Homework IV

Bayesian Data Analysis - UPC Spring 2022

Alexander J Ohrt

17 april, 2022

Exercise 4.4: Two Methods of Training Workers; Comparison of Means

The Human Resources Department of a large company wishes to compare two methods of training industrial workers to perform a skilled task. Twenty workers are selected: 10 of them are randomly assigned to be trained using method A, and the other 10 are assigned to be trained using method B. After the training is complete, all the workers are tested on the speed of performance at the task. The times taken to complete the task are

```
method.A <- c(115, 120, 111, 123, 116, 121, 118, 116, 127, 129)
method.B <- c(123, 131, 113, 119, 123, 113, 128, 126, 125, 128)
df <- rbind(c(method.A,mean(method.A), sd(method.A)), c(method.B,mean(method.B), sd(method.B)))
rownames(df) <- c("Method A", "Method B")
colnames(df) <- c(rep("", 10), "Mean", "Standard Error")
```

Table 1: Times Taken to Complete Task for each Method

											Mean	Standard Error
Method A	115	120	111	123	116	121	118	116	127	129	119.6	5.581716
Method B	123	131	113	119	123	113	128	126	125	128	122.9	6.172520

a) Find Posterior Distributions of Parameters μ_A, μ_B

We assume that the observations come from $N(\mu_A, \sigma)$ and $N(\mu_A, \sigma)$, where $\sigma = 6$. Use independent $N(m, s)$ prior distributions for μ_A, μ_B, where $m = 100, s = 20$. The posterior distributions of the parameters are found using Stan (via R).

First we define the Stan model and fit it.

```
# Define model and call stan.
data_list <- list(
  nA = length(method.A),
  nB = length(method.B),
  tA = method.A,
  tB = method.B
)

fit <- stan("4-4_training_workers.stan", iter = 1000, chains = 4,
            data = data_list, seed = 1)
```

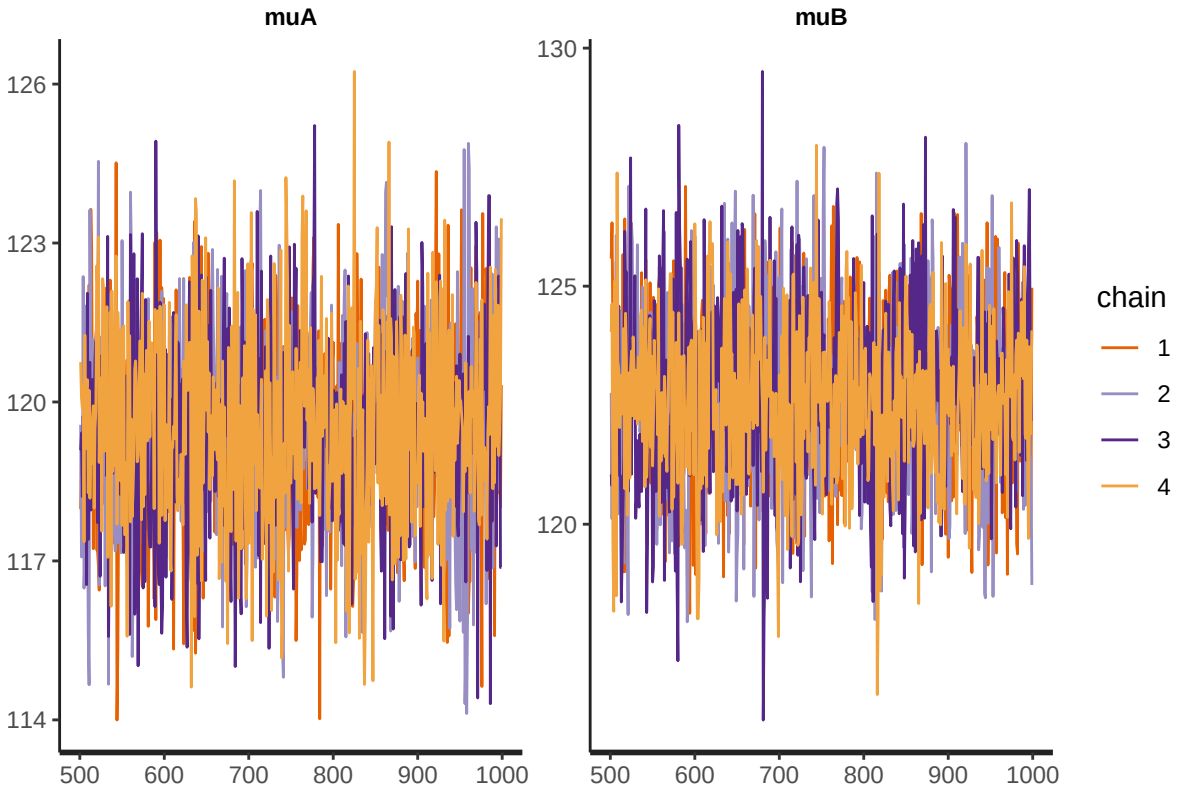
```
## Trying to compile a simple C file
```

The convergence analysis shows that the chains have converged, because `Rhat =1`. Moreover, the traceplots seem to show that the chains have converged to similar values.

```
# Convergence analysis.
print(fit)
```

```
## Inference for Stan model: 4-4_training_workers.
## 4 chains, each with iter=1000; warmup=500; thin=1;
## post-warmup draws per chain=500, total post-warmup draws=2000.
##
##        mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## muA  119.47    0.04 1.83 115.77 118.22 119.49 120.65 123.06  2183    1
## muB  122.70    0.04 1.86 119.02 121.42 122.73 123.97 126.29  2018    1
## lp__ -10.73    0.04 0.94 -13.13 -11.11 -10.45 -10.05  -9.81   668    1
##
## Samples were drawn using NUTS(diag_e) at Sun Apr 17 21:26:01 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
traceplot(fit)
```




```
posterior <- as.data.frame(fit)
head(posterior)
```

```
##        muA      muB       lp__
## 1 117.9793 125.5803 -11.241568
## 2 119.0696 126.3306 -11.651754
## 3 118.8418 124.1635 -10.132137
## 4 118.1667 124.8950 -10.682264
## 5 120.1579 122.0561  -9.915097
```

```
## 6 118.8075 123.2357  -9.876855
```

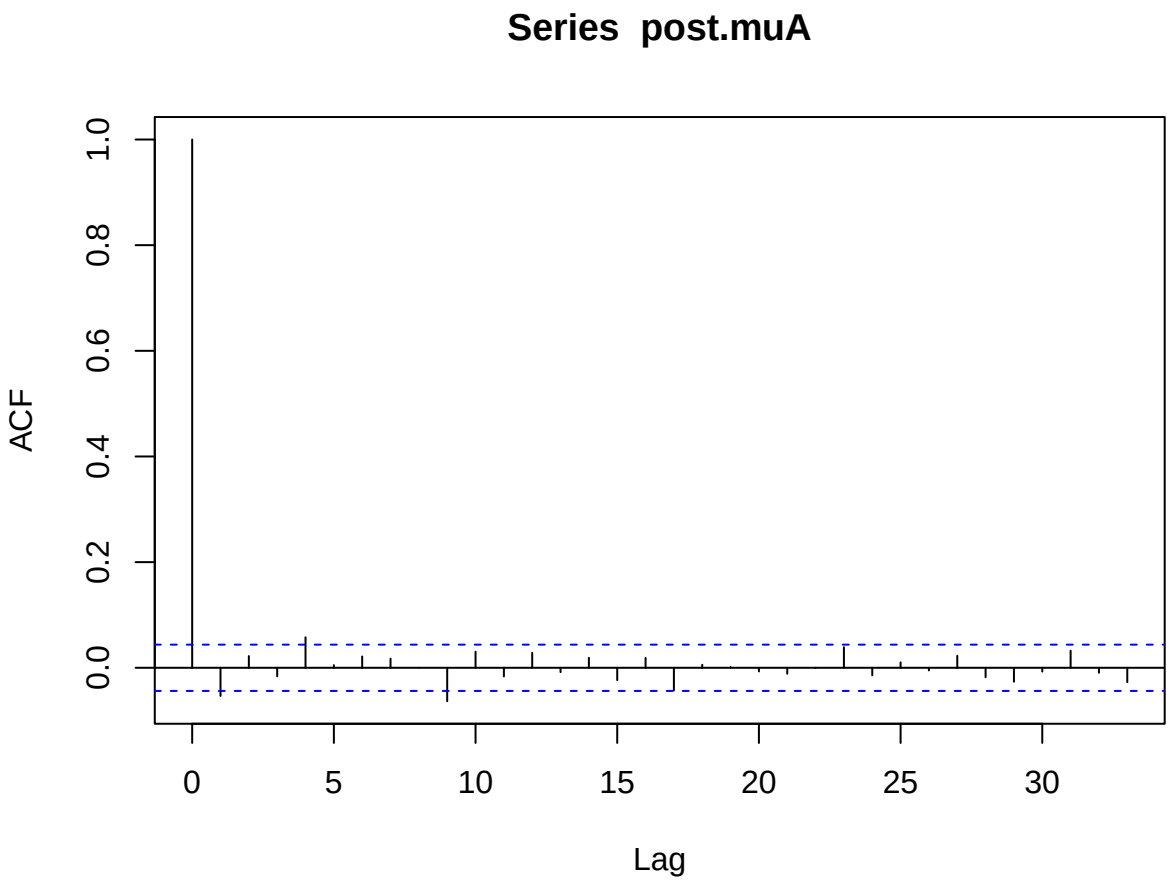
```
dim(posterior)
```

```
## [1] 2000    3
```

