Homework 04

Ihsan Kahveci

2022-05-09

Contents

1 Questions

1.1 *Q1*

1.1.1 *Q1.a*

For a student i in school j, our Bayesian random effects one-way analysis of variance model is written as:

$$y_i = \alpha_{j[i]} + \epsilon_i,$$
$$\epsilon_i \overset{iid}{\sim} N(0, \sigma_y^2),$$
$$\alpha_j \overset{iid}{\sim} N(\mu_\alpha, \sigma_\alpha^2)$$

where the standard deviation of error in estimating individual student performance (σ_y), the mean performance across all schools (μ_α), and the standard deviation in performance across all schools (σ_α) are the unknown parameters to be estimated.

1.1.2 *Q1.b*

For this scenario, I used uninformative priors because there is no a priori information about the schools or students:

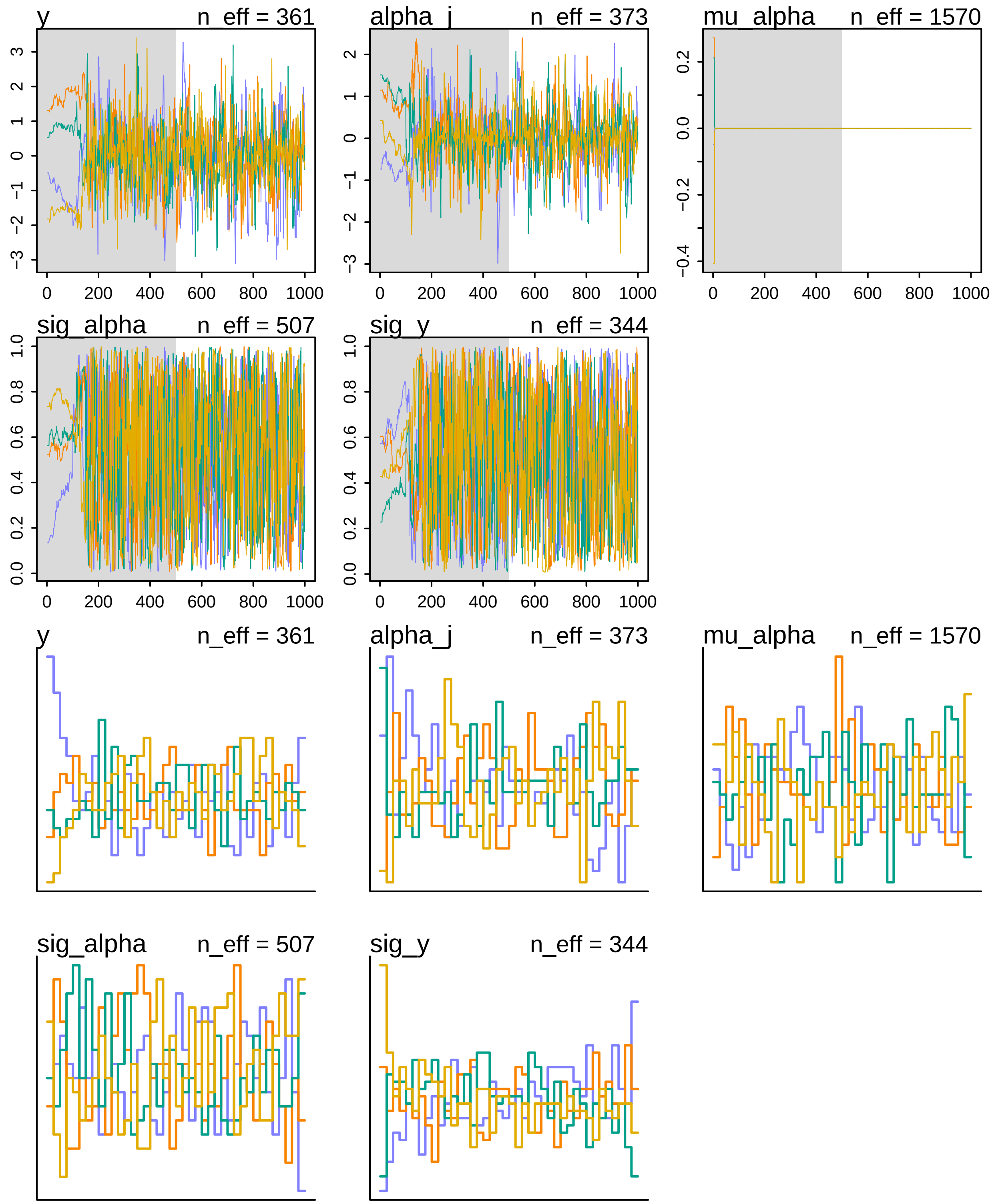
$$\mu_\alpha \sim N(0, 0.0001)$$
$$\sigma_\alpha \sim \text{Uniform}(0, 1)$$
$$\sigma_y \sim \text{Uniform}(0, 1)$$

