STA2201-Lab-8

Alice Huang

22/03/2023

Child mortality in Sri Lanka

In this lab you will be fitting a couple of different models to the data about child mortality in Sri Lanka, which was used in the lecture. Here's the data and the plot from the lecture:

```
lka <- read_csv(here("lka.csv"))
```

```
## Rows: 27 Columns: 7
## -- Column specification ----------------------------------------------------
## Delimiter: ","
## chr (3): country_name, country_code, source
## dbl (4): year, logit_ratio, se, ratio
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio")
```

Ratio of neonatal to other child mortality (logged), Sri Lanka
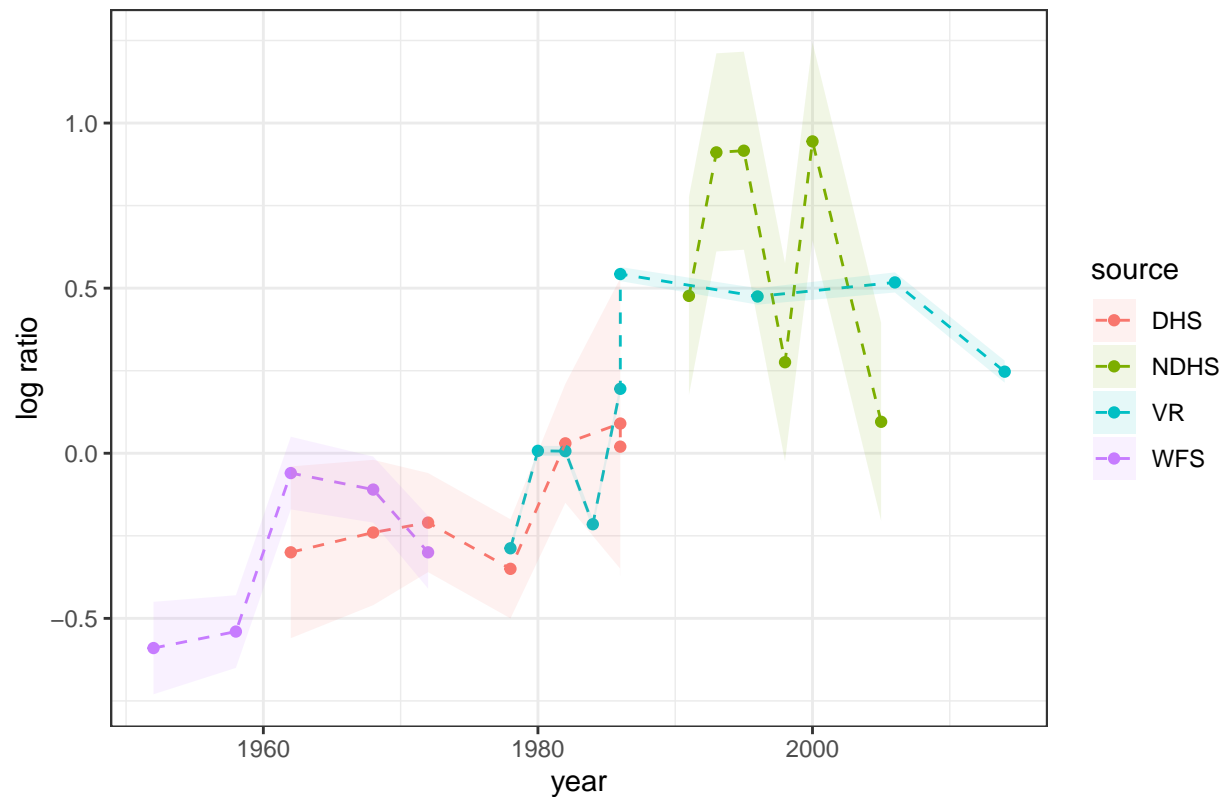
Fitting a linear model

Let's firstly fit a linear model in time to these data. Here's the code to do this:

```
observed_years <- lka$year
years <- min(observed_years):max(observed_years)
nyears <- length(years)

stan_data <- list(y = lka$logit_ratio, year_i = observed_years - years[1]+1,
                  T = nyears, years = years, N = length(observed_years),
                  mid_year = mean(years), se = lka$se)

mod <- stan(data = stan_data, file = here("lka_linear_me.stan"))
```

```
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 3.4e-05 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.34 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
```

```
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.049 seconds (Warm-up)
## Chain 1:                0.045 seconds (Sampling)
## Chain 1:                0.094 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 7e-06 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.07 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.063 seconds (Warm-up)
## Chain 2:                0.044 seconds (Sampling)
## Chain 2:                0.107 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 8e-06 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
```

```
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.051 seconds (Warm-up)
## Chain 3:                0.039 seconds (Sampling)
## Chain 3:                0.09 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 8e-06 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.053 seconds (Warm-up)
## Chain 4:                0.041 seconds (Sampling)
## Chain 4:                0.094 seconds (Total)
## Chain 4:
```