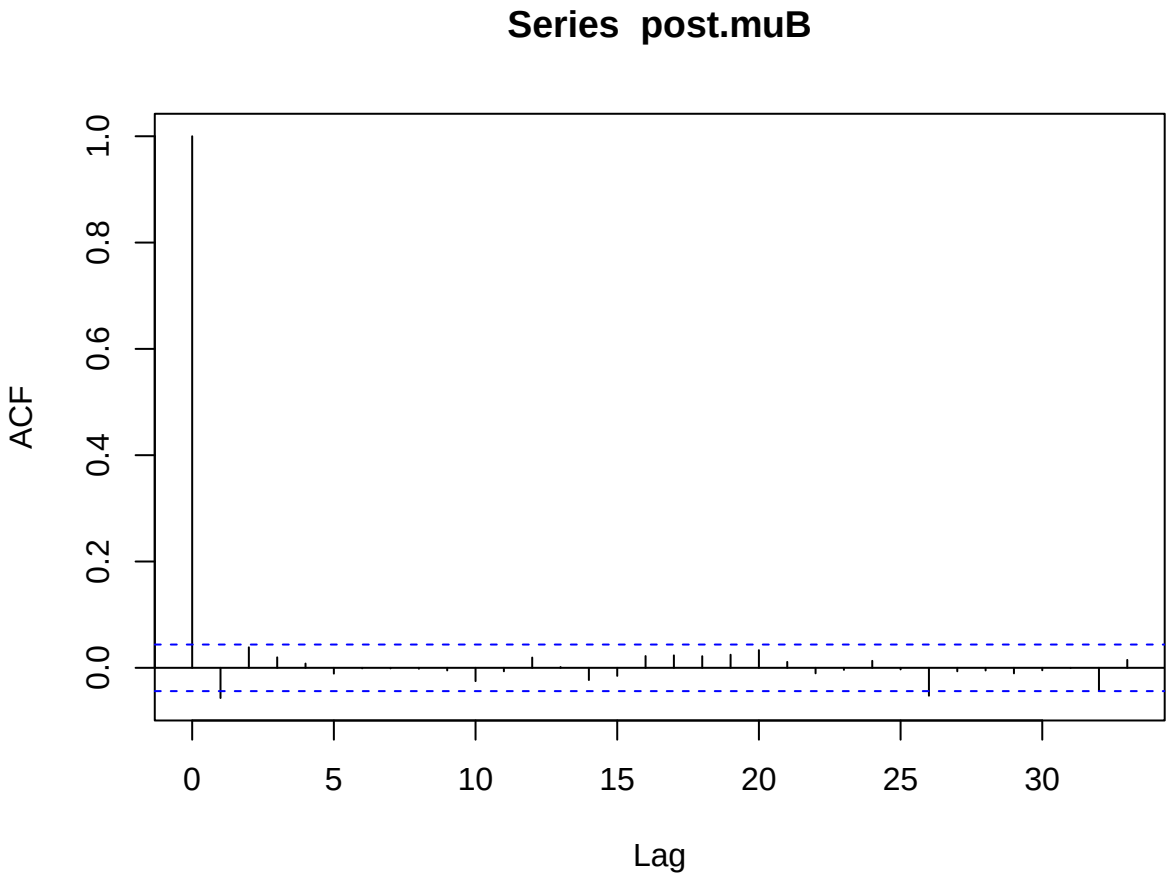
```
post.muA <- posterior[,1]
post.muB <- posterior[,2]
```

From the autocorrelation plots below we can see that there is little to no significant autocorrelation, which is a good diagnostic result for the MCMC sampler.

```
acf(post.muA)
```

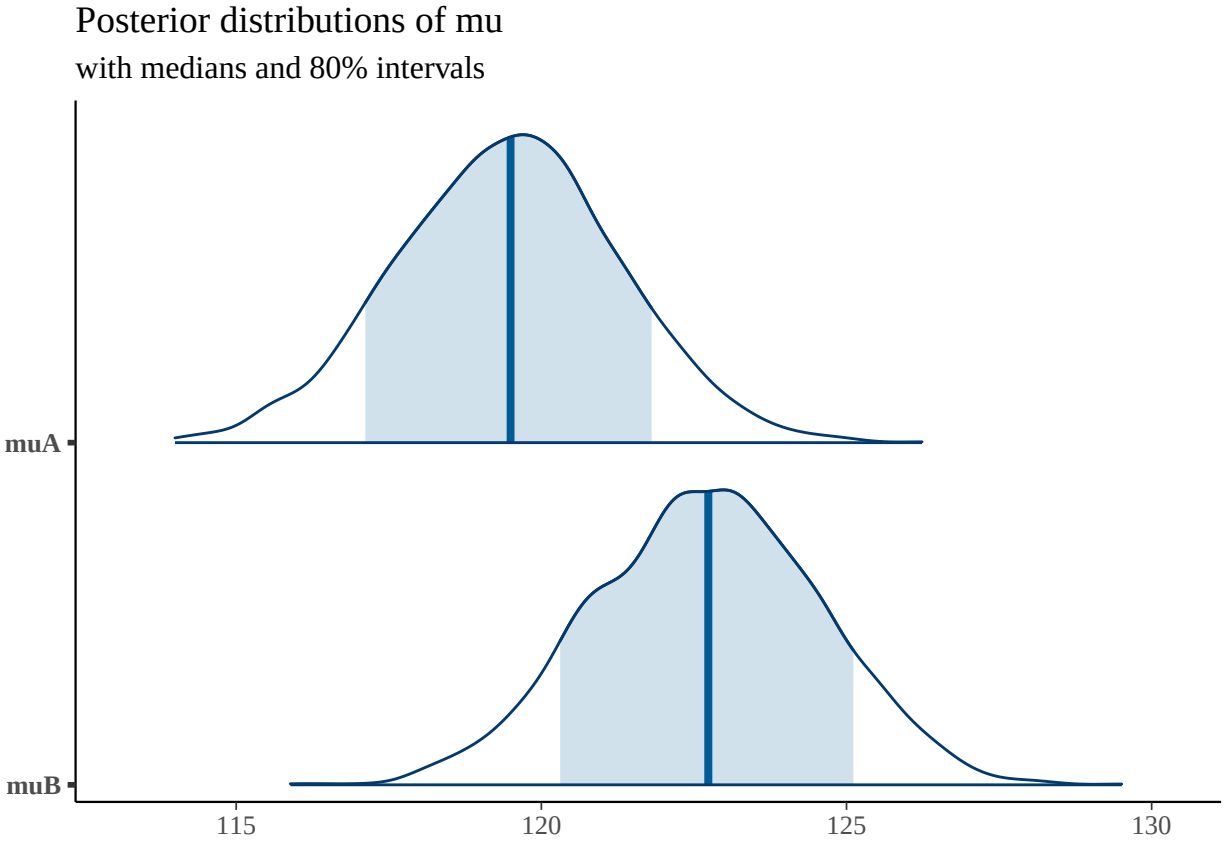



```
acf(post.muB)
```



The posterior distributions are plotted with medians and 80% intervals below.

```
plot_title <- ggtitle("Posterior distributions of mu", "with medians and 80% intervals")
mcmc_areas(posterior,
  pars = c("muA", "muB"),
  prob = 0.8) + plot_title
```