1.1.3 *Q1.c*

```
## Running MCMC with 4 sequential chains, with 1 thread(s) per chain...
##
## Chain 1 Iteration:   1 / 1000 [  0%]  (Warmup)
## Chain 1 Iteration: 100 / 1000 [ 10%]  (Warmup)
## Chain 1 Iteration: 200 / 1000 [ 20%]  (Warmup)
## Chain 1 Iteration: 300 / 1000 [ 30%]  (Warmup)
## Chain 1 Iteration: 400 / 1000 [ 40%]  (Warmup)
## Chain 1 Iteration: 500 / 1000 [ 50%]  (Warmup)
## Chain 1 Iteration: 501 / 1000 [ 50%]  (Sampling)
## Chain 1 Iteration: 600 / 1000 [ 60%]  (Sampling)
## Chain 1 Iteration: 700 / 1000 [ 70%]  (Sampling)
## Chain 1 Iteration: 800 / 1000 [ 80%]  (Sampling)
## Chain 1 Iteration: 900 / 1000 [ 90%]  (Sampling)
## Chain 1 Iteration: 1000 / 1000 [100%]  (Sampling)
## Chain 1 finished in 0.5 seconds.
## Chain 2 Iteration:   1 / 1000 [  0%]  (Warmup)
## Chain 2 Iteration: 100 / 1000 [ 10%]  (Warmup)
## Chain 2 Iteration: 200 / 1000 [ 20%]  (Warmup)
## Chain 2 Iteration: 300 / 1000 [ 30%]  (Warmup)
## Chain 2 Iteration: 400 / 1000 [ 40%]  (Warmup)
## Chain 2 Iteration: 500 / 1000 [ 50%]  (Warmup)
## Chain 2 Iteration: 501 / 1000 [ 50%]  (Sampling)
## Chain 2 Iteration: 600 / 1000 [ 60%]  (Sampling)
```

```
## Chain 2 Iteration: 700 / 1000 [ 70%]  (Sampling)
## Chain 2 Iteration: 800 / 1000 [ 80%]  (Sampling)
## Chain 2 Iteration: 900 / 1000 [ 90%]  (Sampling)
## Chain 2 Iteration: 1000 / 1000 [100%]  (Sampling)
## Chain 2 finished in 0.9 seconds.
## Chain 3 Iteration:   1 / 1000 [  0%]  (Warmup)
## Chain 3 Iteration: 100 / 1000 [ 10%]  (Warmup)
## Chain 3 Iteration: 200 / 1000 [ 20%]  (Warmup)
## Chain 3 Iteration: 300 / 1000 [ 30%]  (Warmup)
## Chain 3 Iteration: 400 / 1000 [ 40%]  (Warmup)
## Chain 3 Iteration: 500 / 1000 [ 50%]  (Warmup)
## Chain 3 Iteration: 501 / 1000 [ 50%]  (Sampling)
## Chain 3 Iteration: 600 / 1000 [ 60%]  (Sampling)
## Chain 3 Iteration: 700 / 1000 [ 70%]  (Sampling)
## Chain 3 Iteration: 800 / 1000 [ 80%]  (Sampling)
## Chain 3 Iteration: 900 / 1000 [ 90%]  (Sampling)
## Chain 3 Iteration: 1000 / 1000 [100%]  (Sampling)
## Chain 3 finished in 0.5 seconds.
## Chain 4 Iteration:   1 / 1000 [  0%]  (Warmup)
## Chain 4 Iteration: 100 / 1000 [ 10%]  (Warmup)
## Chain 4 Iteration: 200 / 1000 [ 20%]  (Warmup)
## Chain 4 Iteration: 300 / 1000 [ 30%]  (Warmup)
## Chain 4 Iteration: 400 / 1000 [ 40%]  (Warmup)
## Chain 4 Iteration: 500 / 1000 [ 50%]  (Warmup)
## Chain 4 Iteration: 501 / 1000 [ 50%]  (Sampling)
## Chain 4 Iteration: 600 / 1000 [ 60%]  (Sampling)
## Chain 4 Iteration: 700 / 1000 [ 70%]  (Sampling)
## Chain 4 Iteration: 800 / 1000 [ 80%]  (Sampling)
## Chain 4 Iteration: 900 / 1000 [ 90%]  (Sampling)
## Chain 4 Iteration: 1000 / 1000 [100%]  (Sampling)
## Chain 4 finished in 0.5 seconds.
##
## All 4 chains finished successfully.
## Mean chain execution time: 0.6 seconds.
## Total execution time: 2.7 seconds.
```

