

BDA Project

Arthur Aspelin, Jannica Savander, Christian Segercrantz

11/2021

Contents

1. Introduction	1
2. Description of the data	3
3. Description of the models	3
4. Priors	3
Convergence diagnostics	3
Predictive performance assessment (if applicable)	3
Posterior predictive checks	9
Model comparison with LOO-CV	14
Sensitivity analysis	19
Discussion	20
Conclusion	20
Self-reflection	20
References	20

1. Introduction

The motivation for this project is to estimate the parameters for blood pressure data with the help of Bayesian methods. High blood pressure corresponds with different diseases, such as diabetes and heart diseases. This means that it is essential to predict distribution of blood pressure and its parameters in an accurate way.

Solving the problem, firstly, we want to estimate what type of distribution can describe blood pressure. Then with different Bayesian models estimate the parameters for the distribution. We will also investigate how the parameters differ when dividing the data into different age groups. Higher blood pressure could correspond with higher age.

Main modeling idea is to test with different Bayesian models how to get accurate estimates of the parameters that could describe blood pressure.

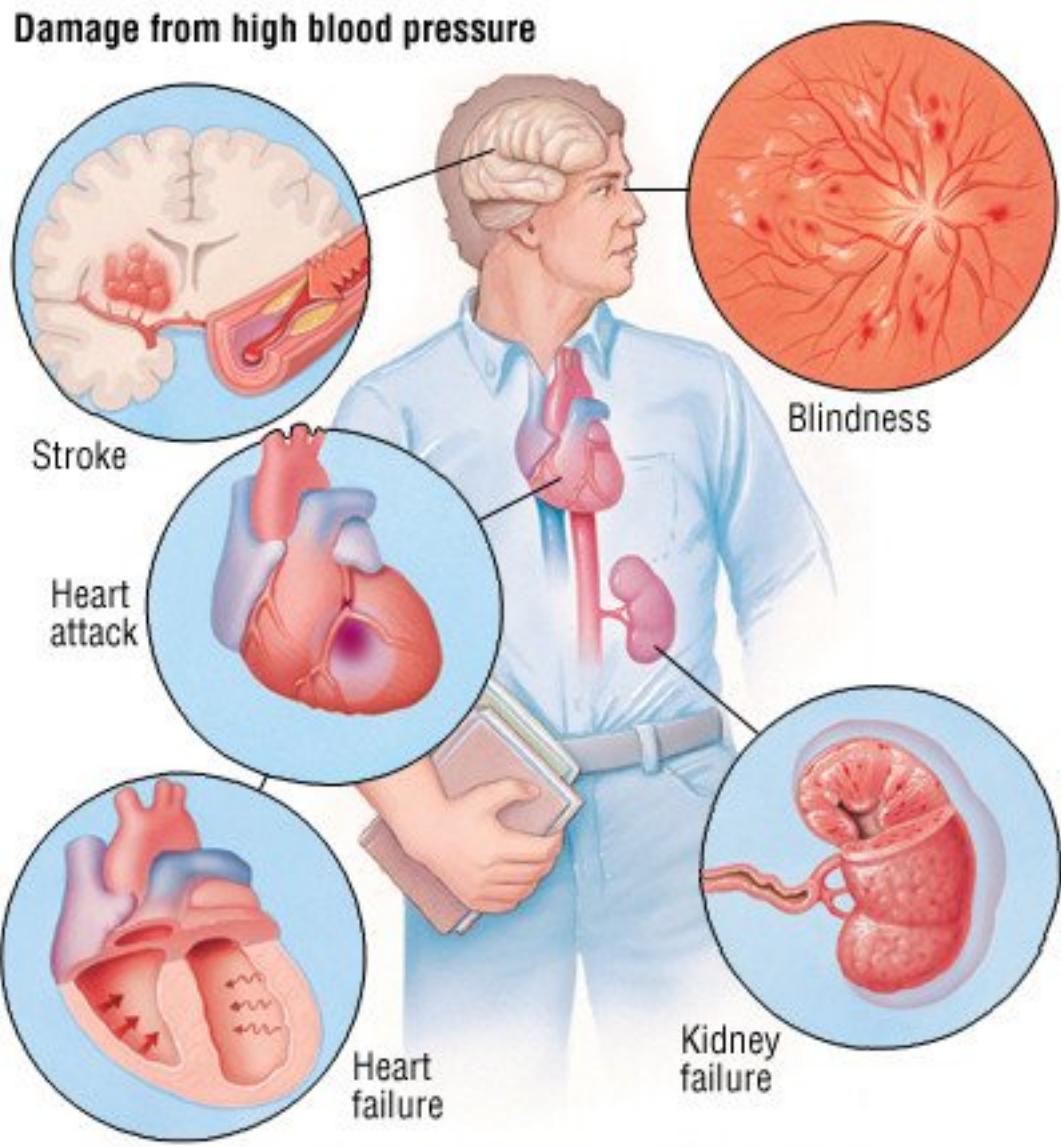


Figure 1: Damage that can be caused from high blood pressure

2. Description of the data

We used blood pressure data combined with age data from the [Diabetes Dataset from Kaggle](#). According to the data description, the dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases, and one data point corresponds to a female patient of Pima Indian heritage. All patients are at least 21 years old.

The dataset has more columns than we used, for example number of pregnancies, BMI and diabetes classification. We only used the columns BloodPressure, describing the diastolic blood pressure, and Age, describing the age of the patient in years. The diastolic blood pressure is the pressure the heart applies on the walls of the arteries between the beats (Mayo Clinic Staff (2021)). The unit for the diastolic blood pressure is mmHg, and a normal value is usually below 80. Higher values might indicate hypertension, which increases with age in Western countries (Gurven et al. (2012)).

3. Description of the models

4. Priors

We use weakly informative priors for parameters

$$\mu$$

and

$$\sigma$$

. They are chosen based on the blood pressure data that we get from the data set. From the plotted data we get weakly