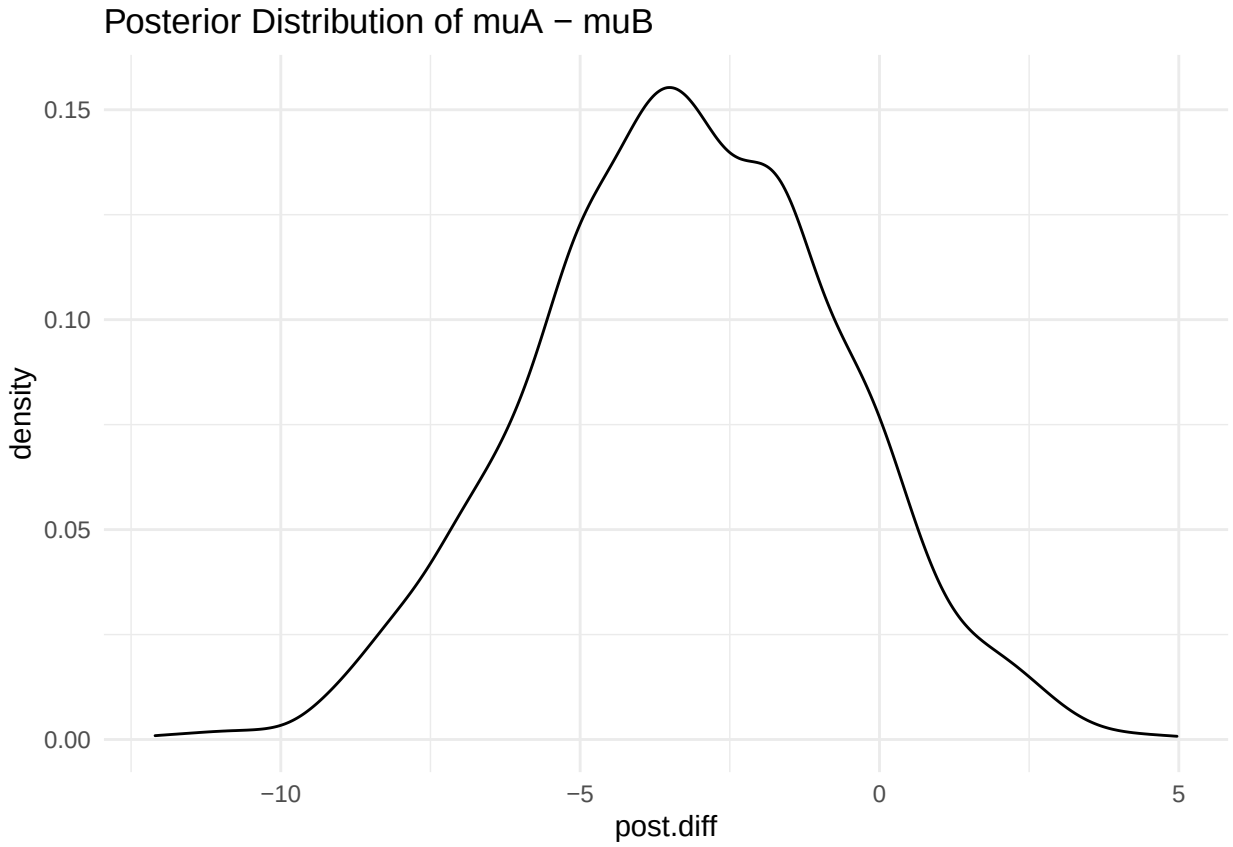
The Stan code from file `4-4_training_workers.stan` is given below for completeness. Note that all the `.stan` files are also submitted.

```
data{
  int<lower=0> nA;
  int<lower=0> nB;

  real<lower=0> tA[nA];
  real<lower=0> tB[nB];
}

parameters{
  real muA;
  real muB;
}

model{
  tA ~ normal(muA, 6);
  tB ~ normal(muB, 6);

  muA ~ normal(100, 20);
  muB ~ normal(100, 20);
}
```

b) Find Posterior Distribution of $\mu_A - \mu_B$

The posterior distribution of $\mu_A - \mu_B$ is found and plotted below.

```r
post.diff <- post.muA - post.muB
ggplot(tibble(post.diff)) +
  geom_density(aes(post.diff)) +
  theme_minimal() +
  ggtitle("Posterior Distribution of muA - muB")
```



c) Find a 95% Bayesian Credible Interval (CI) for $\mu_A - \mu_B$

A 95% quantile CI for $\mu_A - \mu_B$ is found below.

```r
(CI.perc <- quantile(post.diff, probs = c(0.025, 0.975)))
```

```
##      2.5%     97.5%
## -8.226049  1.763004
```

d) Repeat the Previous Problems, Supposing that σ is Unknown

The steps are repeated supposing that σ is unknown. Notice that we still assume that it is equal in both populations. When σ is unknown, we have to define a prior distribution for this parameter as well. We will solve the problem assuming two different priors for σ;

1) $\pi_1 \sim N(6, 100)$ (less informative)
2) $\pi_1 \sim N(6, 10)$ (more informative)

Case 1)

Using the less informative prior among the two, the results are given in the following.

```
fit2 <- stan("4-4_training_workers2.stan", iter = 1000, chains = 4,
             data = data_list, seed = 1)
```

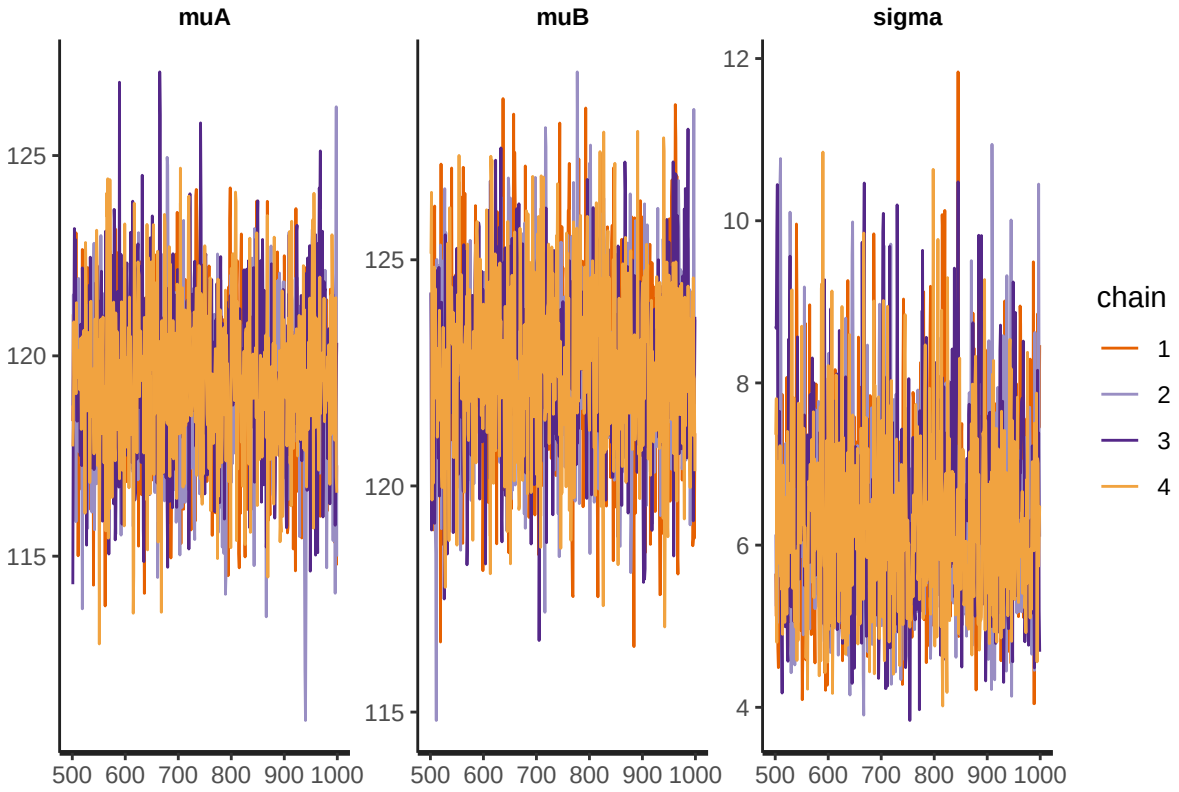
```
## Trying to compile a simple C file
```

The convergence analysis shows that the chains have converged.

```
# Convergence analysis.
print(fit2)
```

```
## Inference for Stan model: 4-4_training_workers2.
## 4 chains, each with iter=1000; warmup=500; thin=1;
## post-warmup draws per chain=500, total post-warmup draws=2000.
##
##         mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## muA   119.35    0.05 2.00 115.31 118.10 119.38 120.68 123.18  1482    1
## muB   122.68    0.05 1.94 118.79 121.43 122.70 123.91 126.68  1414    1
## sigma   6.34    0.03 1.16   4.55   5.52   6.18   6.97   9.13  1278    1
## lp__  -45.36    0.04 1.25 -48.62 -45.98 -45.06 -44.41 -43.90   967    1
##
## Samples were drawn using NUTS(diag_e) at Sun Apr 17 22:01:10 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
traceplot(fit2)
```




```
posterior2 <- as.data.frame(fit2)
head(posterior2)
```

```
##        muA      muB    sigma      lp__
```

```
## 1 118.3713 121.9154 5.648676 -44.07501
## 2 121.0362 121.9142 5.123533 -44.64660
## 3 116.7179 123.6629 6.687170 -45.11593
## 4 120.0099 121.6072 5.559775 -44.05935
## 5 118.7621 119.0277 4.727198 -47.86441
## 6 117.0429 120.3615 4.489503 -48.09736
```

