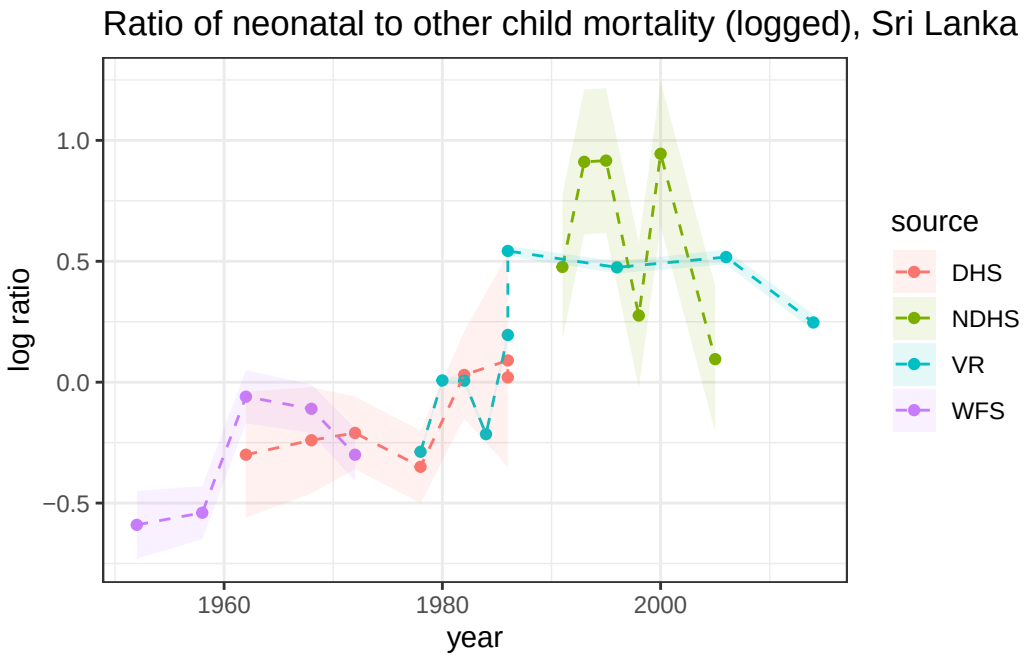
Week 10: Temporal data

25/03/24

Child mortality in Sri Lanka

```
library(tidyverse)
library(here)
library(rstan)
library(tidybayes)
library(janitor)

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



Fitting a linear model

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

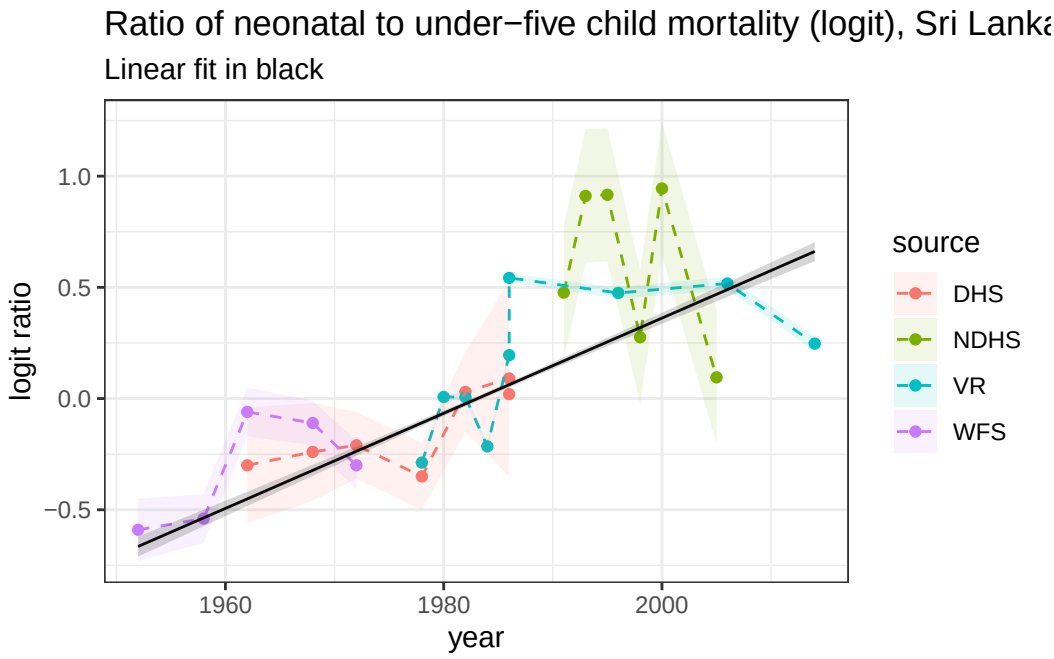
```
observed_years <- lka$year
years <- min(observed_years):max(observed_years)
nyears <- length(years)