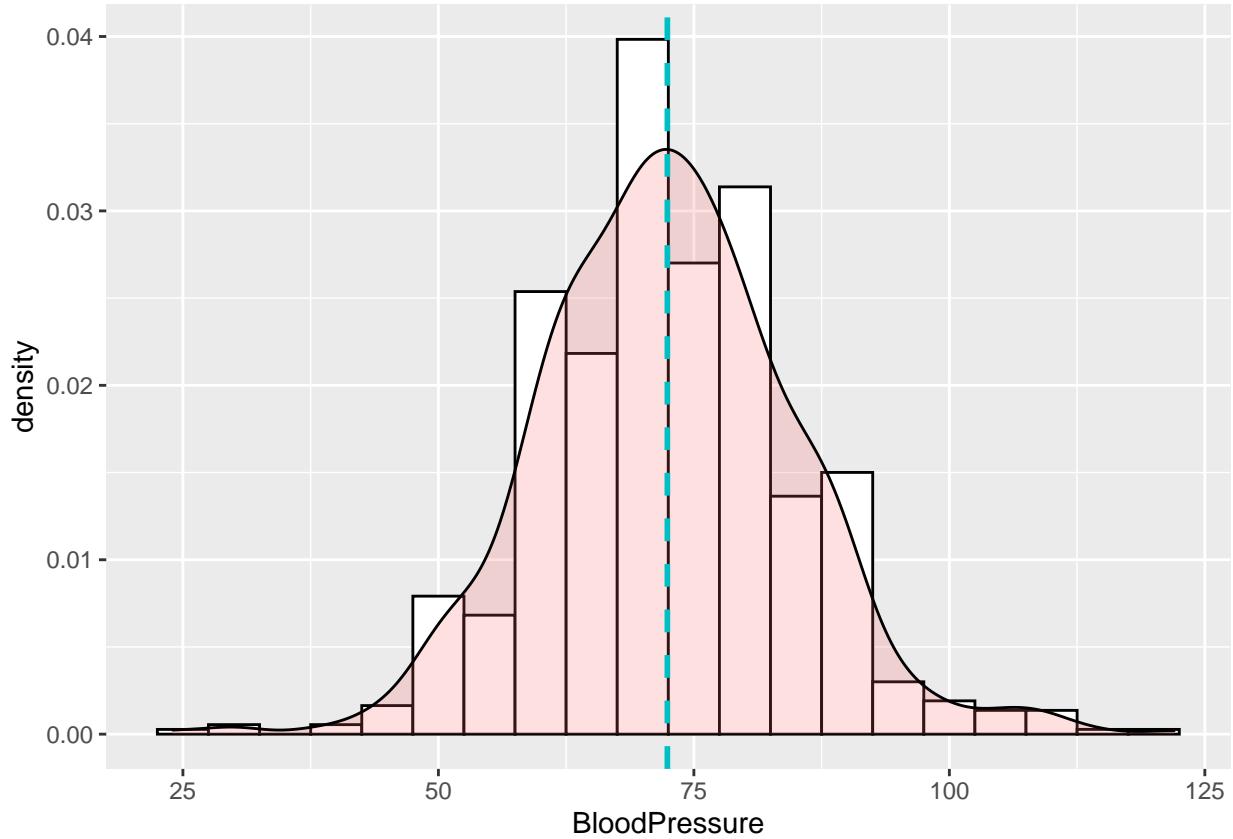
Convergence diagnostics

Predictive performance assessment (if applicable)

```
data <- data %>%
  filter(BloodPressure > 0) %>%
  select(BloodPressure, Age) %>%
  mutate(AgeGroup = case_when(
    Age <= 30      ~ "Young",
    Age > 30       ~ "Old")
  )
knitr::kable(head(data),
             caption = "The first rows of the dataset, with the additional 'AgeGroup' column.")
```

Table 1: The first rows of the dataset, with the additional ‘AgeGroup’ column.

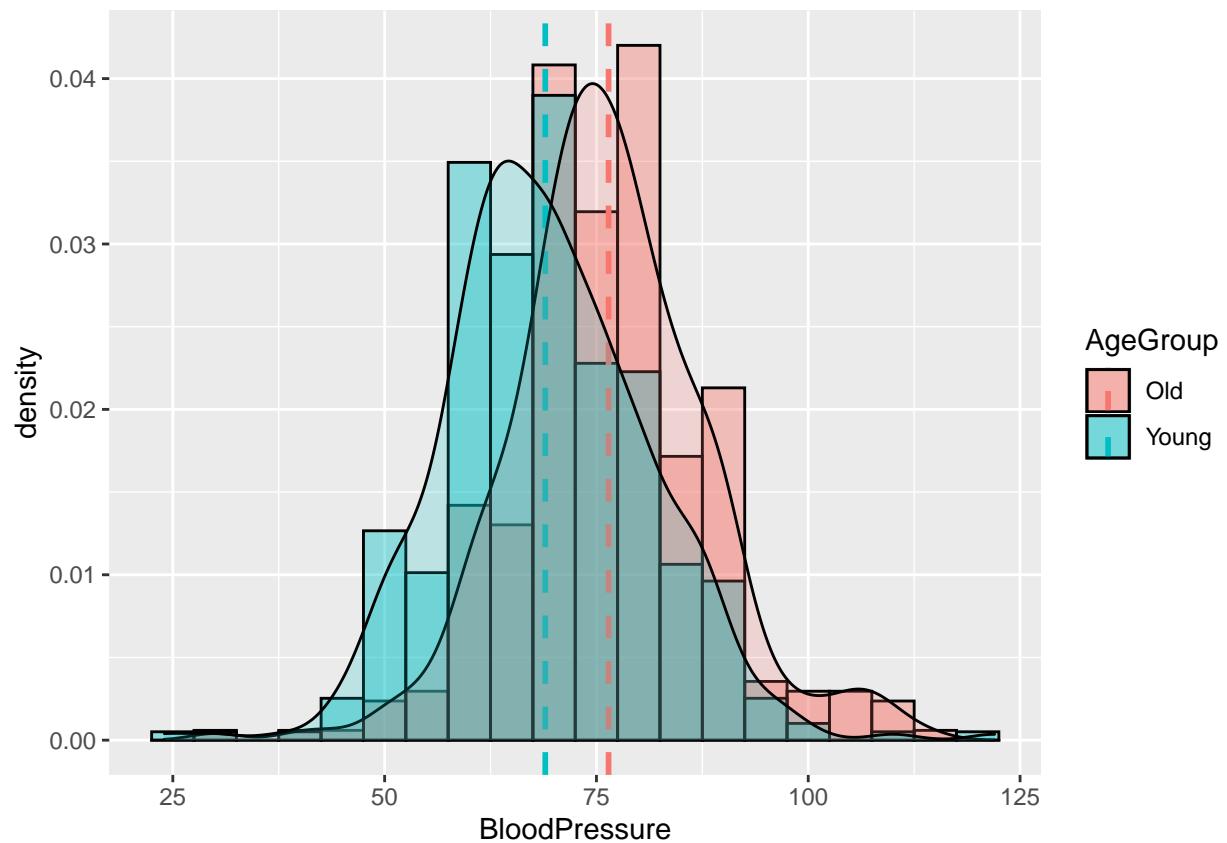
BloodPressure	Age	AgeGroup
72	50	Old
66	31	Old
64	32	Old
66	21	Young
40	33	Old
74	30	Young

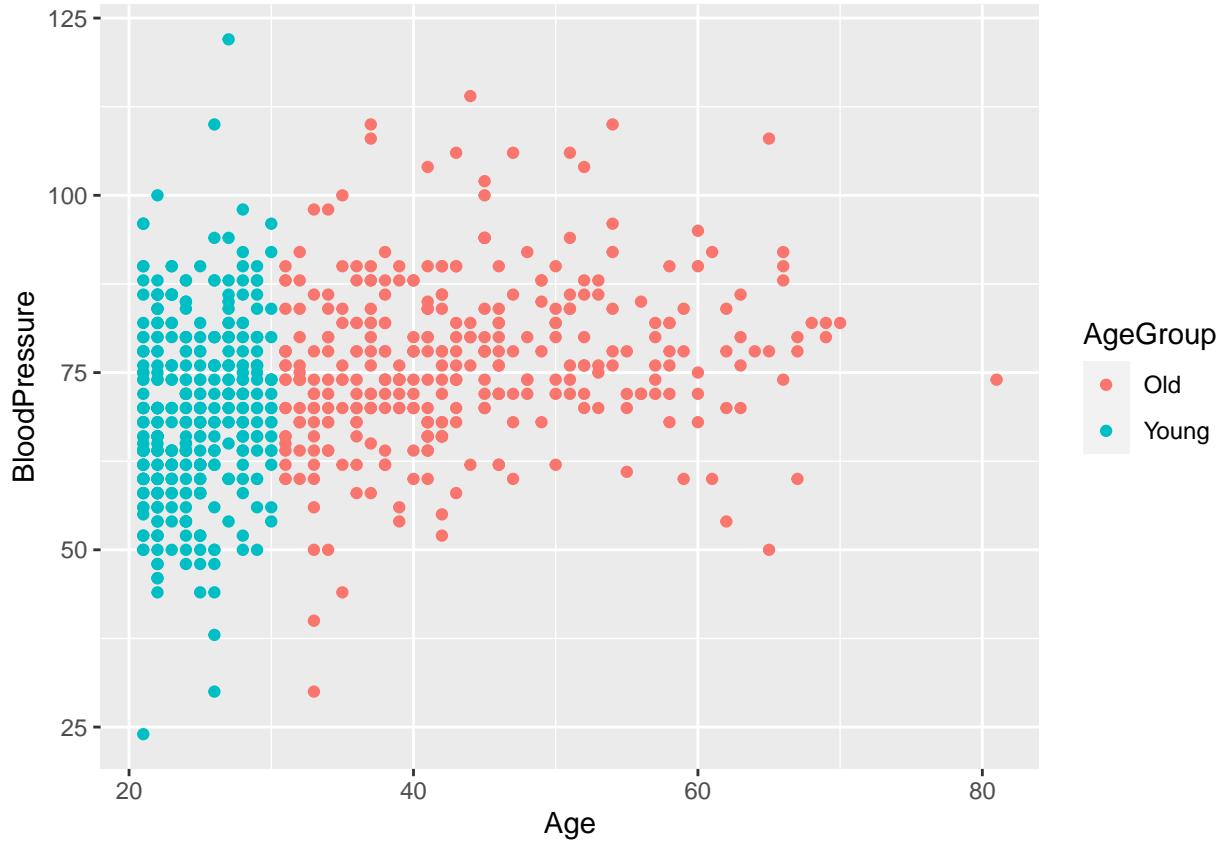


```

means <- data %>%
  group_by(AgeGroup) %>%
  summarise(mean = mean(BloodPressure), n = n())

ggplot(data, aes(x=BloodPressure, fill=AgeGroup)) +
  geom_histogram(aes(y=..density..), binwidth = 5, colour="black", position = "identity", alpha = 0.4) +
  geom_vline(data = means, aes(xintercept=mean, color = AgeGroup), linetype="dashed", size=1) +
  geom_density(alpha=.2)
  