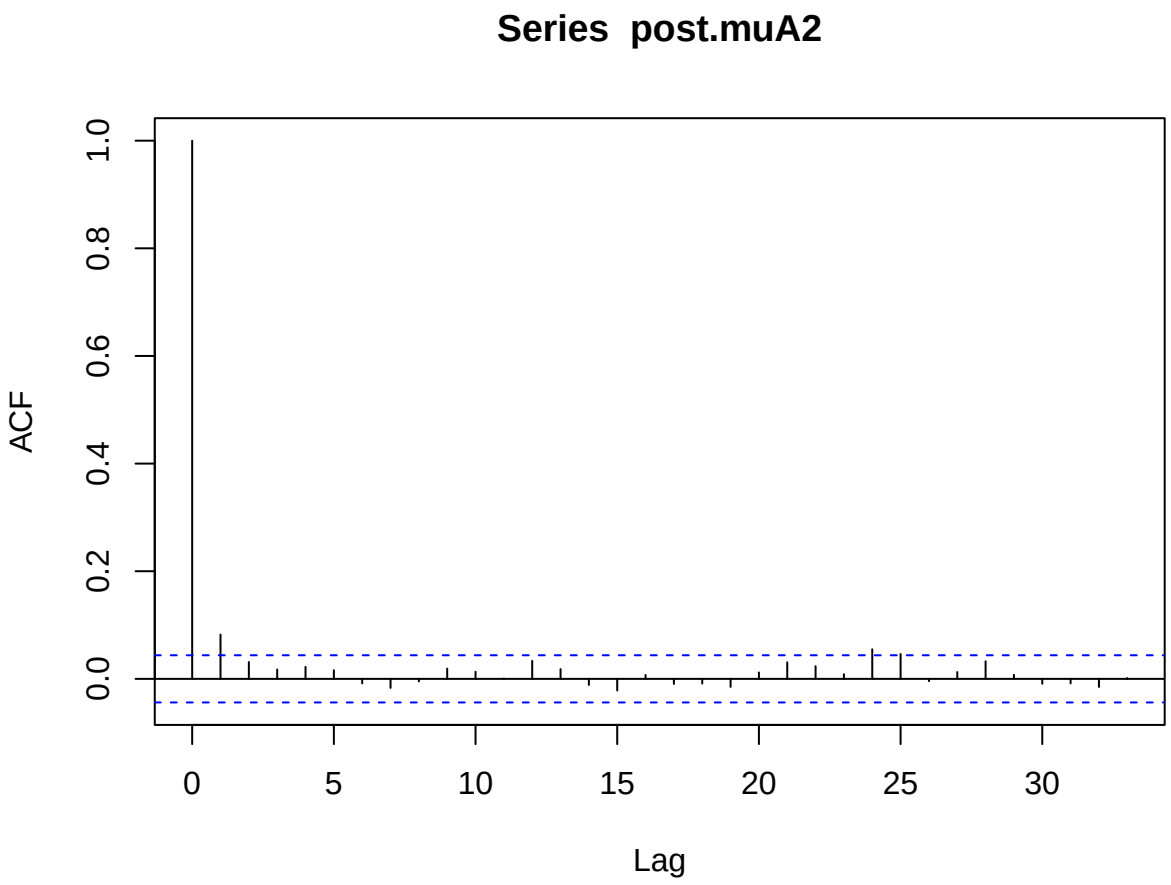
```
dim(posterior2)
```

```
## [1] 2000    4
```

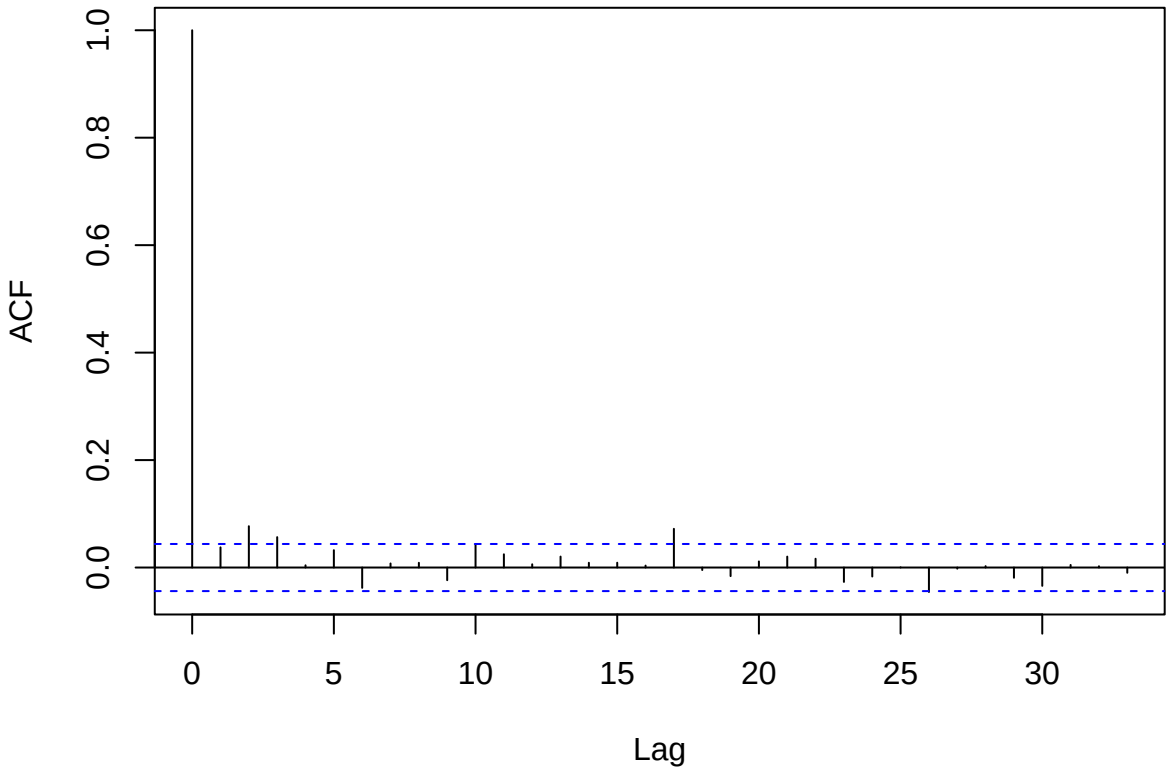
```
post.muA2 <- posterior2[,1]
post.muB2 <- posterior2[,2]
post.sigma2 <- posterior[,3]
```

From the autocorrelation plots below we can see that there is little to no significant autocorrelation in the series for μ_A and μ_B, which is a good diagnostic result for the MCMC sampler. Notice that the series for σ presents significant autocorrelation in the first lags, but it declines rapidly with increasing lags.