1.1.4 *Q1.d*

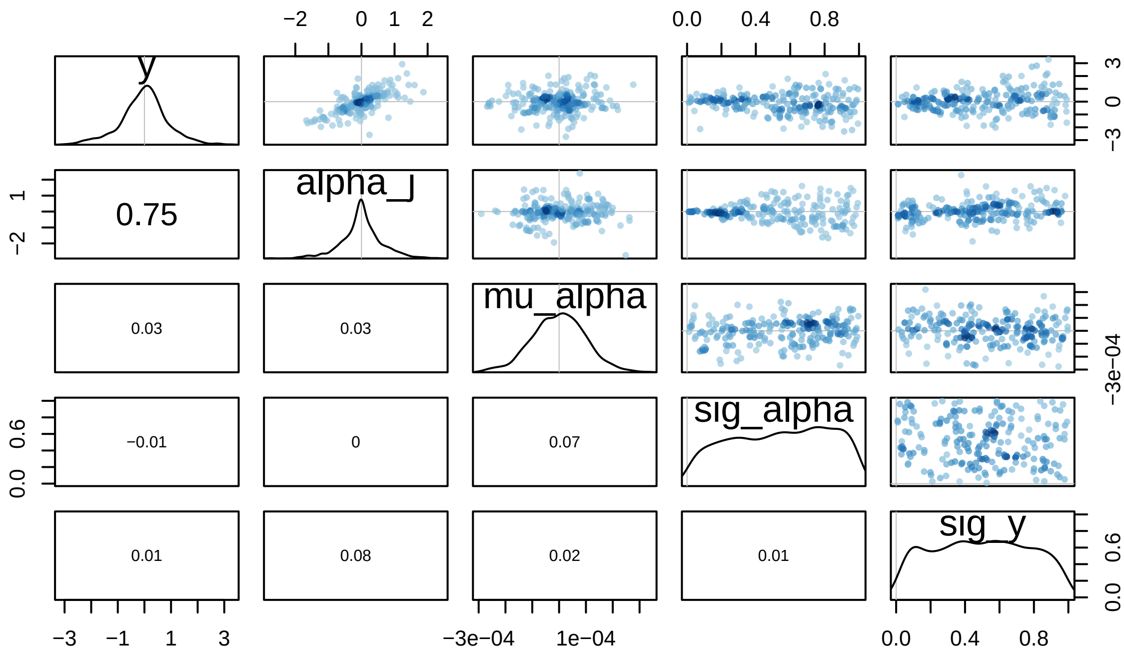


The model is converged in 1000 iterations.