```





```
# Stan code

data {
    int<lower=0> N;                                //Amount of data points
    vector[N] y;                                    //Data points
    real mean_mu_prior;                            //Expected value of the mean prior
    real<lower=0> mean_sigma_prior;                //variance of the mean prior
    real<lower=0> var_prior;                        //Variance of the variance prior
}

parameters {
    real mu;
    real<lower=0> sigma;
}

model {
    //prior
    mu ~ normal(mean_mu_prior, mean_sigma_prior);
    sigma ~ inv_chi_square(var_prior);
    //likelihoods
    y ~ normal(mu, sigma);
}

generated quantities {
    real ypred;
    vector[N] log_lik;
    ypred = normal_rng(mu, sigma);
```

```

for (n in 1:(N)){
  log_lik[n] = normal_lpdf(y[n] | mu, sigma);
}
}

data_old <- data %>%
  filter(AgeGroup == "Old")

mean_mu_prior_old = 75
mean_sigma_prior_old = 10
var_prior_old = 20
data_nonhiera_old <- list(
  y = data_old$BloodPressure,
  N = length(data_old$BloodPressure),
  mean_mu_prior = mean_mu_prior_old,
  mean_sigma_prior = mean_sigma_prior_old,
  var_prior = var_prior_old
)

fit_nonhiera_old = sampling(nonhieramodel,
  data = data_nonhiera_old,           # named list of data
  chains = 4,                      # number of Markov chains
  warmup = 1000,                   # number of warmup iterations per chain
  iter = 2000,                     # total number of iterations per chain
  cores = 4,                       # number of cores (could use one per chain)
  refresh = 0                      # no progress shown
)

data_young <- data %>%
  filter(AgeGroup == "Young")

mean_mu_prior_young = 65
mean_sigma_prior_young = 10
var_prior_young = 20
data_nonhiera_young <- list(
  y = data_young$BloodPressure,
  N = length(data_young$BloodPressure),
  mean_mu_prior = mean_mu_prior_young,
  mean_sigma_prior = mean_sigma_prior_young,
  var_prior = var_prior_young
)

fit_nonhiera_young = sampling(nonhieramodel,
  data = data_nonhiera_young,         # named list of data
  chains = 4,                      # number of Markov chains
  warmup = 1000,                   # number of warmup iterations per chain
  iter = 2000,                     # total number of iterations per chain
  cores = 4,                       # number of cores (could use one per chain)
  refresh = 0                      # no progress shown
)

data {
  int<lower=0> N;                  //Amount of data points
  vector[N] y;                      //
  real mean_mu_prior;               //
  real<lower=0> mean_sigma_prior;   //

```

```

    real<lower=0> var_prior;           // 
}

parameters {
    real mu;
    real<lower=0> sigma;
    real mu_hypo;
    real<lower=0> tau;
}

model {
    //hyperpriors
    mu_hypo ~ normal(mean_mu_prior, mean_sigma_prior);
    tau ~ inv_chi_square(var_prior);
    //prior
    mu ~ normal(mu_hypo, tau);
    sigma ~ inv_chi_square(var_prior);
    //likelihoods
    y ~ normal(mu, sigma);
}

generated quantities {
    real ypred;
    vector[N] log_lik;
    ypred = normal_rng(mu, sigma);
    for (n in 1:(N)){
        log_lik[n] = normal_lpdf(y[n] | mu, sigma);
    }
}

mean_mu_prior = 70
mean_sigma_prior = 10
var_prior = 20
data_hiera_old <- list(
    y = data_old$BloodPressure,
    N = length(data_old$BloodPressure),
    mean_mu_prior = mean_mu_prior,
    mean_sigma_prior = mean_sigma_prior_old,
    var_prior = var_prior
)
data_hiera_young <- list(
    y = data_young$BloodPressure,
    N = length(data_young$BloodPressure),
    mean_mu_prior = mean_mu_prior,
    mean_sigma_prior = mean_sigma_prior,
    var_prior = var_prior
)

fit_hiera_old = sampling(hieramodel,
    data = data_hiera_old,          # named list of data
    chains = 4,                     # number of Markov chains
    warmup = 1000,                  # number of warmup iterations per chain
    iter = 2000,                    # total number of iterations per chain
    cores = 4,                      # number of cores (could use one per chain)
    refresh = 0                     # no progress shown
)

```

```

)
fit_hiera_young = sampling(hieramodel,
  data = data_hiera_young,                      # named list of data
  chains = 4,                                     # number of Markov chains
  warmup = 1000,                                    # number of warmup iterations per chain
  iter = 2000,                                     # total number of iterations per chain
  cores = 4,                                       # number of cores (could use one per chain)
  refresh = 0                                      # no progress shown
)

```

Posterior predictive checks

```

knitr::kable(head(monitor(fit_nonhiera_old, print = FALSE), 3),
  caption = "The diagnostics of the nonhierarcical model for the old age group.")

```

Table 2: The diagnostics of the nonhierarcical model for the old age group.

	meane_mean	2.5%25%50%75%97.5%	%_eff	Rhavalid	Q5	Q50	Q95	MCSEM	CSEM	CSEM	CSEM	CSEM	CSEM	Q95	B5	SDESS	ESS
mu	76.40.0100.62175.276.076.476.877.7	3743	1	1	75.476.477.40.034	0.016	0.012	0.013	0.026	0.0073770	2652						
sigma	1.40.0070.42310.711.111.411.712.3	3819	1	1	10.811.412.10.017	0.011	0.008	0.012	0.022	0.0053860	3044						
ypred	6.40.18311.4151.568.976.484.298.8	3888	1	1	57.476.495.10.361	0.283	0.214	0.231	0.534	0.1293895	3798						

```

knitr::kable(head(monitor(fit_nonhiera_young, print = FALSE), 3),
  caption = "The diagnostics of the nonhierarcical model for the young age group.")

```

Table 3: The diagnostics of the nonhierarcical model for the young age group.

	meane_mean	2.5%25%50%75%97.5%	%_eff	Rhavalid	Q5	Q50	Q95	MCSEM	CSEM	CSEM	CSEM	CSEM	CSEM	Q95	B5	SDESS	ESS
mu	68.90.0110.6057.768.568.969.370.1	3054	1	1	68.068.969.90.037	0.015	0.013	0.012	0.041	0.0083114	2274						
sigma	1.90.0080.41211.111.611.912.212.7	2882	1	1	11.311.912.60.017	0.009	0.007	0.012	0.018	0.0052954	2280						
ypred	8.80.19311.955.060.968.876.991.5	3792	1	1	49.168.888.20.588	0.206	0.297	0.255	0.504	0.1373824	3604						

```

knitr::kable(head(monitor(fit_hiera_old, print = FALSE), 3),
  caption = "The diagnostics of the hierarcical model for the old age group.")