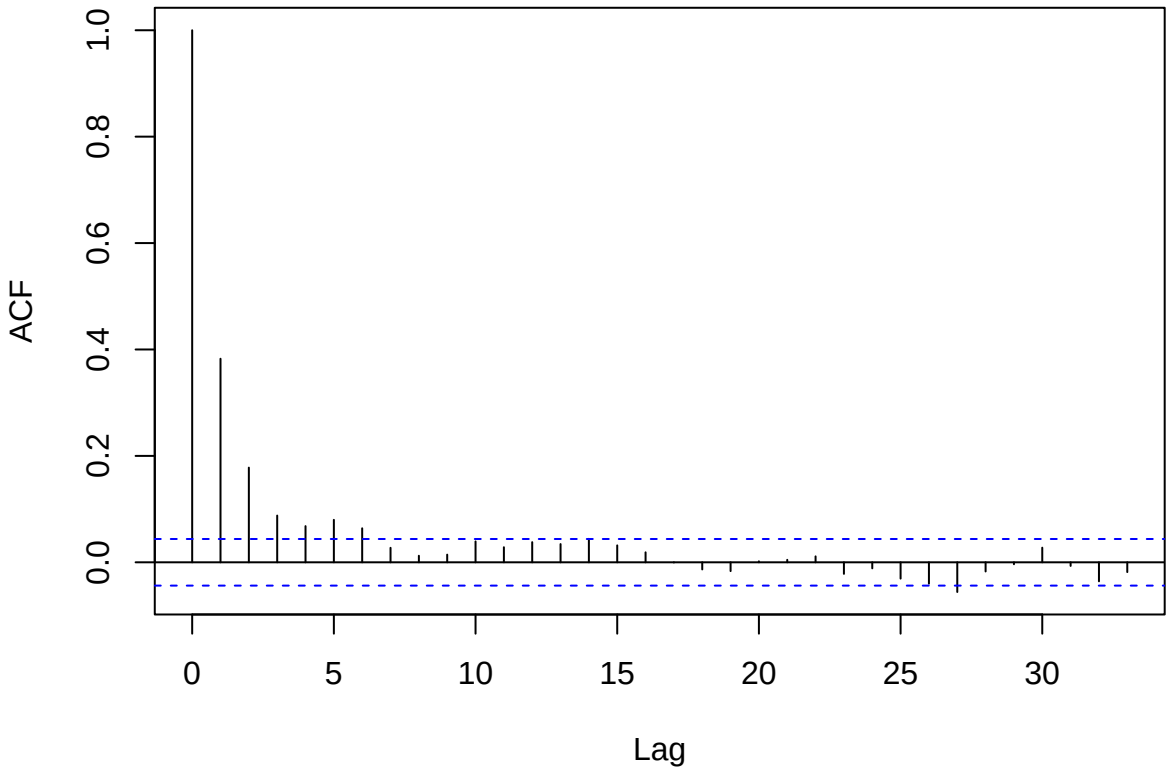
```
acf(post.muA2)
```



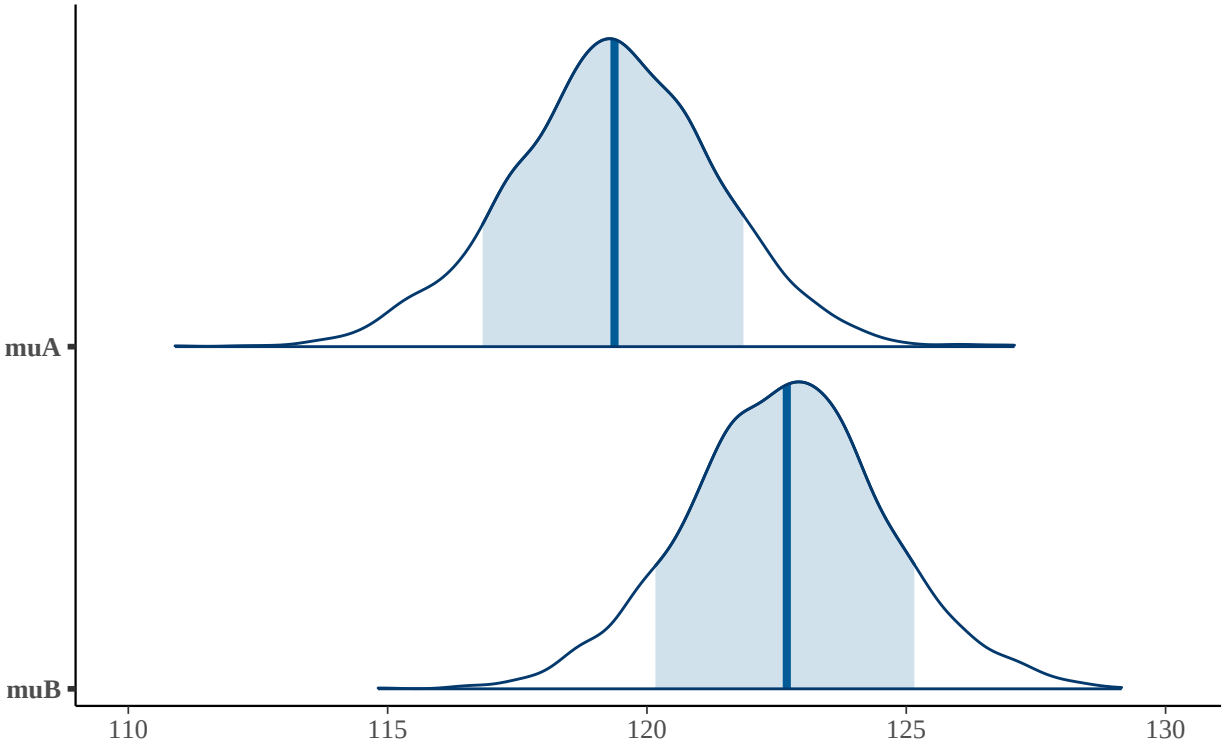

```
acf(post.muB2)
```




```
acf(post.sigma2)
```




```
plot_title <- ggtitle("Case 1: Posterior distributions of mu", "with medians and 80% intervals")
mcmc_areas(posterior2,
  pars = c("muA", "muB"),
  prob = 0.8) + plot_title
```

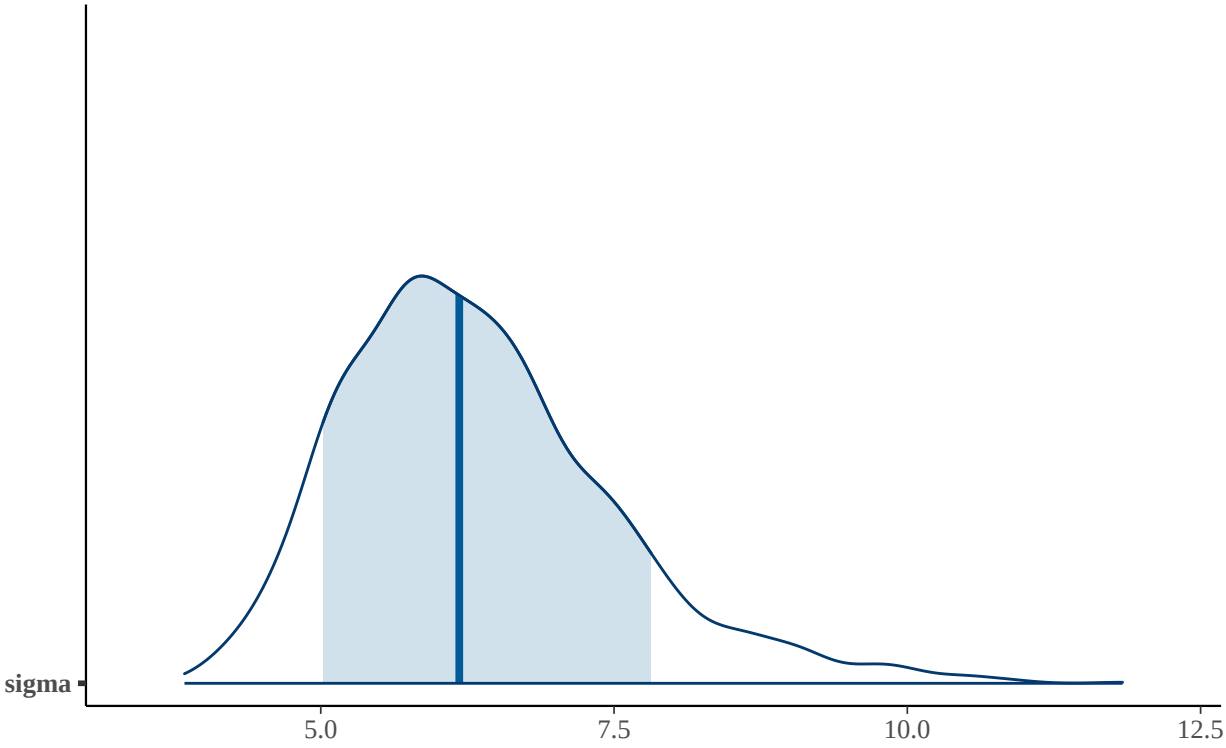


The posterior of σ is also plotted, for completeness, even though this is not essential to our task at hand.

```
plot_title <- ggtitle("Case 1: Posterior distribution of sigma", "with median and 80% interval")
mcmc_areas(posterior2,
  pars = c("sigma"),
  prob = 0.8) + plot_title
```

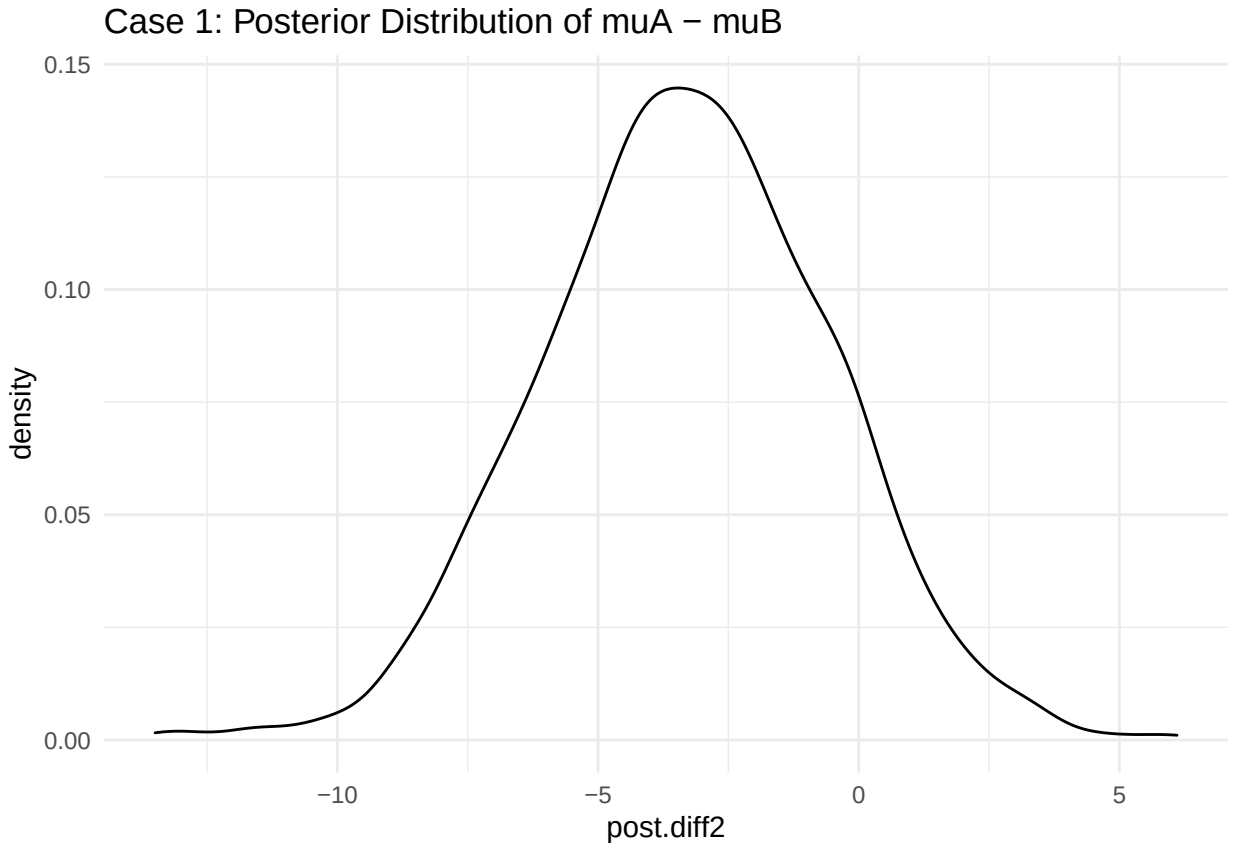
Case 1: Posterior distribution of sigma

with median and 80% interval



The posterior of $\mu_A - \mu_B$ is given below.

```
post.diff2 <- post.muA2 - post.muB2
ggplot(tibble(post.diff2)) +
  geom_density(aes(post.diff2)) +
  theme_minimal() +
  ggtitle("Case 1: Posterior Distribution of muA - muB")
```