1.1.5 *Q1.e*

Table 1: Predictive distribution sumeduy of Netherlands TFR, 2020-2025

vars	mean	sd	5.5%	94.5%	n_eff	Rhat4
y	-0.018	0.892	-1.564	1.380	360.570	1.004
alpha_j	0.005	0.633	-1.019	1.054	372.965	1.007
mu_alpha	0.000	0.000	0.000	0.000	1569.986	1.004
sig_alpha	0.541	0.283	0.079	0.957	506.969	1.003
sig_y	0.495	0.275	0.072	0.926	344.425	1.035

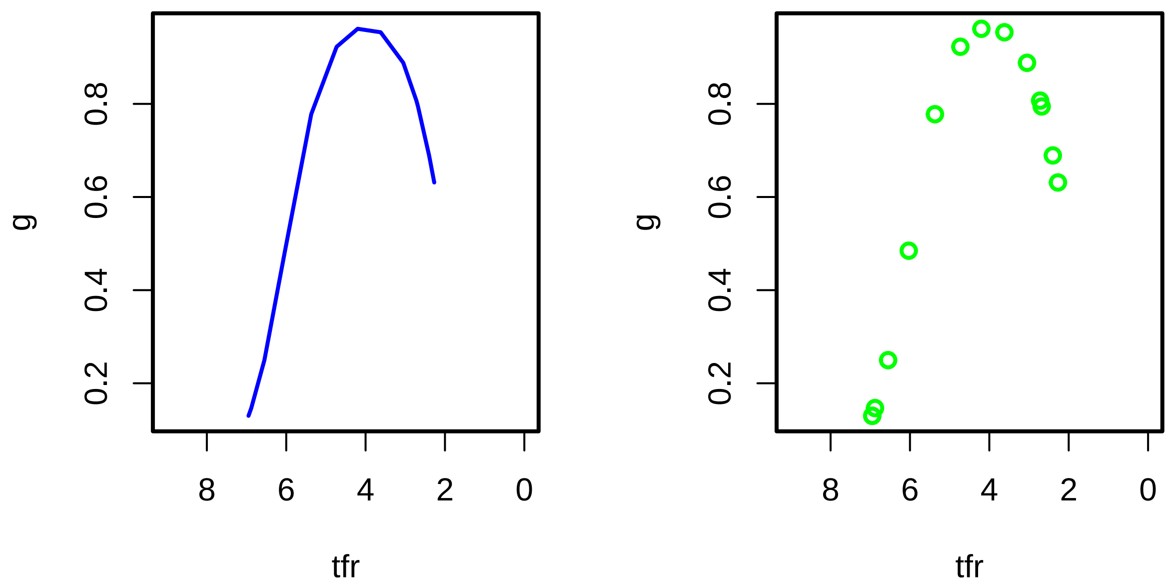


This model was fit using STAN which uses Hamiltonian Monte Carlo algorithm.

1.2 *Q2*

1.2.1 *Q2.a & b*

Using Adrian's demo example:



The plots shows a great fit. As expected, the third phase region of the plot is blank because Peru has not reached Phase 3 yet.

1.2.2 *Q2.c*

1.3 *Q3*

1.3.1 *Q3.a*

Table 2: Total fertility rates, Netherlands, 1950-2020

Period Start	TFR
1950	3.052
1955	3.097
1960	3.166
1965	2.795
1970	2.100
1975	1.598
1980	1.515
1985	1.555
1990	1.592
1995	1.599
2000	1.740
2005	1.746
2010	1.732
2015	1.660

The start of Phase III of the fertility model is defined by two consecutive five-year increases of TFR while staying below a TFR of 2. Looking at TFR data for the Netherlands, we see that Phase III starts with the period beginning in **1985**.

1.3.2 *Q3.b*

I fit an order 1 autoregressive model to the subset of Netherlands TFR data in Phase III, and extract some model parameters below. *Note that the AR(1) model was fit using the "mle" method.*