Extract the results:

```
res <- mod %>%
  gather_draws(mu[t]) %>%
  median_qi() %>%
  mutate(year = years[t])
```
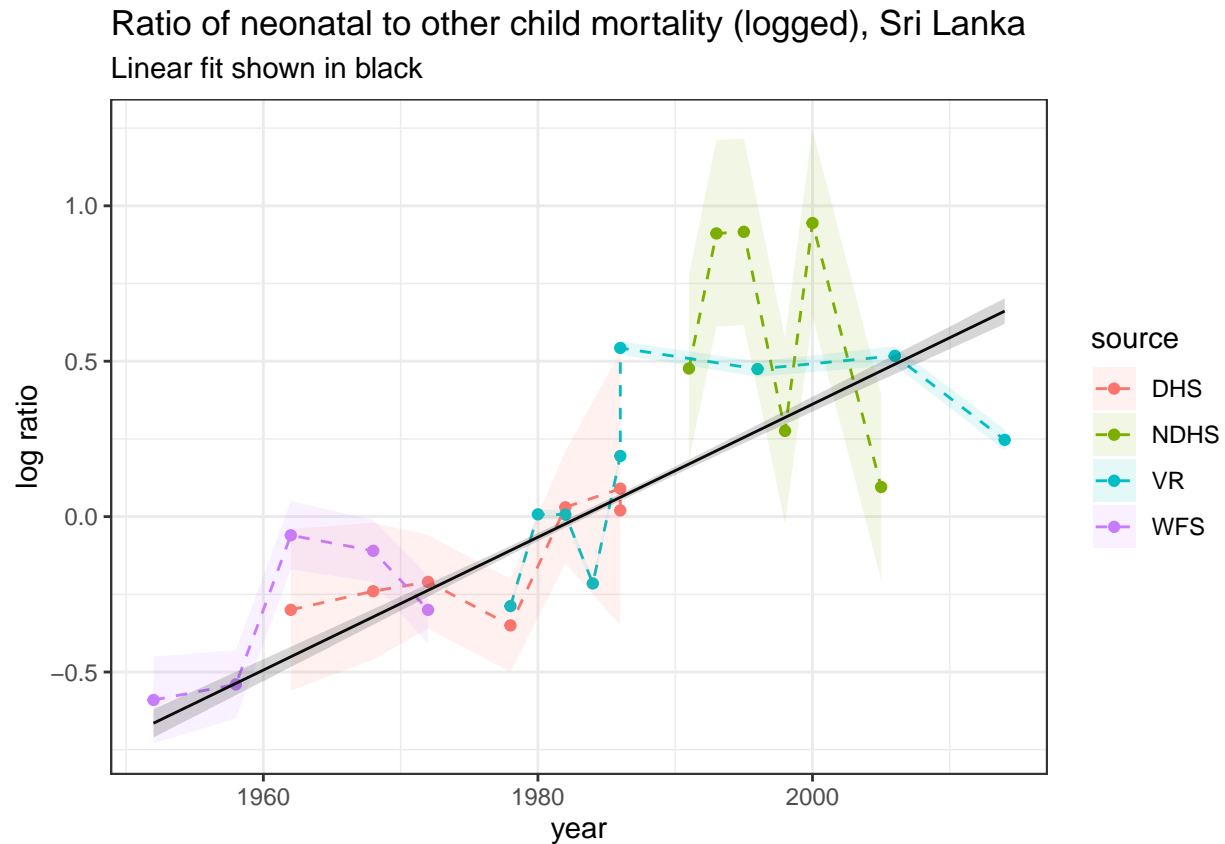
Plot the results:

```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
```

```
geom_line(data = res, aes(year, .value)) +
geom_ribbon(data = res, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
theme_bw()+
labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka",
     y = "log ratio", subtitle = "Linear fit shown in black")
```


Question 1

Project the linear model above out to 2023 by adding a `generated quantities` block in Stan (do the projections based on the expected value μ). Plot the resulting projections on a graph similar to that above.

```
stan_data <- list(y = lka$logit_ratio, year_i = observed_years - years[1]+1,
                   T = nyears, years = years, N = length(observed_years),
                   mid_year = mean(years), se = lka$se, P=9)

mod2 <- stan(data = stan_data,
             file = here("lab10.stan"))
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\Alice\Desktop\GRAD
## SCHOOL\STA2201\STA2201AppliedStatAliceHuang\lab10.stan'
```

```
##
```

5

```
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000155 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.55 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.065 seconds (Warm-up)
## Chain 1:                0.05 seconds (Sampling)
## Chain 1:                0.115 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 9e-06 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.064 seconds (Warm-up)
## Chain 2:                0.057 seconds (Sampling)
## Chain 2:                0.121 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 8e-06 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
```

```
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.072 seconds (Warm-up)
## Chain 3:                0.052 seconds (Sampling)
## Chain 3:                0.124 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 8e-06 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.053 seconds (Warm-up)
## Chain 4:                0.048 seconds (Sampling)
## Chain 4:                0.101 seconds (Total)
## Chain 4:
```

```r
res2 <- mod2 %>%
   gather_draws(mu[t]) %>%
  median_qi() %>%
  mutate(year = years[t])

res_p2 <- mod2 %>%
  gather_draws(mu_p[p]) %>%
  median_qi() %>%
```

```
  mutate(year = years[nyears]+p)
```

```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res2, aes(year, .value)) +
  geom_ribbon(data=res2, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2) +
  geom_line(data = res_p2, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p2, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio", subti
```