A 95% quantile CI for $\mu_A - \mu_B$ is found below.

```
(CI.perc2 <- quantile(post.diff2, probs = c(0.025, 0.975)))
```

```
##      2.5%     97.5%
## -8.684915  1.974175
```

The Stan code in the file `4-4_training_workers3.stan` is given below for completeness. It is very similar to the code used earlier.

```
data{
  int<lower=0> nA;
  int<lower=0> nB;

  real<lower=0> tA[nA];
  real<lower=0> tB[nB];
}

parameters{
  real muA;
  real muB;
  real<lower=0> sigma;
}

model{
  tA ~ normal(muA, sigma);
  tB ~ normal(muB, sigma);

  muA ~ normal(100, 20);
```

```
  muB ~ normal(100, 20);
  sigma ~ normal(6,100);
}
```

Case 2)

Using the more informative Gaussian prior, the results are given in the following.

```
fit3 <- stan("4-4_training_workers3.stan", iter = 1000, chains = 4,
             data = data_list, seed = 1)
```

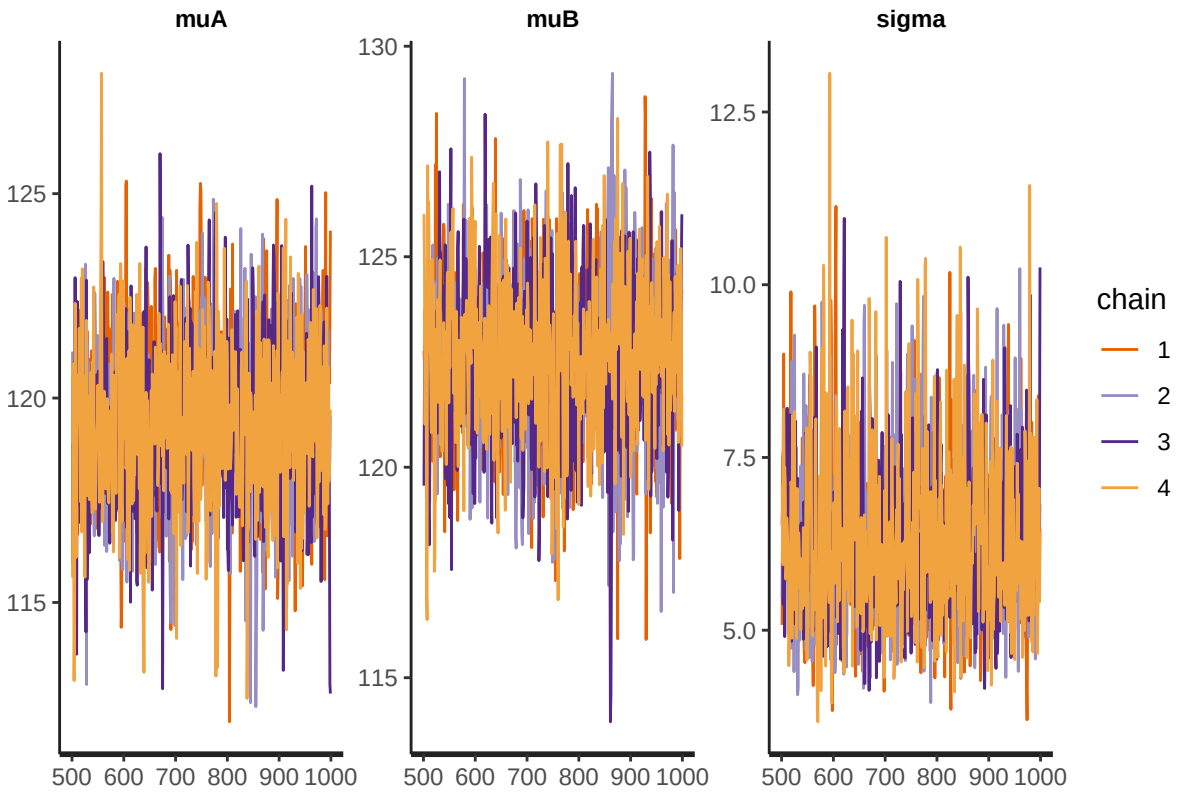
```
## Trying to compile a simple C file
```

The convergence analysis shows that the chains have converged.

```
print(fit3)
```

```
## Inference for Stan model: 4-4_training_workers3.
## 4 chains, each with iter=1000; warmup=500; thin=1;
## post-warmup draws per chain=500, total post-warmup draws=2000.
##
##         mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## muA   119.41    0.05 2.03 115.56 118.02 119.38 120.84 123.19  1747    1
## muB   122.64    0.05 1.95 118.80 121.37 122.71 123.93 126.21  1666    1
## sigma   6.33    0.03 1.17   4.46   5.52   6.17   7.01   9.05  1551    1
## lp__  -45.41    0.04 1.28 -48.75 -45.99 -45.07 -44.47 -43.92   828    1
##
## Samples were drawn using NUTS(diag_e) at Sun Apr 17 22:02:08 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
traceplot(fit3)
```




```
posterior3 <- as.data.frame(fit3)
head(posterior3)
```

```
##        muA      muB    sigma      lp__
## 1 120.4746 122.7607 5.076882 -44.28855
## 2 117.3510 122.4354 7.514575 -45.32296
## 3 115.7993 123.7562 7.609505 -46.28177
## 4 116.6465 123.0047 8.998260 -47.18480
## 5 121.0754 121.8027 5.646579 -44.34474
## 6 120.9411 121.6110 5.843328 -44.30687
```

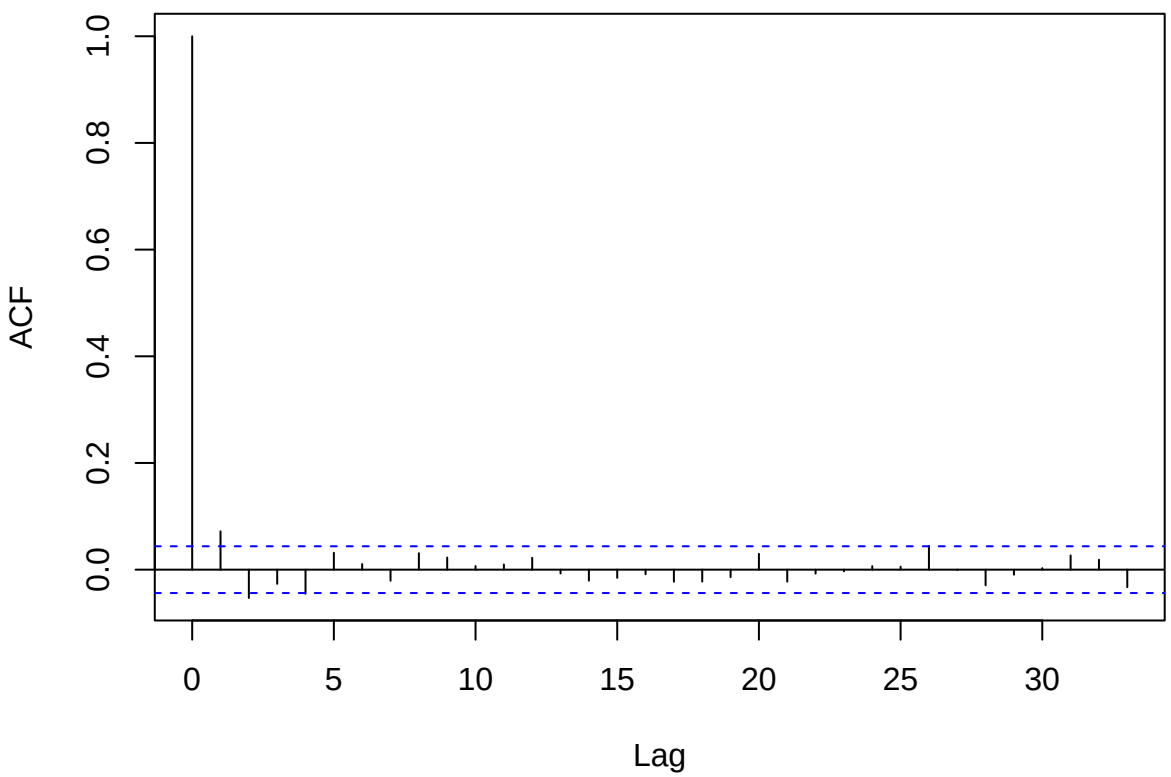
```
dim(posterior3)
```

```
## [1] 2000    4
```