```

Table 4: The diagnostics of the hierarcical model for the old age group.

	meane_mean	2.5%25%50%75%97.5%	%_eff	Rhavalid	Q5	Q50	Q95	MCSEM	CSEM	CSEM	CSEM	CSEM	CSEM	Q95	B5	SDESS	ESS
mu	76.40.0150.61775.276.076.476.877.6	1523	1	1	75.476.477.40.042	0.017	0.016	0.019	0.045	0.0111612	1791						
sigma	1.40.0090.42810.611.111.411.712.3	2158	1	1	10.711.412.10.017	0.014	0.009	0.012	0.030	0.0062189	2198						
mu_hyp	40.0150.62075.276.076.476.877.6	1561	1	1	75.476.477.40.043	0.020	0.014	0.018	0.053	0.0111639	1791						

```

knitr::kable(head(monitor(fit_hiera_young, print = FALSE), 3),
  caption = "The diagnostics of the hierarcical model for the young age group.")

```

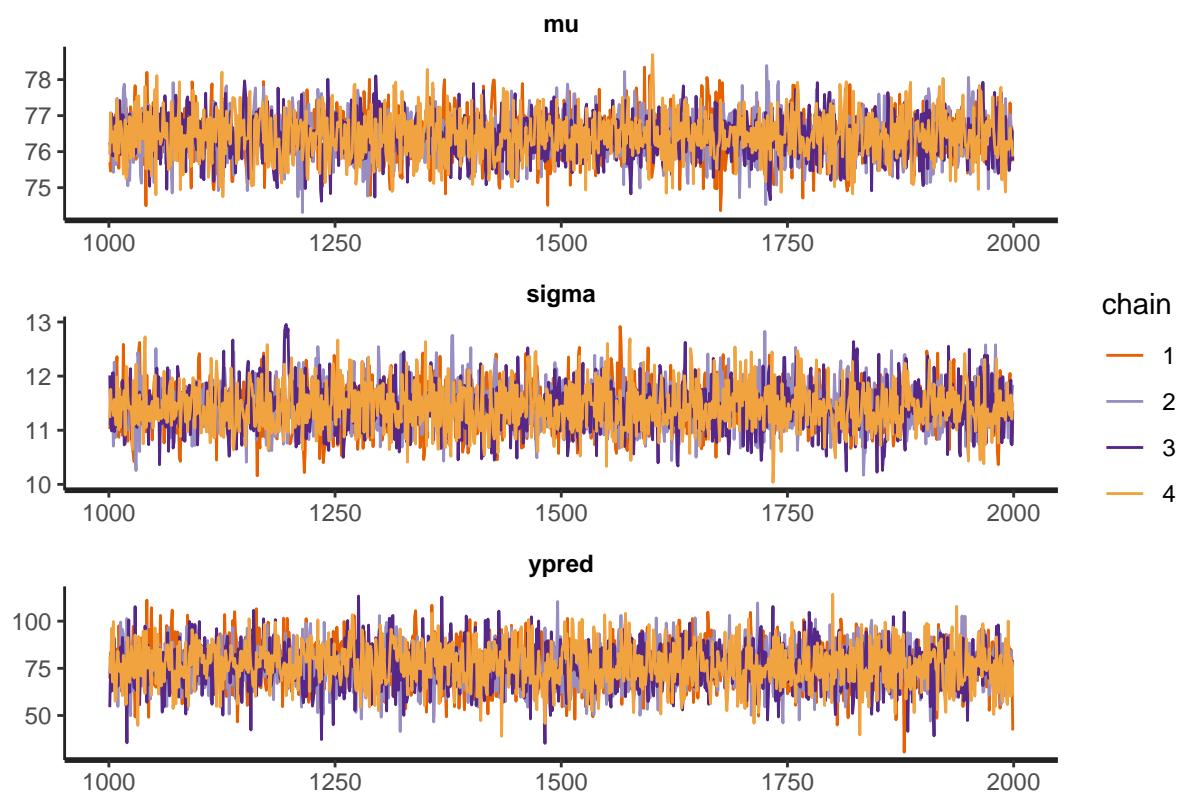


Figure 2: The trace plot of the nonhierarcical model for the old age group.

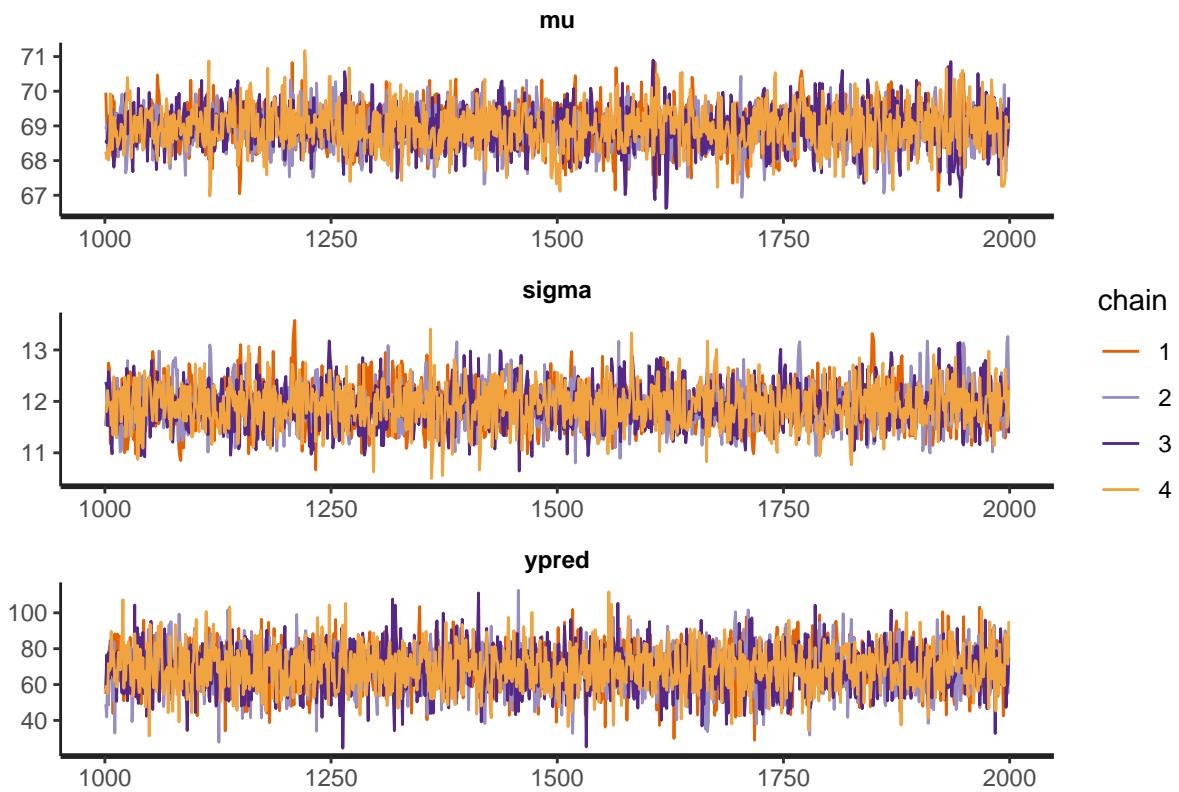


Figure 3: The trace plot of the nonhierarcical model for the young age group.