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, P = 8)

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

Extract the results:

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

Plot the results:

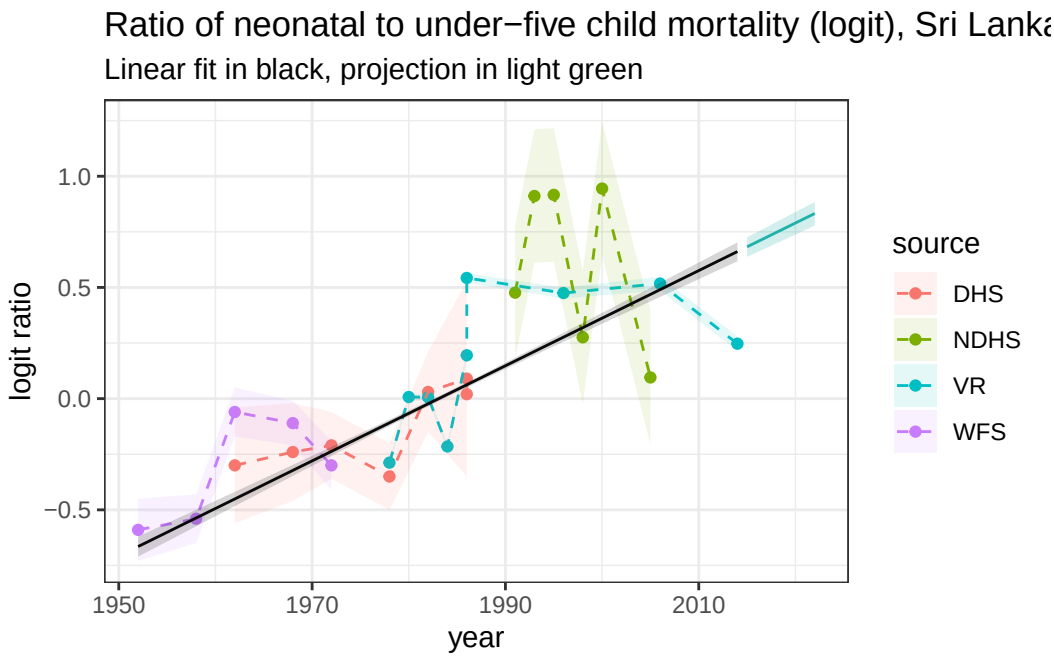
```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data = res, aes(year, .value)) +
  geom_ribbon(data = res, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
  theme_bw()+
  labs(title = "Ratio of neonatal to under-five child mortality (logit), Sri Lanka",
       y = "logit ratio", subtitle = "Linear fit in black")
```



Question 1

```
res_new <- mod |>
  gather_draws(projected_mu[p]) |>
  median_qi() |>
  mutate(year = years[nyears]+p)
```