Ratio of neonatal to other child mortality (logged), Sri Lanka
Linear fit shown in black
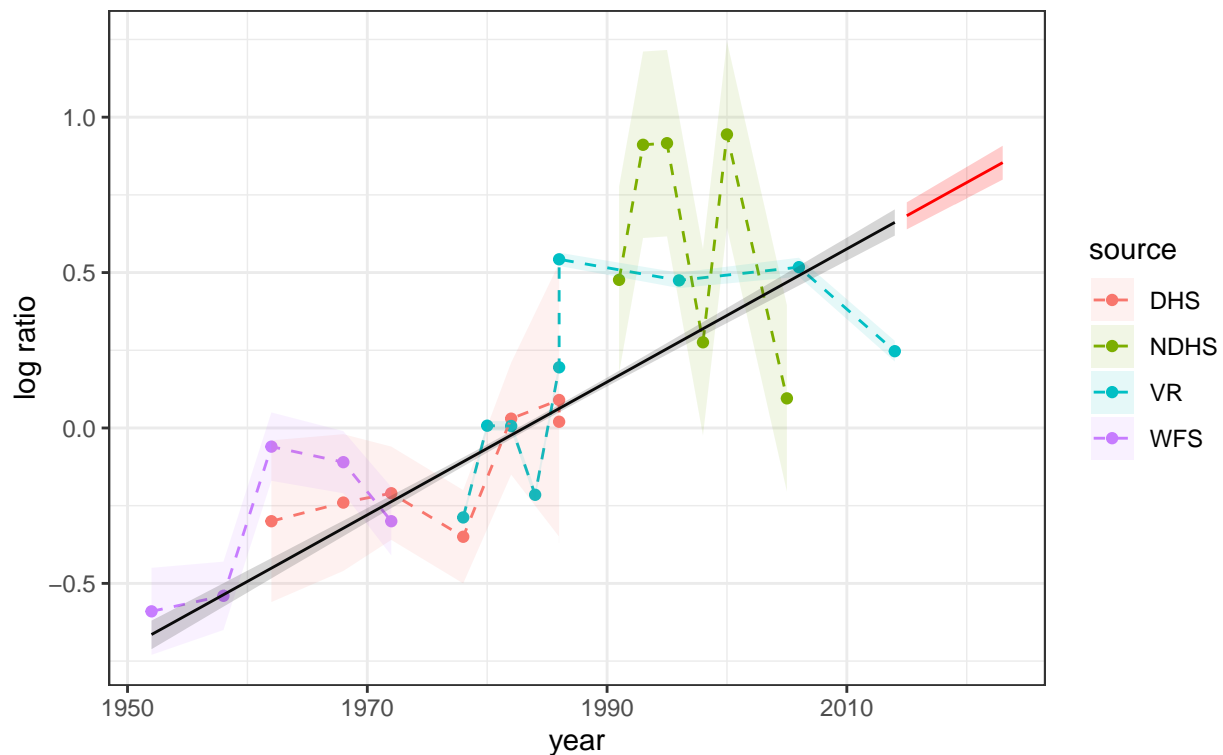
Question 2

Code up and estimate a first order random walk model to fit to the Sri Lankan data, taking into account measurement error, and project out to 2023.

```
stan_data <- list(y = lka$logit_ratio, year_i = observed_years - years[1]+1,
                  T = nyears, years = years, N = length(observed_years),
                  se = lka$se, P=9)
```