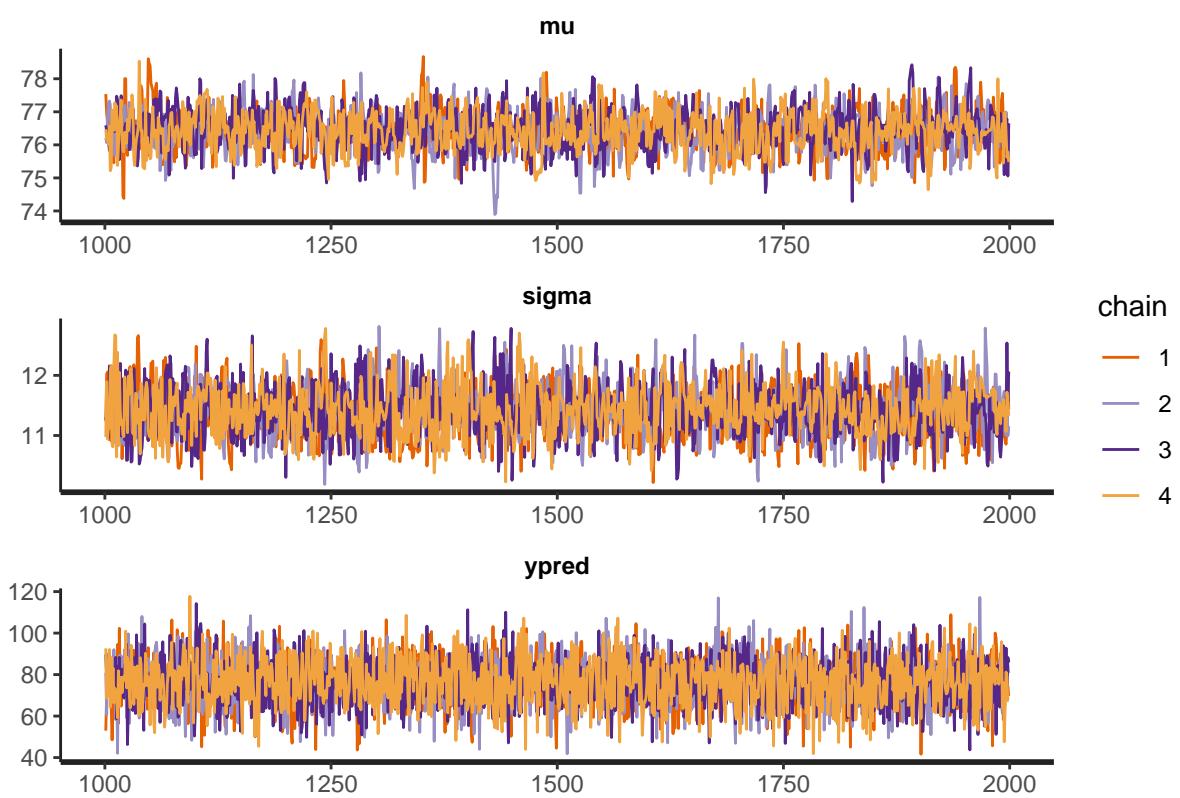


Figure 4: The trace plot of the hierarcical model for the old age group.

Table 5: The diagnostics of the hierarchical model for the young age group.

	mean	e_mean	2.5%	25%	50%	75%	97.5%	efRhat	validQ5	Q50	Q95	MCSE	MCSE	MCSE	MCSE	MCSE	MCSE	MCSE	MCSE	MCSE	ESS
mu	69.00	0.160	60.60	67.868	66.9	69.4	70.1	1387	1	68.069	69.90	0.034	0.024	0.015	0.020	0.043	0.011	1399	1721		
sigma	1.90	0.0090	0.413	1.111	6.11	9.12	212.7	2242	1	11.211	9.12	6.020	0.010	0.009	0.009	0.032	0.006	2268	1844		
mu_169	69.00	0.160	61.067	68.868	66.9	69.4	70.1	1388	1	67.969	69.90	0.024	0.021	0.015	0.018	0.046	0.012	1400	1833		

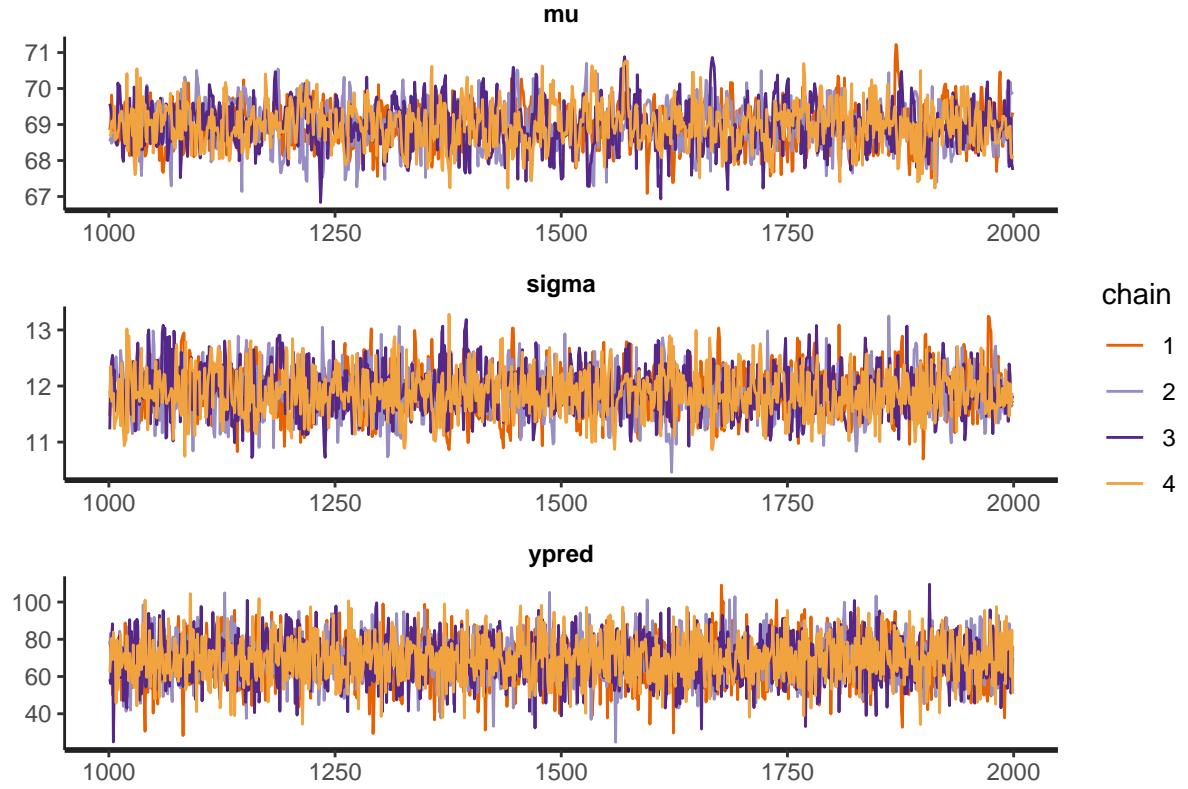
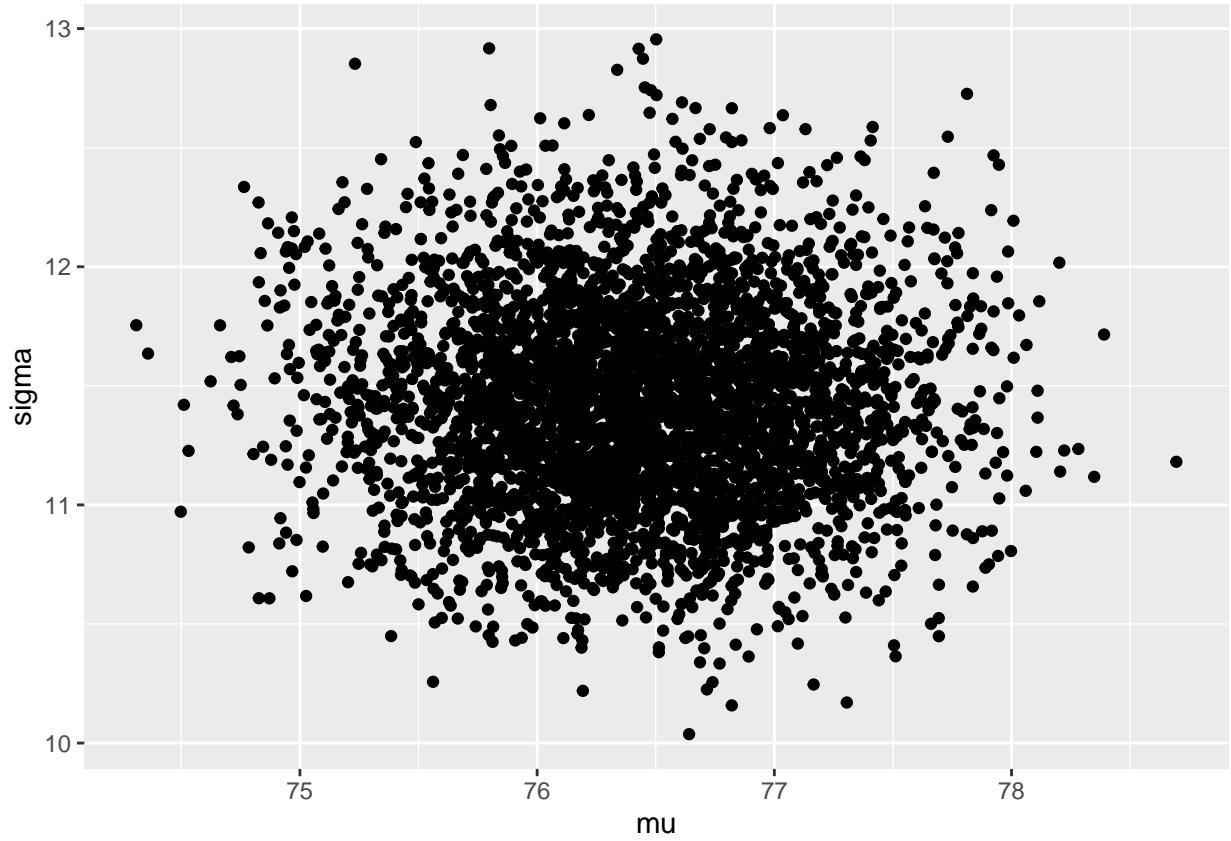