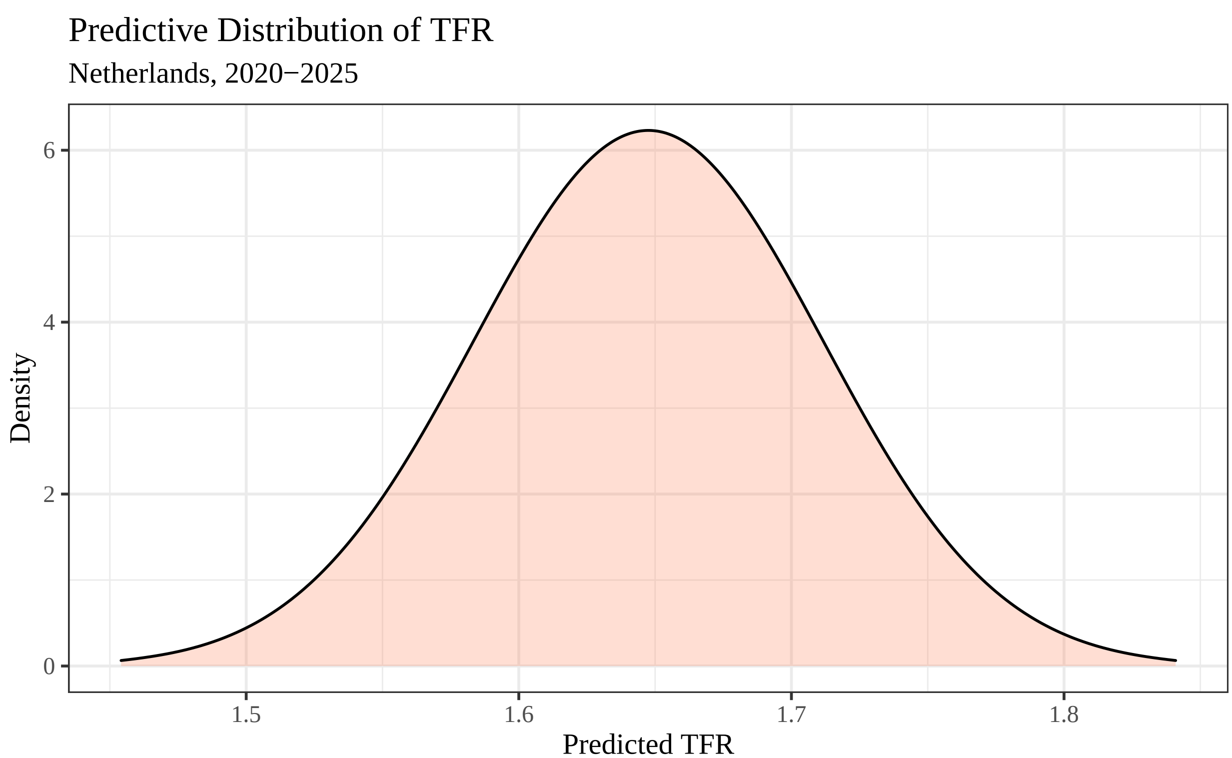
Table 3: Netherlands Phase III AR(1) model parameters