8

```r
mod3 <- stan(data = stan_data,
             file = here("lab10_2.stan"))
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\Alice\Desktop\GRAD
## SCHOOL\STA2201\STA2201AppliedStatAliceHuang\lab10_2.stan'


##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 8.2e-05 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.82 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.39 seconds (Warm-up)
## Chain 1:                0.255 seconds (Sampling)
## Chain 1:                0.645 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 1.3e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.13 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
```

```
## Chain 2:  Elapsed Time: 0.352 seconds (Warm-up)
## Chain 2:                 0.256 seconds (Sampling)
## Chain 2:                 0.608 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 1.3e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.13 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.331 seconds (Warm-up)
## Chain 3:                 0.244 seconds (Sampling)
## Chain 3:                 0.575 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 2e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.2 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.399 seconds (Warm-up)
## Chain 4:                 0.258 seconds (Sampling)
## Chain 4:                 0.657 seconds (Total)
## Chain 4:
```
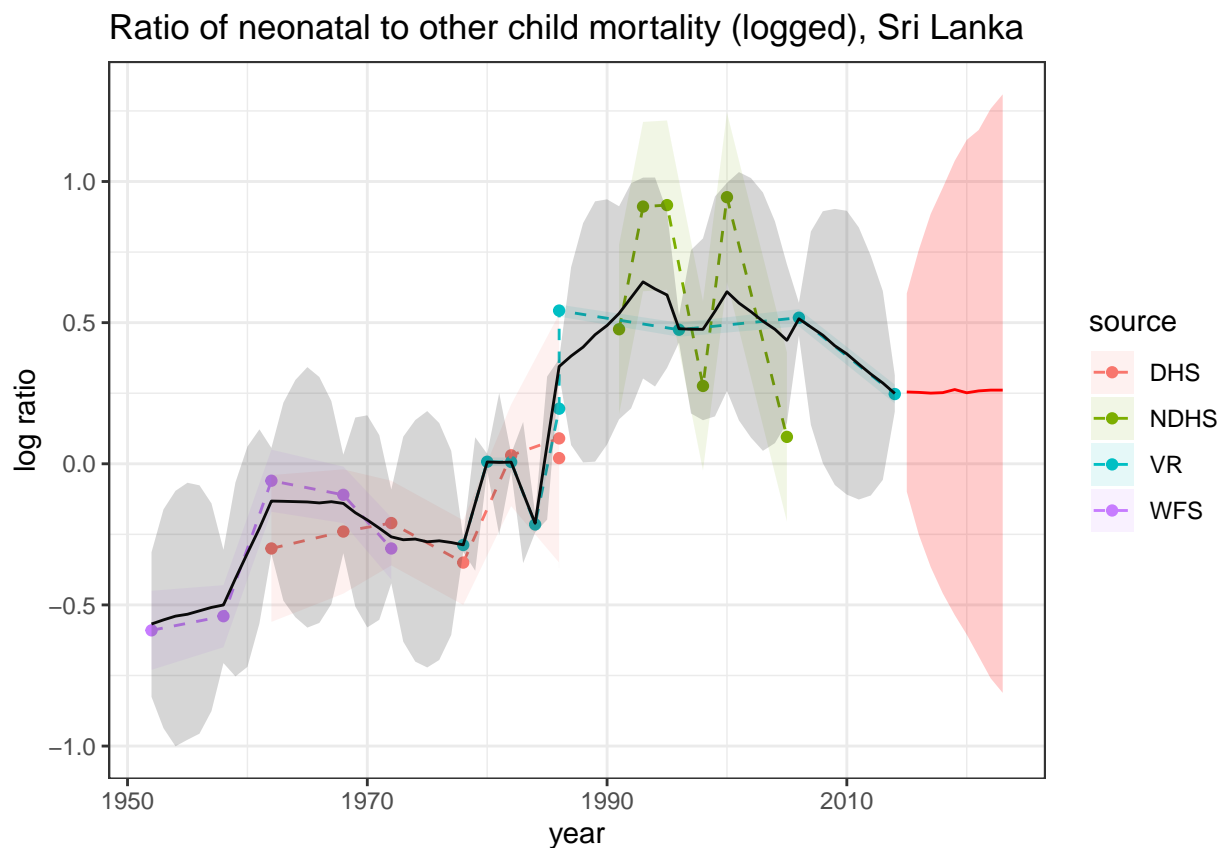
```
res3 <- mod3 %>%
   gather_draws(mu[t]) %>%
  median_qi() %>%
  mutate(year = years[t])

res_p3 <- mod3 %>%
  gather_draws(mu_p[p]) %>%
  median_qi() %>%
  mutate(year = years[nyears]+p)
```

```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res3, aes(year, .value)) +
  geom_ribbon(data=res3, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2) +
  geom_line(data = res_p3, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p3, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio")
```