Figure 5: The trace plot of the hierarchical model for the young age group.



Model comparison with LOO-CV

```

loo_nonhiera_old <- loo(fit_nonhiera_old, pars="log_lik")
loo_nonhiera_old

##
## Computed from 4000 by 338 log-likelihood matrix
##
##           Estimate    SE
## elpd_loo   -1309.0 17.4
## p_loo       2.6   0.5
## looic      2618.0 34.8
## -----
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.

loo_nonhiera_young <- loo(fit_nonhiera_young, pars="log_lik")
loo_nonhiera_young

##
## Computed from 4000 by 395 log-likelihood matrix
##
##           Estimate    SE
## elpd_loo   -1544.6 18.1

```

```

## p_loo      2.5  0.6
## looic     3089.2 36.3
## -----
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
loo_hiera_old <- loo(fit_nonhiera_old, pars="log_lik")
loo_hiera_old

##
## Computed from 4000 by 338 log-likelihood matrix
##
##           Estimate    SE
## elpd_loo   -1309.0 17.4
## p_loo       2.6  0.5
## looic      2618.0 34.8
## -----
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
loo_hiera_young <- loo(fit_nonhiera_young, pars="log_lik")
loo_hiera_young

##
## Computed from 4000 by 395 log-likelihood matrix
##
##           Estimate    SE
## elpd_loo   -1544.6 18.1
## p_loo       2.5  0.6
## looic      3089.2 36.3
## -----
## Monte Carlo SE of elpd_loo is 0.0.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
print("The model for the old:")

## [1] "The model for the old:"
loo_compare(loo_nonhiera_old, loo_hiera_old)

##           elpd_diff se_diff
## model1  0.0      0.0
## model2  0.0      0.0

print("The model for the young:")

## [1] "The model for the young:"
loo_compare(loo_nonhiera_young, loo_hiera_young)

##           elpd_diff se_diff
## model1  0.0      0.0

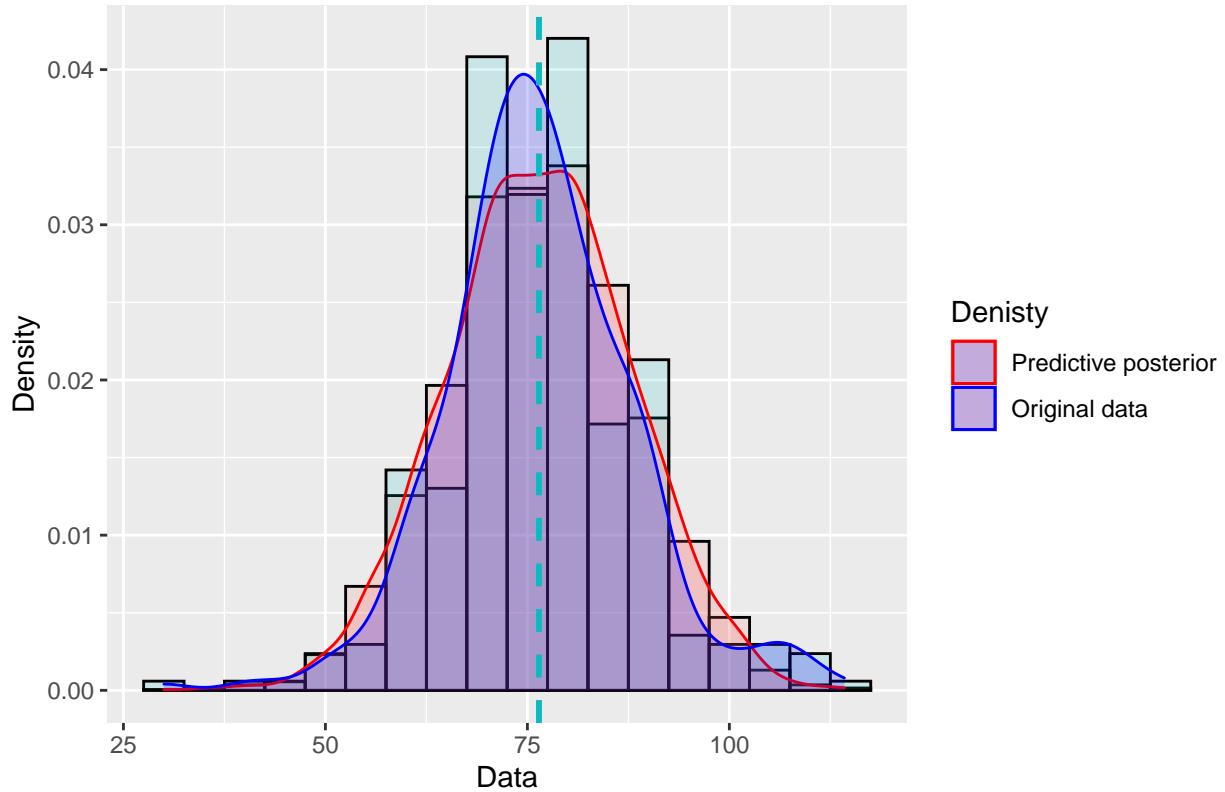
```

```

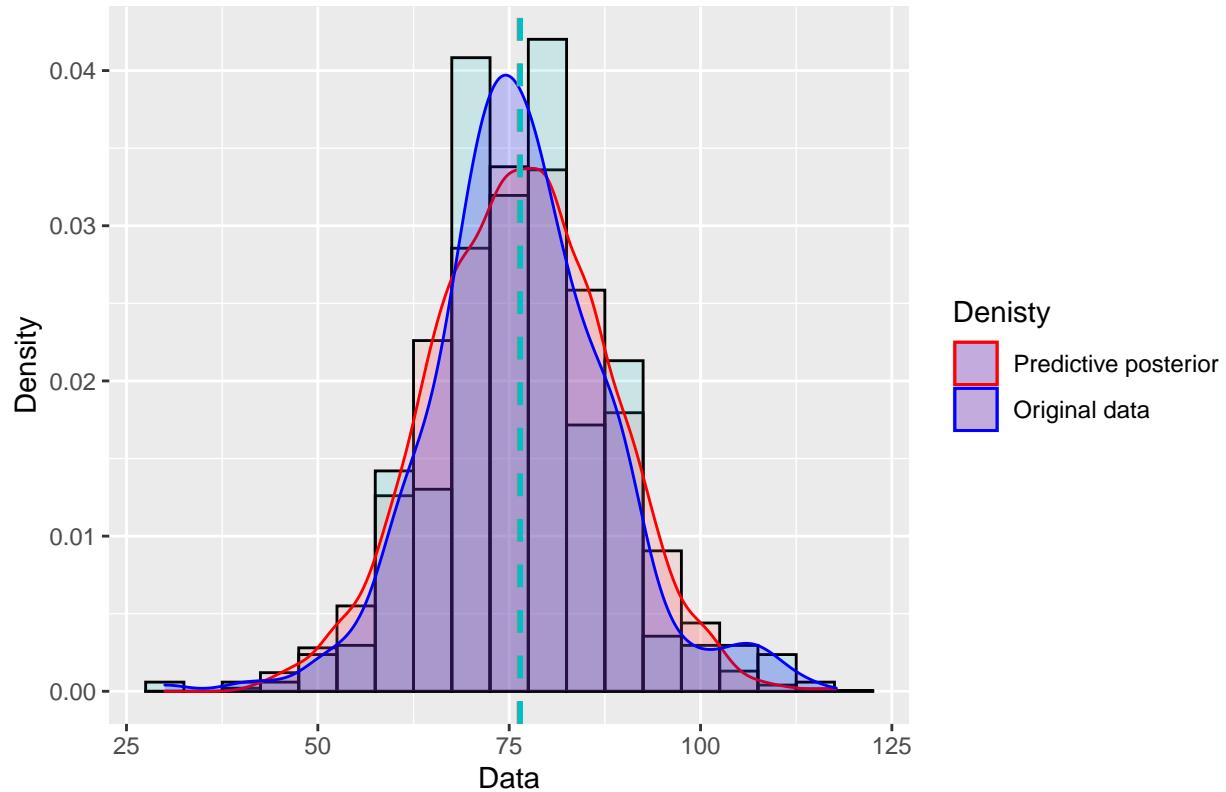
## model2 0.0      0.0
extract_hiera_old <- data.frame(extract(fit_hiera_old))
extract_nonhiera_old <- data.frame(extract(fit_nonhiera_old))
extract_hiera_young <- data.frame(extract(fit_hiera_young))
extract_nonhiera_young <- data.frame(extract(fit_nonhiera_young))