term	estimate	std.error
ar1	0.542	0.152
intercept	1.648	0.024

1.3.3 *Q3.c*

Table 4: Predictive distribution sumeduy of Netherlands TFR, 2020-2025

Mean	Median	2.5% PI	97.5% PI
1.648	1.648	1.525	1.77

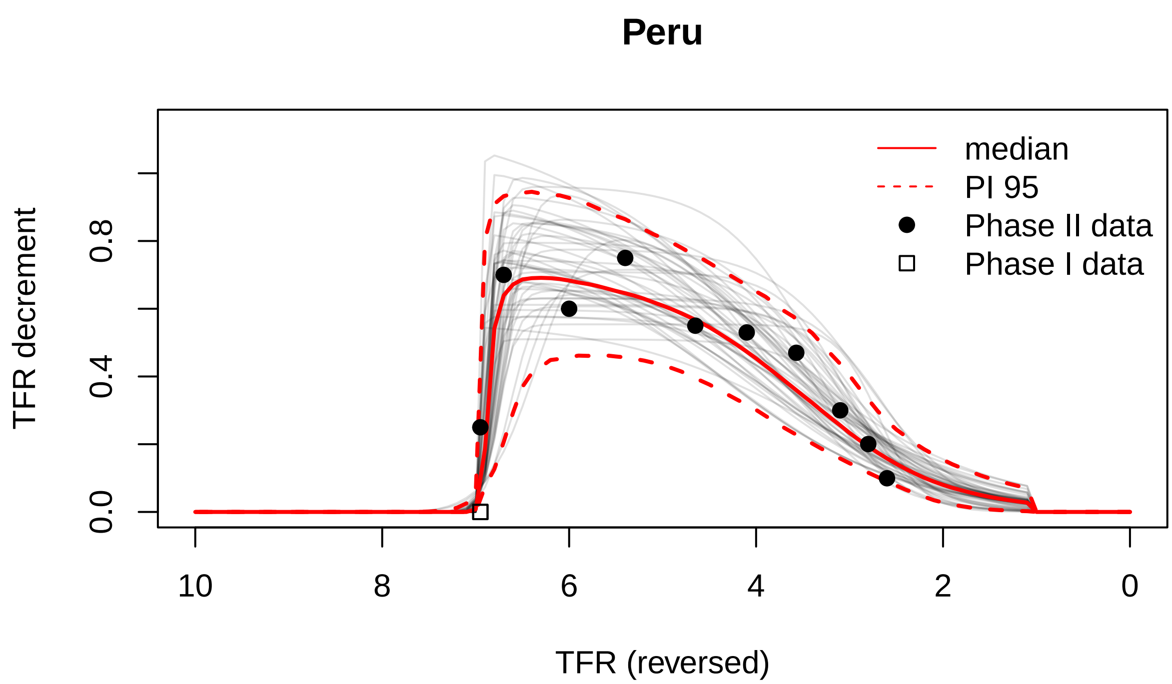


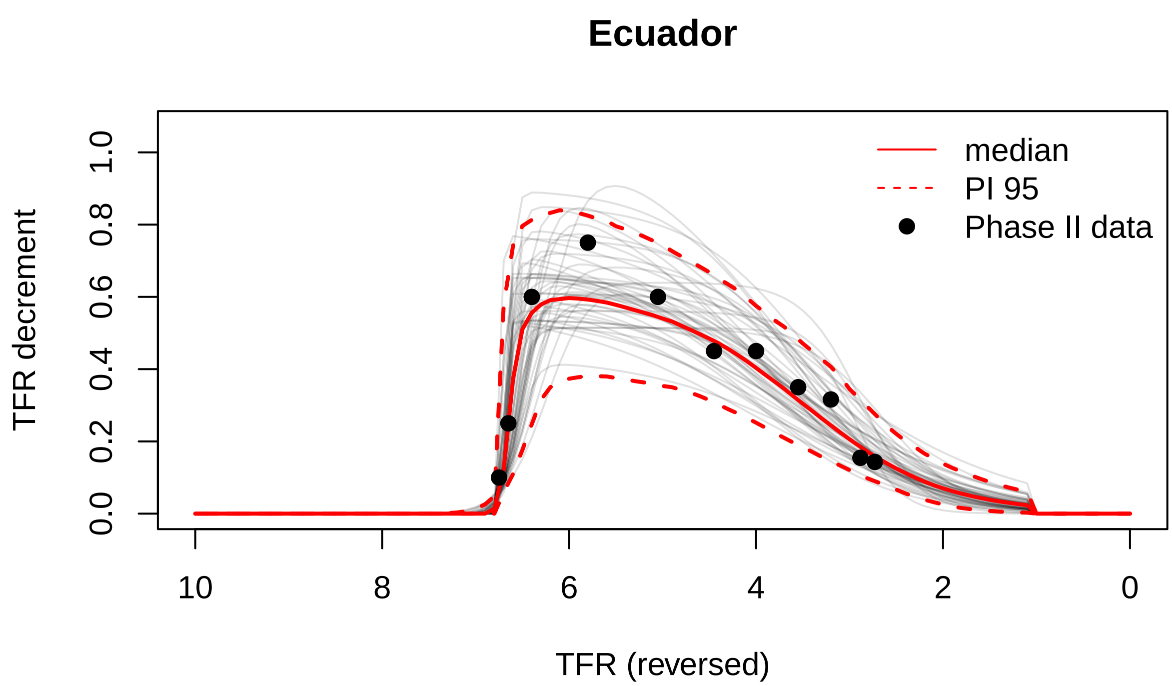
1.4 *Q4*

1.4.1 *Q4.a*

The fully converged simulation is loaded using the `README` file contained with the data.