Ratio of neonatal to other child mortality (logged), Sri Lanka

Question 3

Now alter your model above to estimate and project a second-order random walk model (RW2).

```
stan_data <- list(y = lka$logit_ratio, year_i = observed_years - years[1]+1,
                  T = nyears, years = years, N = length(observed_years),
                  se = lka$se, P=9)


mod4 <- stan(data = stan_data,
             file = here("lab10_3.stan"))
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\Alice\Desktop\GRAD
## SCHOOL\STA2201\STA2201AppliedStatAliceHuang\lab10_3.stan'


##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000103 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.03 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 2.319 seconds (Warm-up)
## Chain 1:                1.905 seconds (Sampling)
## Chain 1:                4.224 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0.000124 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 1.24 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
```

```
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 2.415 seconds (Warm-up)
## Chain 2:                1.97 seconds (Sampling)
## Chain 2:                4.385 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 2.8e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.28 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 2.693 seconds (Warm-up)
## Chain 3:                2.321 seconds (Sampling)
## Chain 3:                5.014 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 2.5e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.25 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
```
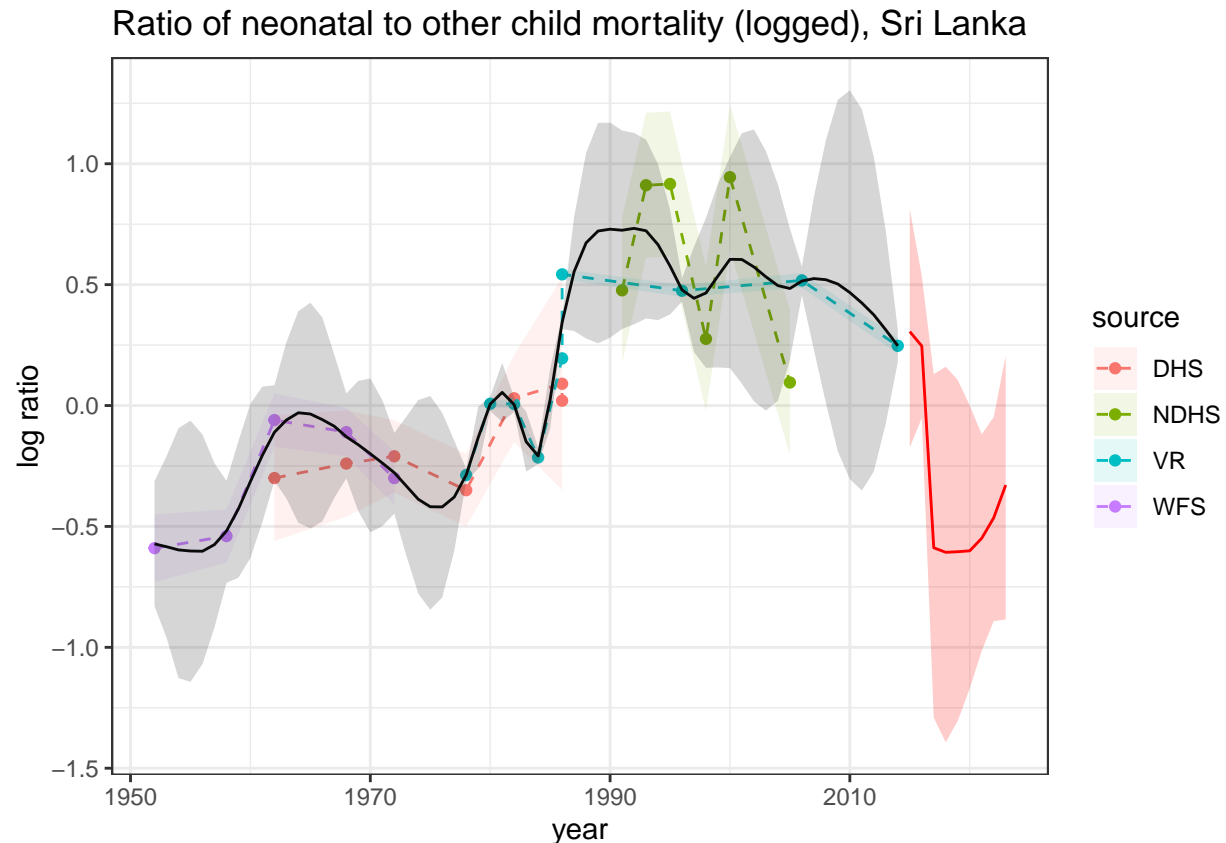
```
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 1.856 seconds (Warm-up)
## Chain 4:                1.486 seconds (Sampling)
## Chain 4:                3.342 seconds (Total)
## Chain 4:
```

```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```r
res4 <- mod4 %>%
   gather_draws(mu[t]) %>%
  median_qi() %>%
  mutate(year = years[t])

res_p4 <- mod4 %>%
  gather_draws(mu_p[p]) %>%
  median_qi() %>%
  mutate(year = years[nyears]+p)
```

```r
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res4, aes(year, .value)) +
  geom_ribbon(data=res4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2) +
  geom_line(data = res_p4, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio")
```

Ratio of neonatal to other child mortality (logged), Sri Lanka

Question 4

Run the first order and second order random walk models, including projections out to 2023. Compare these estimates with the linear fit by plotting everything on the same graph.

```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res2, aes(year, .value), col="blue") +
  geom_ribbon(data=res2, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, colour = "blue") +
  geom_line(data = res_p2, aes(year, .value), col = "blue") +
  geom_ribbon(data=res_p2, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "blue") +
  geom_line(data=res3, aes(year, .value), col="green") +
  geom_ribbon(data=res3, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill="green") +
  geom_line(data = res_p3, aes(year, .value), col = "green") +
  geom_ribbon(data=res_p3, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "green") +
  geom_line(data=res4, aes(year, .value), col="red") +
  geom_ribbon(data=res4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, col="red") +
  geom_line(data = res_p4, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio", subti
```