```

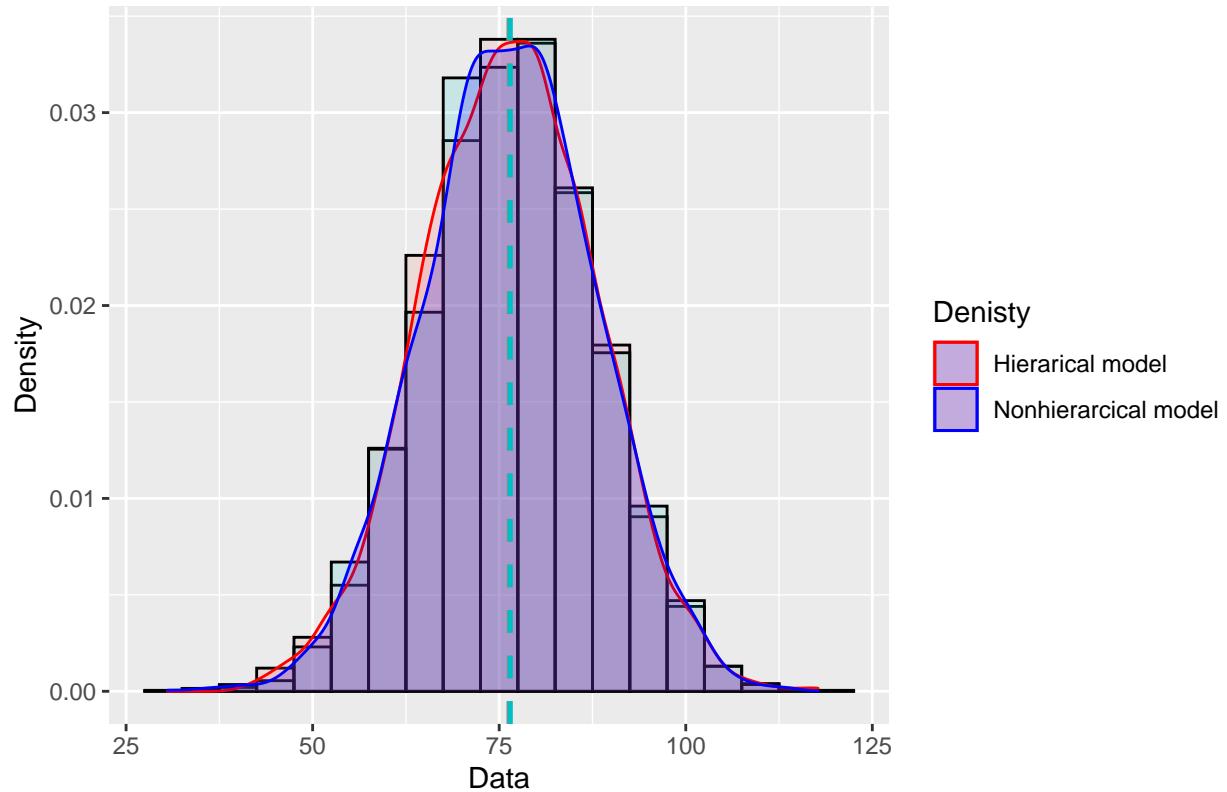
Hieratical posterior predictive vs original data for the old age group



