schools.sim$sims.list$sigma[,2]  %>%
  data.frame() %>% setNames(c("sigma_2")) %>%
  ggplot(aes(x= sigma_2))+
  geom_density() +
  labs( title  = "Posterior distribution  of sigma_2", subtitle  = "Control group")

ggsave(plot  = grid.arrange(p2,p3,p4,p5,p1,  ncol  = 2),
       filename  = "../12_Assignment2/imgs/01_posterior_pars.png",
       width = 16, height = 14, units  = "cm")
```

Code for the comparison

```
# comparison

# mu[1]
mu_1 <- data.frame(high_var = modelo_high$sims.list$mu[,1],
                   low_var  = modelo_low$sims.list$mu[,1],
                   non_inf  = modelo_noninf$sims.list$mu[,1]) %>%
  mutate(par = "mu_1")
# mu[2]
mu_2 <- data.frame(high_var = modelo_high$sims.list$mu[,2],
                   low_var  = modelo_low$sims.list$mu[,2],
                   non_inf  = modelo_noninf$sims.list$mu[,2]) %>%
  mutate(par = "mu_2")
# sigma[1]
sigma_1 <- data.frame(high_var = modelo_high$sims.list$sigma[,1],
                      low_var  = modelo_low$sims.list$sigma[,1],
                      non_inf  = modelo_noninf$sims.list$sigma[,1])  %>%
  mutate(par = "sigma1")

# sigma[2]
sigma_2 <- data.frame(high_var = modelo_high$sims.list$sigma[,2],
                      low_var  = modelo_low$sims.list$sigma[,2],
                      non_inf  = modelo_noninf$sims.list$sigma[,2])  %>%
  mutate(par = "sigma2")
# delta
delta  <- data.frame(high_var = modelo_high$sims.list$delta,
                     low_var  = modelo_low$sims.list$delta,
                     non_inf  = modelo_noninf$sims.list$delta)  %>%
  mutate(par = "delta")

pars_df <- rbind(mu_1, mu_2, sigma_1, sigma_2, delta)
p1 <- pars_df %>% gather(variable, value, high_var:non_inf) %>%
  ggplot(aes(x = value,  color  = variable)) +
  geom_density() +
  facet_wrap(~par, scale  = "free",  ncol  = 2)

ggsave(plot  = p1,
       filename  = "../12_Assignment2/imgs/comparison.png",
       width = 16, height = 14, units  = "cm")
```