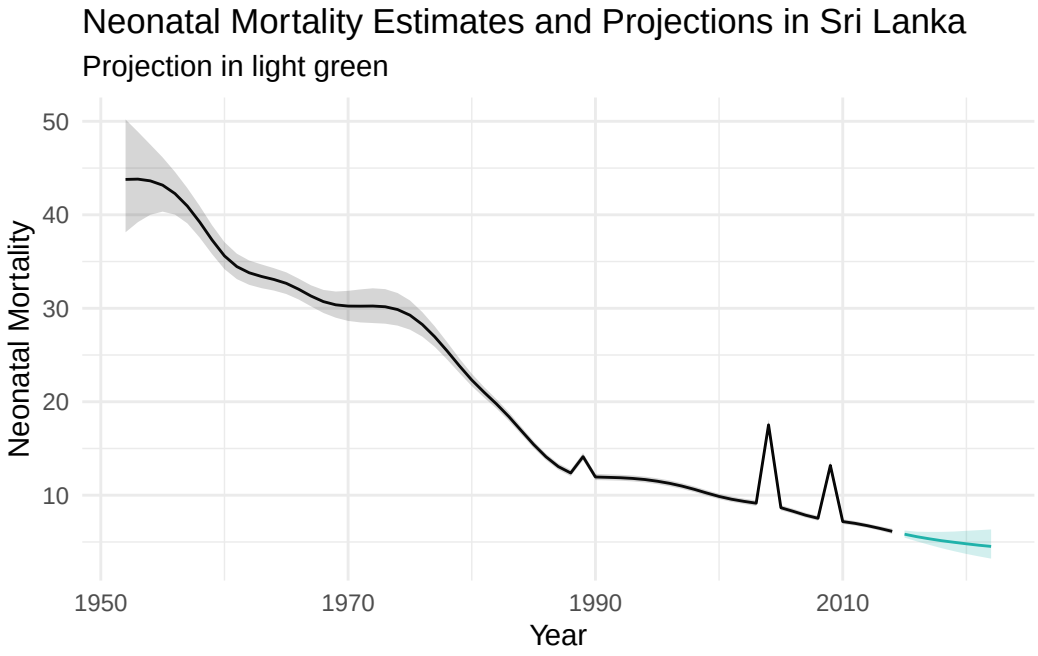
```
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data = res, aes(year, .value)) +
  geom_ribbon(data = res, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
  geom_line(data = res_new, aes(year, .value), col = 'lightseagreen') +
  geom_ribbon(data = res_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2,
  theme_bw()+
  labs(title = "Ratio of neonatal to under-five child mortality (logit), Sri Lanka",
       y = "logit ratio", subtitle = "Linear fit in black, projection in light green")
```



We added a generated quantities block in the stan file for the linear model that generates projections for the years 2014-2022. The added projections can be seen by the red line in the plot

Question 2

```
lka_5 <- read.csv(here("data", "lka_estimated.csv"), skip = 5)
lka_5 <- lka_5 |> filter(Year > 1951) |> mutate(year = Year)
lka_5 <- clean_names(lka_5)

ratio_estimate <- rbind(res %>% select(.value, .lower, .upper, year),
                        res_new %>% select(.value, .lower, .upper, year)) |>
                        mutate(ratio_est = 1/(1 + exp(-.value)),
                               ratio_lower = 1/(1 + exp(-.lower)),
                               ratio_upper = 1/(1 + exp(-.upper))
                               )
estimate <- left_join(lka_5, ratio_estimate, by = "year") |>
                mutate(neo_est = estimate * ratio_est,
                       neo_lower = lower_bound * ratio_lower,
                       neo_upper = upper_bound * ratio_upper)
estimate <- na.omit(estimate)

ggplot(estimate, aes(x = year)) +
  geom_line(data = subset(estimate,year <= 2014),aes(y = neo_est)) +
  geom_ribbon(data = subset(estimate,year <= 2014),aes(ymin = neo_lower, ymax = neo_upper)
  geom_line(data = subset(estimate,year > 2014),aes(y = neo_est), color = "lightseagreen")
  geom_ribbon(data = subset(estimate,year > 2014) ,aes(ymin = neo_lower, ymax = neo_upper)

  labs(title = "Neonatal Mortality Estimates and Projections in Sri Lanka",
       y = "Neonatal Mortality",
       x = "Year", subtitle = "Projection in light green") +
  theme_minimal()
```