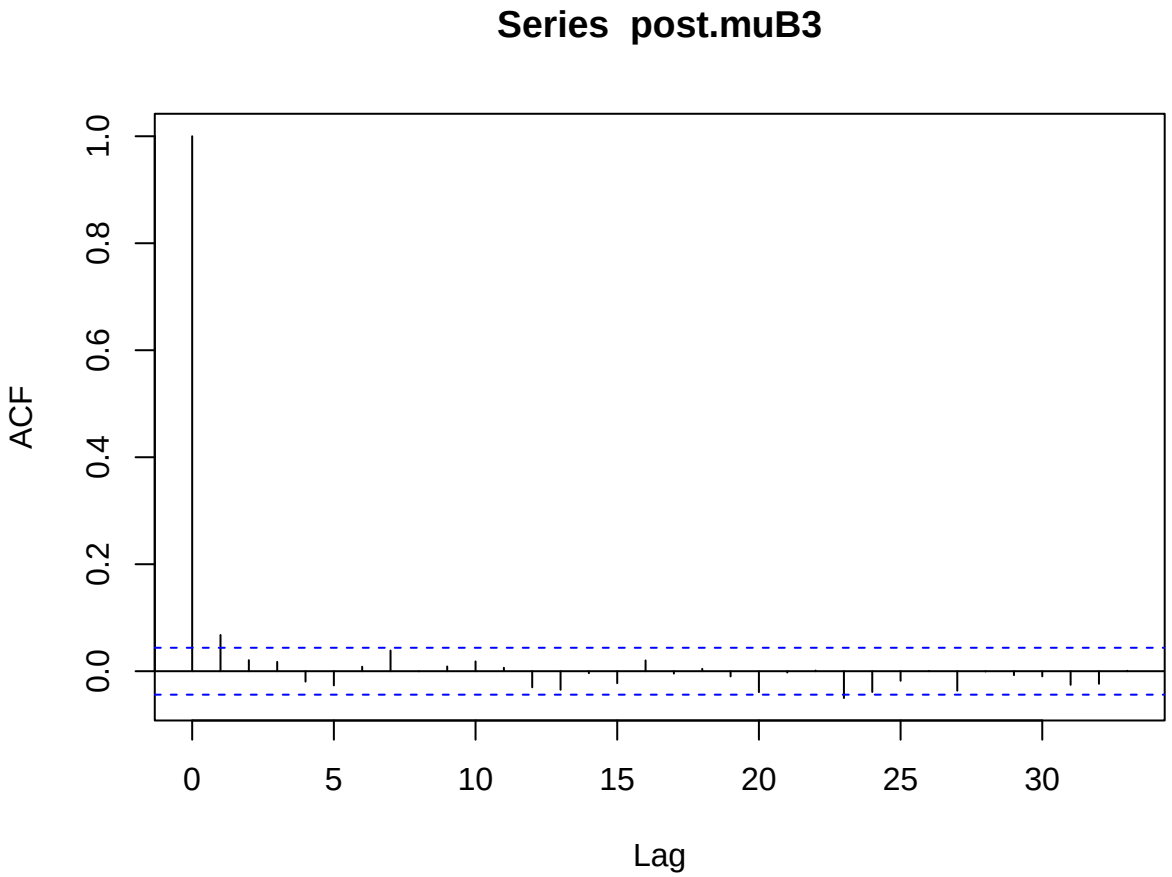
```
post.muA3 <- posterior3[,1]
post.muB3 <- posterior3[,2]
post.sigma3 <- posterior3[,3]
```

From the autocorrelation plots below we can see that there is little to no significant autocorrelation, which is a good diagnostic result for the MCMC sampler.

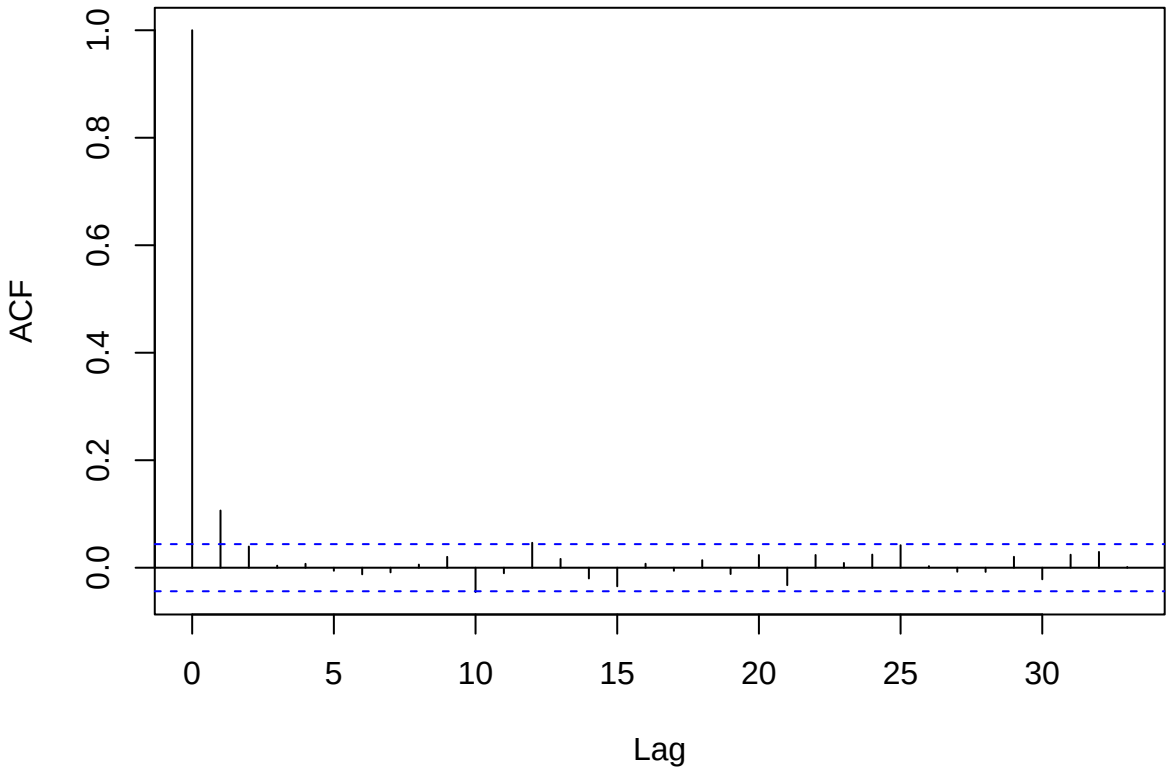
```
acf(post.muA3)
```



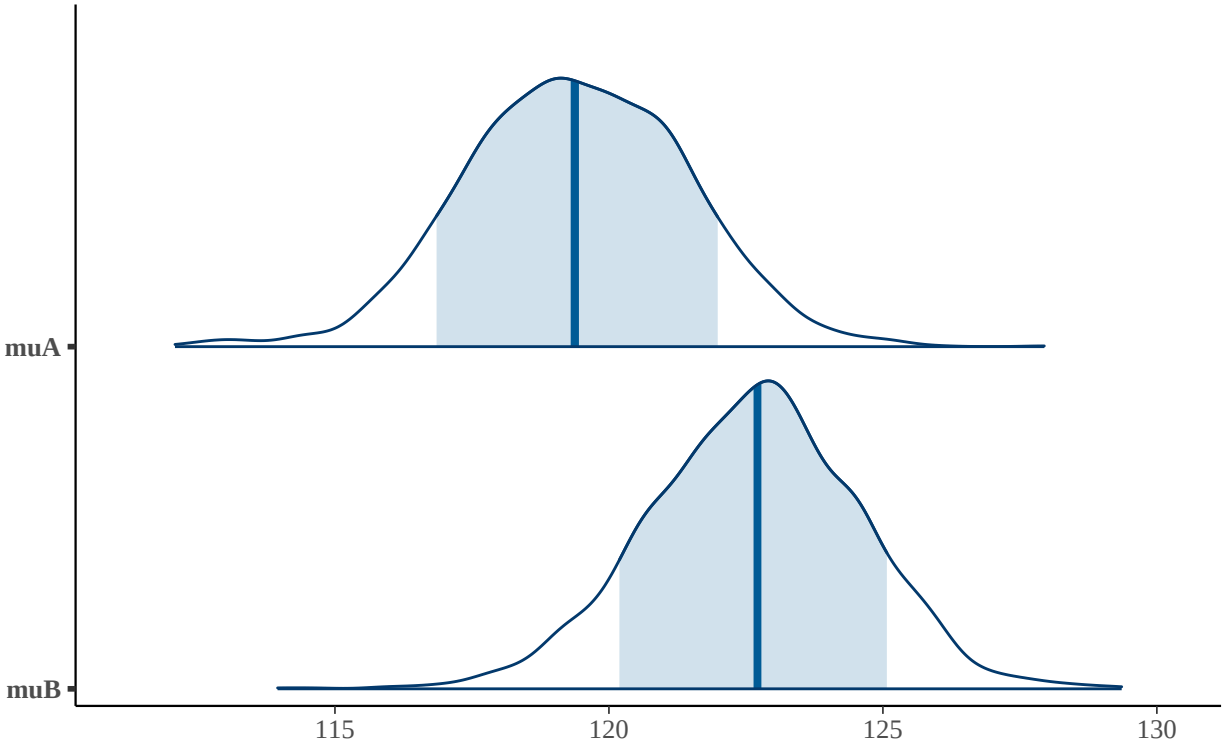

```
acf(post.muB3)
```




```
acf(post.sigma3)
```