Ratio of neonatal to other child mortality (logged), Sri Lanka
Red = 2nd order random walk, Green = 1st order random walk, Blue = linear
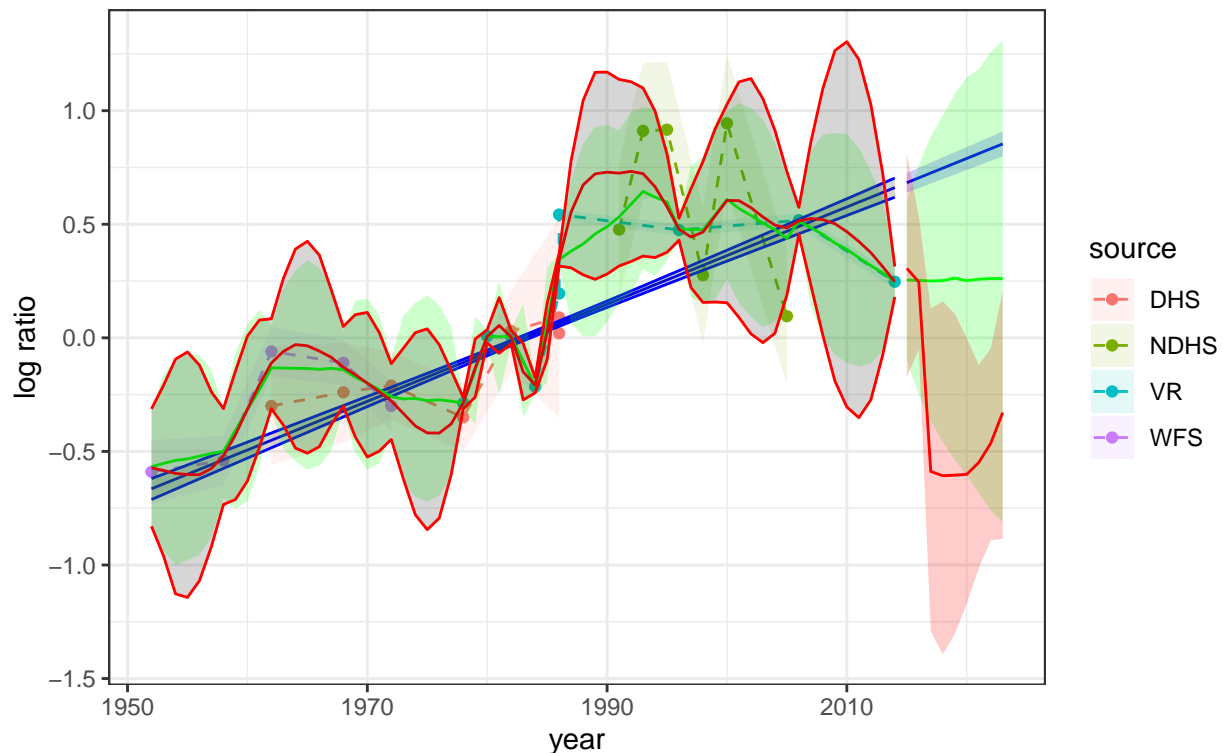
Question 5

Rerun the RW2 model excluding the VR data. Briefly comment on the differences between the two data situations.