1.4.2 *Q4.b*





From the above graphs comparing the Phase II double logistic models for Peru and Ecuador, we see that the TFR decrements maintains higher values across TFR in Peru for both the median and 95% PI, suggesting that fertility is declining faster in Peru than Ecuador We can also observe this trend by looking directly at the TFR for both countries over time:

Table 5: TFR over time

Period Start	Ecuador	Peru
1950	6.75	6.95
1955	6.75	6.95
1960	6.65	6.88
1965	6.40	6.55
1970	5.80	6.03
1975	5.05	5.37
1980	4.45	4.73
1985	3.97	4.20
1990	3.55	3.62
1995	3.27	3.05
2000	2.94	2.72
2005	2.69	2.68
2010	2.56	2.40
2015	2.44	2.27

1.4.3 *Q4.c*

By getting the TFR trajectories for Peru and Ecuador, we can find the posterior predictive probability for many conditions.

First, we can determine the probability of Peru having a higher TFR than Ecuador in each five-year period from 2020 through 2095 by finding the mean number of times Peru has a higher TFR than Ecuador across all simulations:

Period Start	Pr(peru > edu)
2020	0.371
2025	0.384
2030	0.416
2035	0.411
2040	0.414
2045	0.435
2050	0.429
2055	0.434
2060	0.458
2065	0.464
2070	0.472
2075	0.471
2080	0.475
2085	0.489
2090	0.479
2095	0.480

We can also find the probability that the TFR of Peru will be higher than that of Ecuador in all five-year periods from 2020 through 2095 by finding the mean number of simulations where Peru has a higher TFR than Ecuador for all periods. This value is calculated to be **0.075**.