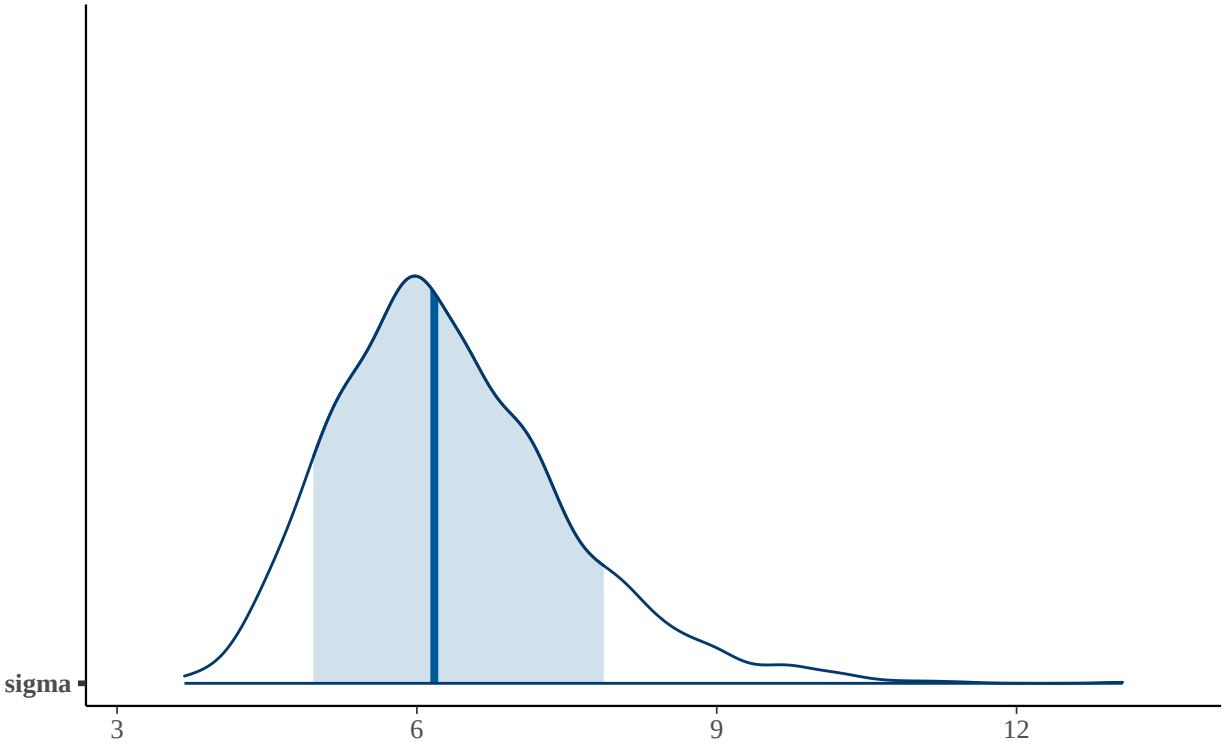



```
plot_title <- ggtitle("Case 2: Posterior distributions of mu", "with medians and 80% intervals")
mcmc_areas(posterior3,
  pars = c("muA", "muB"),
  prob = 0.8) + plot_title
```



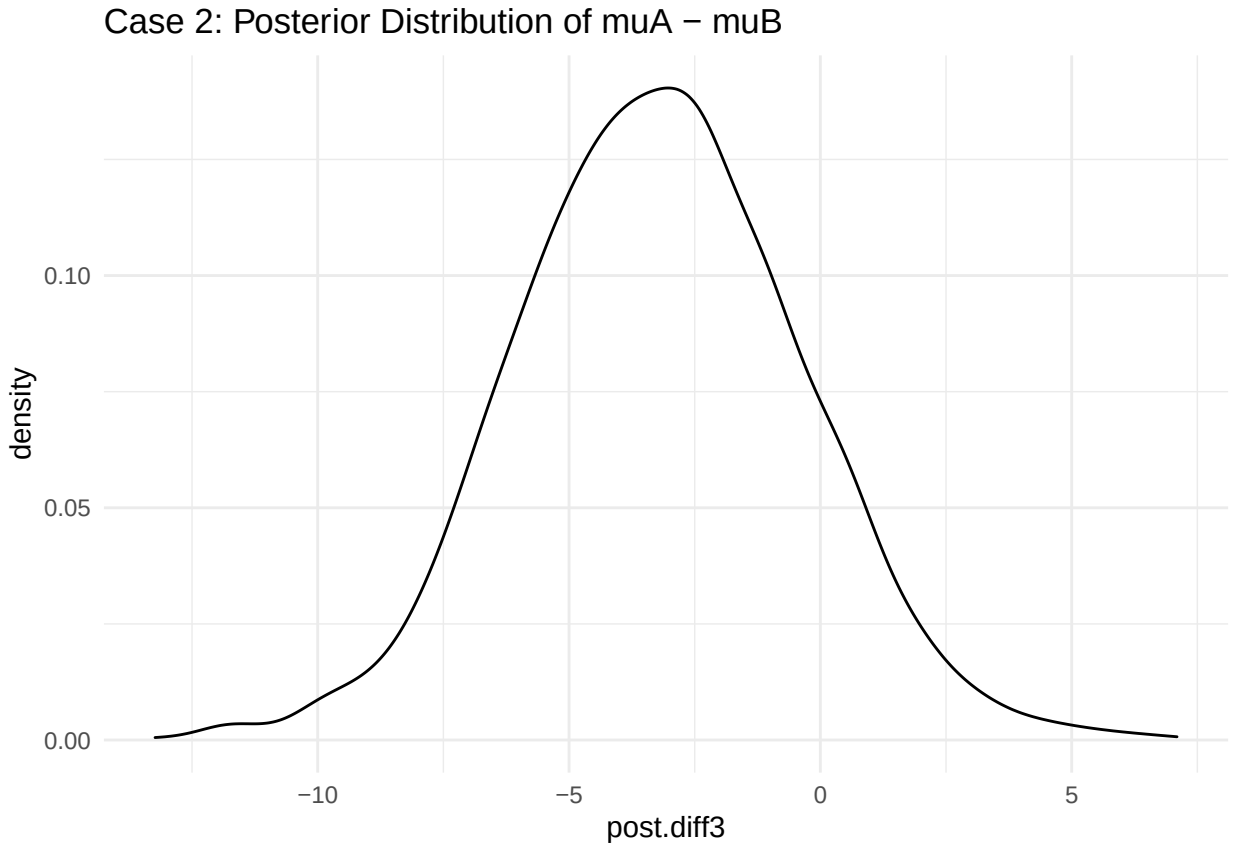
The posterior of σ is also plotted, for completeness, even though this is not essential to our task at hand.

```
plot_title <- ggtitle("Case 2: Posterior distribution of sigma", "with median and 80% interval")
mcmc_areas(posterior3,
  pars = c("sigma"),
  prob = 0.8) + plot_title
```



The posterior of $\mu_A - \mu_B$ is given below.

```
post.diff3 <- post.muA3 - post.muB3
ggplot(tibble(post.diff3)) +
  geom_density(aes(post.diff3)) +
  theme_minimal() +
  ggtitle("Case 2: Posterior Distribution of muA - muB")
```



A 95% quantile CI for $\mu_A - \mu_B$ is found below.

```
(CI.perc3 <- quantile(post.diff3, probs = c(0.025, 0.975)))
```

```
##      2.5%     97.5%
## -8.785599  2.284193
```

The Stan code in the file `4-4_training_workers3.stan` is given below for completeness. The only difference compared to in Case 1 is the change in the prior for sigma.

```
data{
  int<lower=0> nA;
  int<lower=0> nB;

  real<lower=0> tA[nA];
  real<lower=0> tB[nB];
}

parameters{
  real muA;
  real muB;
  real<lower=0> sigma;
}

model{
  tA ~ normal(muA, sigma);
  tB ~ normal(muB, sigma);

  muA ~ normal(100, 20);
```

```
  muB ~ normal(100, 20);
  sigma ~ normal(6,10);
}
```

All in all, we can see that the results are relatively similar in all three cases. Despite the fact that the σ is not precisely defined in either of the two cases in section **d)**, the credible intervals are similar, as seen below

```
df <- rbind(CI.perc, CI.perc2, CI.perc3)
rownames(df) <- c("Const. sigma", "Case 1", "Case 2")
knitr::kable(df)
```

	2.5%	97.5%
Const. sigma	-8.226049	1.763004
Case 1	-8.684915	1.974175
Case 2	-8.785599	2.284193

Moreover, the posterior distributions of μ_A, μ_B and $\mu_A - \mu_B$ look very similar in all three cases. Additionally, the posterior distribution of σ looks very similar for both the defined priors, despite the fact that the variance of the first prior is much larger than the variance of the second prior.

From the analysis we can conclude that, with 95% credibility, the means between the two groups are not different (since 0 is contained in the CI's for the difference of means in all three cases), i.e. that we with 95% credibility conclude that neither method A nor method B yields better speed performance among the workers.