First we filter out the VR dataset.

```
lkavr <- lka %>% filter(source != "VR")
observed_yearsvr <- lkavr$year
yearsvr <- min(observed_yearsvr):max(observed_yearsvr)
nyearsvr <- length(yearsvr)

stan_datavr <- list(y = lkavr$logit_ratio, year_i = observed_yearsvr - yearsvr[1]+1,
                    T = nyearsvr, years = yearsvr, N = length(observed_yearsvr),
                    se = lkavr$se, P=9)


mod4 <- stan(data = stan_datavr,
             file = here("lab10_3.stan"))
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\Alice\Desktop\GRAD
## SCHOOL\STA2201\STA2201AppliedStatAliceHuang\lab10_3.stan'
```

```
##
```

```
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 2.5e-05 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.25 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 5.643 seconds (Warm-up)
## Chain 1:                6.591 seconds (Sampling)
## Chain 1:                12.234 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 2.2e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.22 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 5.137 seconds (Warm-up)
## Chain 2:                2.955 seconds (Sampling)
## Chain 2:                8.092 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 1.5e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.15 seconds.
```

```
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 4.459 seconds (Warm-up)
## Chain 3:                3.436 seconds (Sampling)
## Chain 3:                7.895 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 1.6e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.16 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 4.123 seconds (Warm-up)
## Chain 4:                3.078 seconds (Sampling)
## Chain 4:                7.201 seconds (Total)
## Chain 4:

## Warning: There were 44 divergent transitions after warmup. See
## https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

## Warning: There were 4 chains where the estimated Bayesian Fraction of Missing Information was low. S
## https://mc-stan.org/misc/warnings.html#bfmi-low

## Warning: Examine the pairs() plot to diagnose sampling problems
```
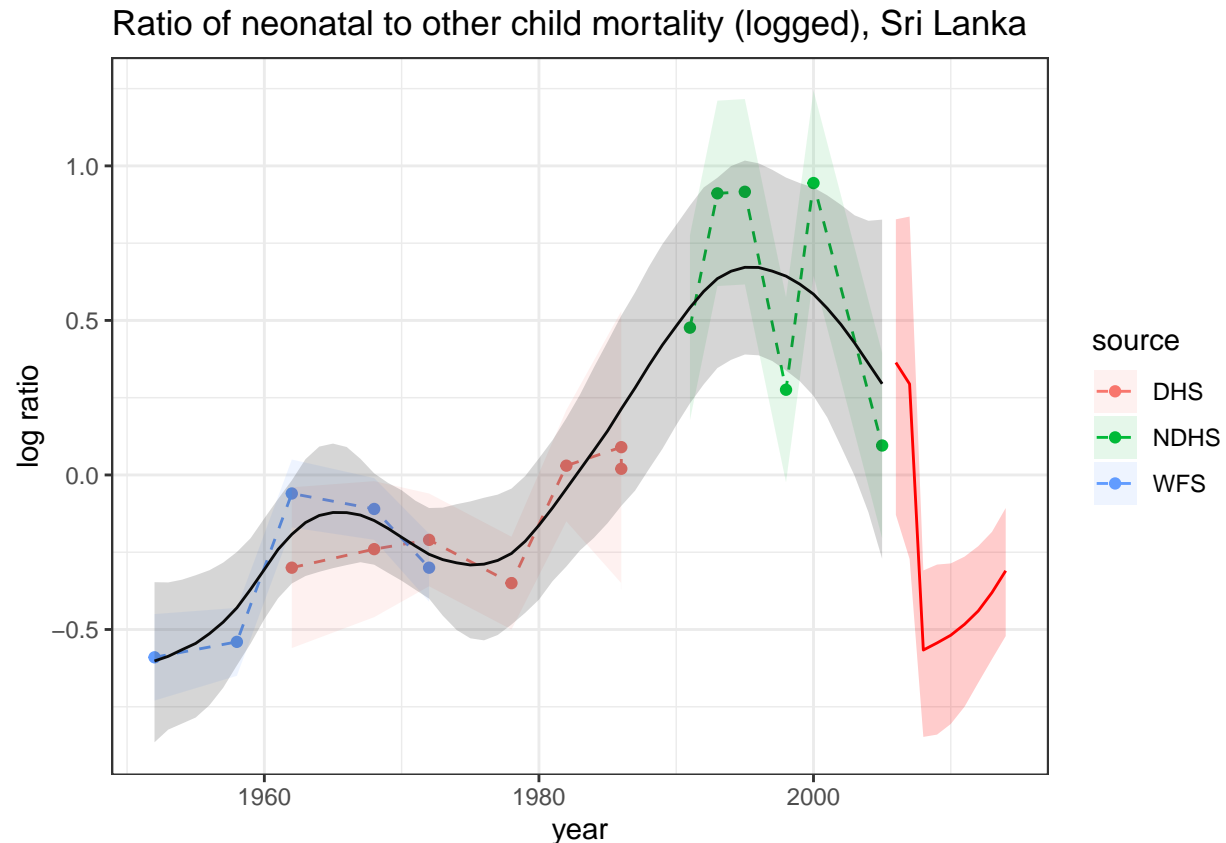
```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```
## Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant:
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#tail-ess
```

```r
res4 <- mod4 %>%
   gather_draws(mu[t]) %>%
  median_qi() %>%
  mutate(year = yearsvr[t])

res_p4 <- mod4 %>%
  gather_draws(mu_p[p]) %>%
  median_qi() %>%
  mutate(year = yearsvr[nyearsvr]+p)
```

```r
ggplot(lkavr, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res4, aes(year, .value)) +
  geom_ribbon(data=res4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2) +
  geom_line(data = res_p4, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio")
```

Ratio of neonatal to other child mortality (logged), Sri Lanka

The VR source is the only one that has the most recent observation in 2014. If we exclude VR, the next most recent observation is from 2005. Since the second order random walk projection depends on the most recent observation and second most recent observation, excluding VR will make the projections be based off older observations, and potentially be less accurate. We see that the projected 2023 value is lower if we exclude the VR source.

Question 6

Briefly comment on which model you think is most appropriate, or an alternative model that would be more appropriate in this context.