2 Appendix

```r
# Prep work ---------------------------------------------------------------
# Load libraries
library(tidyverse)
library(rethinking)
library(bayesTFR)
options(mc.cores = parallel::detectCores())

# Data
data("egsingle", package = "mlmRev")
data("tfr", package = "wpp2019")
tfr_sim_dir <- "TFR/sim01192018"

tfr_all <- tfr %>%
  select(-country_code, -last.observed) %>%
  pivot_longer(
    -name,
    values_to = "tfr",
    names_pattern = "^(.*)-",
    names_to = "year") %>%
  mutate(year = as.numeric(year))

# Control randomness
set.seed(57)


# Question 1 --------------------------------------------------------------

edu_data <- egsingle %>%
  filter(year == .5) %>%
  select(childid, schoolid, math)

# Question 1c -------------------------------------------------------------

dat <- list(
  math = edu_data$math,
  student = edu_data$childid,
  school = edu_data$schoolid)

m1 = ulam(
  alist(
    y ~ dnorm(alpha_j, sig_y),
    alpha_j ~ dnorm(mu_alpha, sig_alpha),
    # Priors
    mu_alpha ~ dnorm(0, 0.0001),
    sig_alpha ~ uniform(0, 1),
    sig_y ~ uniform(0, 1)
    ), data=dat, chains=4)
traceplot(m1)
trankplot(m1)
out = rethinking::precis(m1, depth = 2)
names = rownames(out)
```

```r
summary = bind_cols(vars = names, as_tibble(out))

knitr::kable(
  summary,
  booktabs = TRUE,
  digits = 3,
  caption = "Predictive distribution sumeduy of Netherlands TFR, 2020-2025"
)
pairs(m1)

# Question 2 -------------------------------------------------------------

peru_tfr <- tfr_all %>% filter(name == "Peru") %>% select(-name)

# Question 2a ------------------------------------------------------------
tfr  <- peru_tfr$tfr

dl <- function (x, d, a1, a2, a3, a4) {
  d / (1+exp (-(x-a2)/a1)) - d / (1+exp(-(x-a4)/a3)) }

par (mfrow=c(1,2), lwd=2)

d <- 1
a4 <- 6
a3 <- 0.5
a2 <- 2
a1 <- 0.5
g <- dl(tfr,d,a1,a2,a3,a4)

plot (tfr, g, type="line", xlim=rev(c(0,9)), col="blue")
plot (tfr, g, type="p", xlim=rev(c(0,9)), col="green")


# Question 3 -------------------------------------------------------------

nld_tfr <- tfr_all %>% filter(name == "Netherlands") %>% select(-name)


# Question 3a ------------------------------------------------------------

nld_phase3_year <- nld_tfr %>%
  arrange(year) %>%
  filter(tfr < 2) %>%
  mutate(
    year_diff = lead(year) - year,
    period_5 = year_diff == 5 & lag(year_diff) == 5,
    two_increases = tfr > lag(tfr, 1) & tfr < lead(tfr, 1)
  ) %>%
  filter(period_5 & two_increases) %>%
  slice(1) %>%
  pull(year)

knitr::kable(
```

```r
  nld_tfr,
  booktabs = TRUE,
  digits = 3,
  col.names = c("Period Start", "TFR"),
  caption = "Total fertility rates, Netherlands, 1950-2020"
)

# Question 3b -----------------------------------------------------------

nld_ts <- nld_tfr %>%
  filter(year >= nld_phase3_year) %>%
  pull(tfr) %>%
  ts(start = 1985, end=2015, frequency = 1)

nld_model = arima(nld_ts, order=c(1,0,0))


knitr::kable(
  broom::tidy(nld_model),
  booktabs = TRUE,
  digits = 3,
  caption = "Netherlands Phase III AR(1) model parameters"
)
nld_sd = sqrt(nld_model$sigma2)
nld_mean = nld_model$coef[2]
nld_pred_dist <- qnorm(seq(.001, .999, .001), mean = nld_mean, sd = nld_sd)

nld_pred_tbl <- tibble(
  Mean = mean(nld_pred_dist),
  Median = median(nld_pred_dist),
  `2.5% PI` = Mean - 1.96 * nld_sd,
  `97.5% PI` = Mean + 1.96 * nld_sd)
knitr::kable(
  nld_pred_tbl,
  booktabs = TRUE,
  digits = 3,
  caption = "Predictive distribution sumeduy of Netherlands TFR, 2020-2025"
)

ggplot(tibble(nld_pred_dist), aes(x = nld_pred_dist)) +
  geom_density(fill = "coral", alpha = .25) +
  theme_bw() +
  theme(text = element_text(family = "serif")) +
  labs(
    title = "Predictive Distribution of TFR",
    subtitle = "Netherlands, 2020-2025",
    x = "Predicted TFR",
    y = "Density"
  )


# Question 4 ------------------------------------------------------------
```

```r
tfr_peru_ecu <- tfr_all %>%
  filter(name %in% c("Peru", "Ecuador")) %>%
  pivot_wider(names_from = name, values_from = "tfr") %>%
  rename(`Period Start` = year)


# Question 4.a -----------------------------------------------------------

tfr_phase2_mcmc  <- get.tfr.mcmc(tfr_sim_dir)
tfr_phase3_mcmc <- get.tfr3.mcmc(tfr_sim_dir)
tfr_pred  <- get.tfr.prediction(tfr_sim_dir)


# Question 4.b -----------------------------------------------------------

DLcurve.plot(
  tfr_phase2_mcmc,
  country = "Peru",
  nr.curves = 50,
  pi = 95
)

DLcurve.plot(
  tfr_phase2_mcmc,
  country = "Ecuador",
  nr.curves = 50,
  pi = 95
)

knitr::kable(
  tfr_peru_ecu,
  booktabs = TRUE,
  digits = 3,
  caption = "TFR over time"
)

# Question 4c ------------------------------------------------------------

tfr_traj_peru <- get.tfr.trajectories(tfr_pred, country = "Peru")[-(1:2), ]
tfr_traj_ecu <- get.tfr.trajectories(tfr_pred, country = "Ecuador")[-(1:2), ]

prob_tfr_peru_higher <- rowMeans(tfr_traj_peru > tfr_traj_ecu)

prob_tfr_peru_higher_tbl <- tibble(
  period_start = as.integer(names(prob_tfr_peru_higher)) - 3,
  prob_peru_higher = prob_tfr_peru_higher
)

prob_tfr_peru_higher_all <-
  sum(apply(tfr_traj_peru > tfr_traj_ecu, 2, all)) / ncol(tfr_traj_peru)

knitr::kable(
  prob_tfr_peru_higher_tbl,
```

```
  booktabs = "TRUE",
  digits = 3,
  col.names = c("Period Start", "Pr(peru > edu)")
)
```