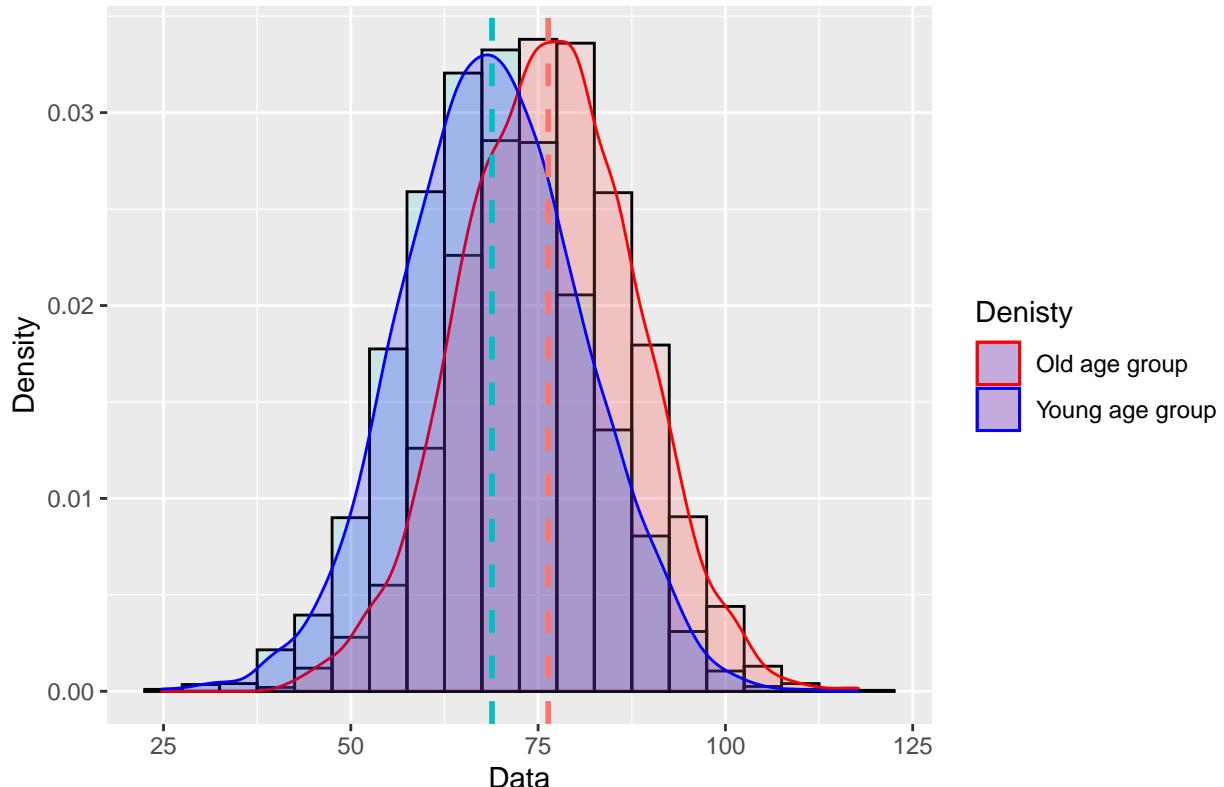
Hierarchical posterior predictive vs original data for the old age group



Posterior predictive distributions of the hierarcical versus non–hierarcical d



Posterior predictive distributions of the non hierarchical models for the old vs young age groups



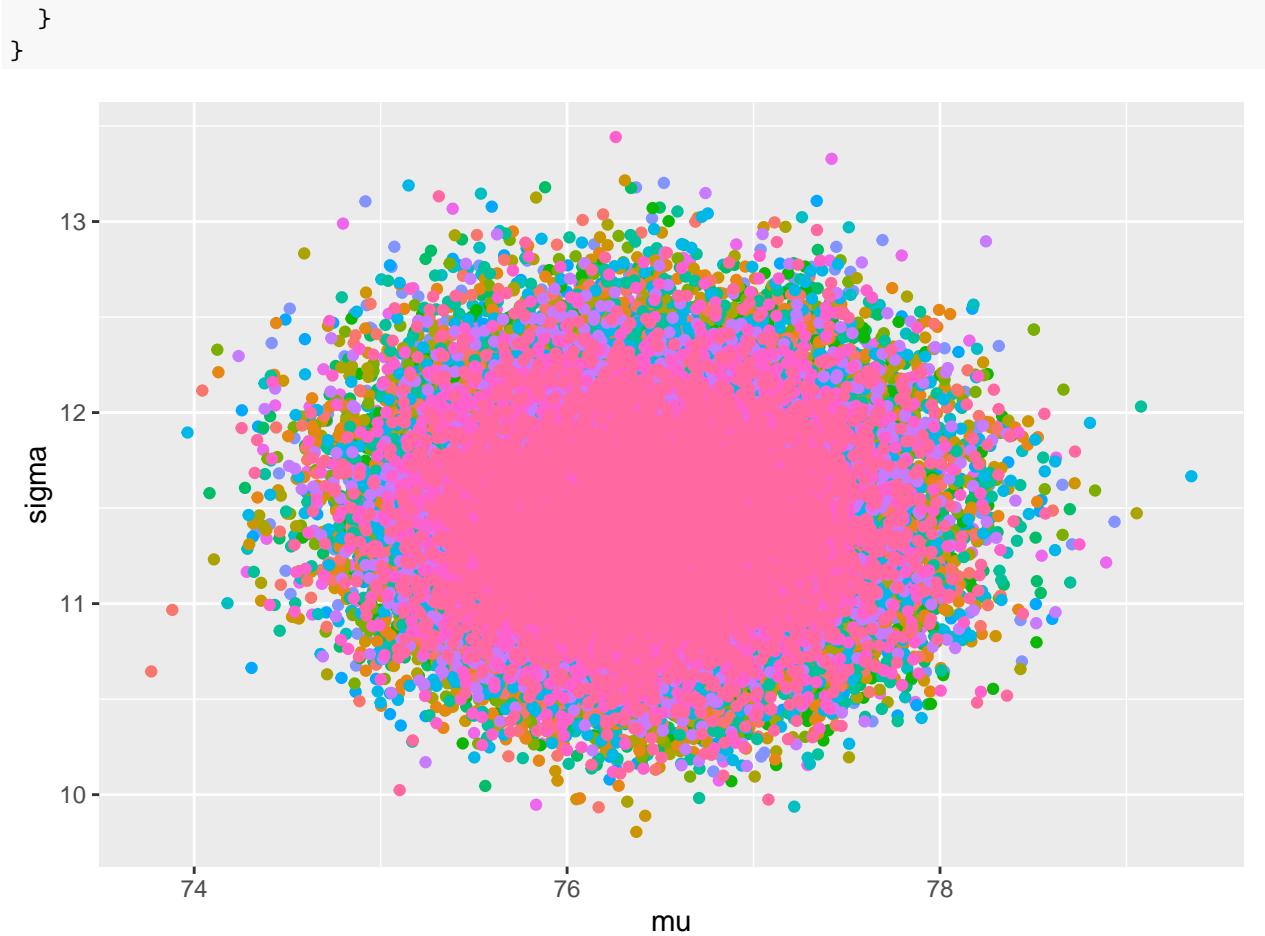
Sensitivity analysis

```

mean_mu_prior_sensitivity = c(0, 50, 100, 1000)
mean_sigma_prior_old_sensitivity = c(1, 10, 100, 1000)
var_prior_old_sensitivity = c(1, 10, 100, 1000)
fit_sensitivity = c()
for (i in 1:length(mean_mu_prior_sensitivity)){
  for (j in 1:length(mean_sigma_prior_old_sensitivity)){
    data_sensitivity <- list(
      y = data_old$BloodPressure,
      N = length(data_old$BloodPressure),
      mean_mu_prior = mean_mu_prior_sensitivity[i],
      mean_sigma_prior = mean_sigma_prior_old_sensitivity[j],
      var_prior = var_prior_old_sensitivity[j]
    )

    fit_sensitivity = c(fit_sensitivity,sampling(nonhieramodel,
      data = data_nonhiera_old,           # named list of data
      chains = 4,                      # number of Markov chains
      warmup = 1000,                   # number of warmup iterations per chain
      iter = 2000,                     # total number of iterations per chain
      cores = 4,                       # number of cores (could use one per chain)
      refresh = 0                      # no progress shown
    ))
  }
}

```



Discussion

Conclusion

Self-reflection

References

Gurven, Michael, Aaron D. Blackwell, Daniel Eid Rodríguez, Jonathan Stieglitz, and Hillard Kaplan. 2012. “Does Blood Pressure Inevitably Rise with Age?” *Hypertension* 60 (1): 25–33. <https://doi.org/10.1161/hypertensionaha.111.189100>.

Mayo Clinic Staff. 2021. “Blood Pressure Chart: What Your Reading Means.” 2021. <https://www.mayoclinic.org/diseases-conditions/high-blood-pressure/in-depth/blood-pressure/art-20050982>.