We run the random walk order 1 model on the data excluding VR.

```
mod5 <- stan(data = stan_datavr,
             file = here("lab10_2.stan"))
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\Alice\Desktop\GRAD
## SCHOOL\STA2201\STA2201AppliedStatAliceHuang\lab10_2.stan'
```

```
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 6.8e-05 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.68 seconds.
```

```
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.843 seconds (Warm-up)
## Chain 1:                0.713 seconds (Sampling)
## Chain 1:                1.556 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 1.7e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.17 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.415 seconds (Warm-up)
## Chain 2:                0.45 seconds (Sampling)
## Chain 2:                0.865 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 1.5e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.15 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
```

```
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.458 seconds (Warm-up)
## Chain 3:                0.494 seconds (Sampling)
## Chain 3:                0.952 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 2.9e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.29 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.766 seconds (Warm-up)
## Chain 4:                0.459 seconds (Sampling)
## Chain 4:                1.225 seconds (Total)
## Chain 4:

## Warning: There were 3 chains where the estimated Bayesian Fraction of Missing Information was low. Se
## https://mc-stan.org/misc/warnings.html#bfmi-low

## Warning: Examine the pairs() plot to diagnose sampling problems

## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```
res5 <- mod5 %>%
    gather_draws(mu[t]) %>%
```

```
  median_qi() %>%
  mutate(year = yearsvr[t])

res_p5 <- mod5 %>%
  gather_draws(mu_p[p]) %>%
  median_qi() %>%
  mutate(year = yearsvr[nyearsvr]+p)
```

```
ggplot(lkavr, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data=res4, aes(year, .value)) +
  geom_ribbon(data=res4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2) +
  geom_line(data = res_p4, aes(year, .value), col = "red") +
  geom_ribbon(data=res_p4, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "red") +
  geom_line(data=res5, aes(year, .value), col="blue") +
  geom_ribbon(data=res5, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "green") +
  geom_line(data = res_p5, aes(year, .value), col = "blue") +
  geom_ribbon(data=res_p5, aes(y= .value, ymin = .lower, ymax = .upper), alpha = 0.2, fill = "green") +
  labs(title = "Ratio of neonatal to other child mortality (logged), Sri Lanka", y = "log ratio")
```


Ratio of neonatal to other child mortality (logged), Sri Lanka
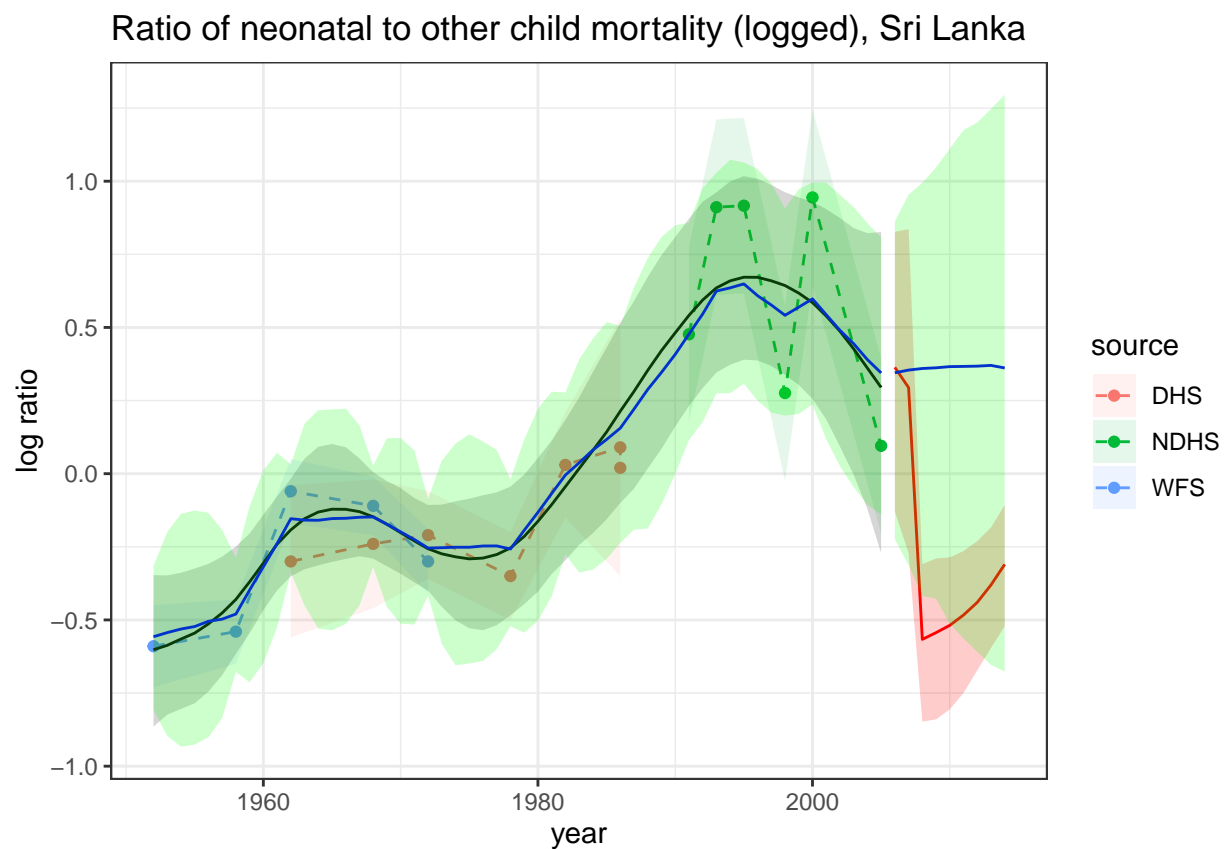
I think the second order random walk model is more appropriate for data excluding the VR source. Since

the first order random walk projection depends on the most recent observation, excluding VR will make the projections be based off only one observation from 2005. The first order random walk projection assumes the projected values will be similar to the 2005 values. However the second order random walk projection will depend on the observations from 2005 and 2000 so perhaps it captures the trend better. The second order random walk model gives more reasonable point estimates