Question 3

```
mod1 <- stan(data = stan_data,
             iter = 3000,
             file = here("stan", "FirstOrderRW.stan"))
```

```
Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
using C compiler: 'Apple clang version 14.0.3 (clang-1403.0.22.14.1)'
using SDK: 'MacOSX13.3.sdk'
clang -arch arm64 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG   -I"/Libra
In file included from <built-in>:1:
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/St
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
namespace Eigen {
^
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
namespace Eigen {
                ^
                ;
In file included from <built-in>:1:
```

```
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/St
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
#include <complex>
         ^~~~~~~~~
3 errors generated.
make: *** [foo.o] Error 1

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 2.9e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.29 seconds.
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 1: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 1: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 1: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 1: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 1: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 1: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 1: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 1: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 1: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 1: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 1: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.12 seconds (Warm-up)
Chain 1:                0.127 seconds (Sampling)
Chain 1:                0.247 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 5e-06 seconds
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.05 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 2: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 2: Iteration:  600 / 3000 [ 20%]  (Warmup)
```

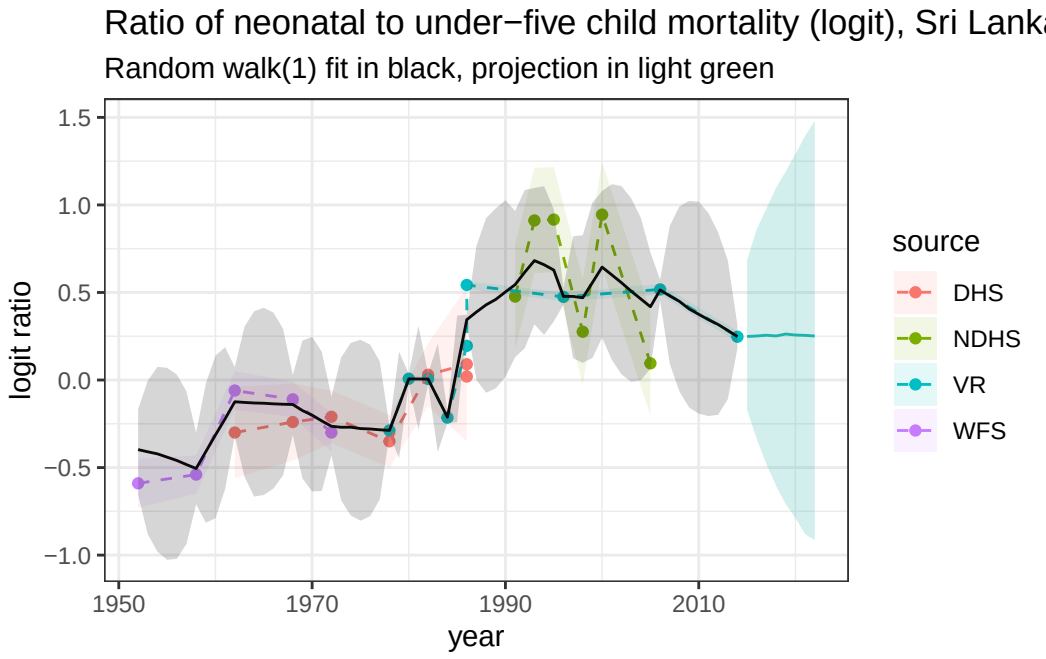
```
Chain 2: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 2: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 2: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 2: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 2: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 2: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 2: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 2: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 2: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.123 seconds (Warm-up)
Chain 2:                0.123 seconds (Sampling)
Chain 2:                0.246 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
Chain 3:
Chain 3: Gradient evaluation took 4e-06 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.04 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 3: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 3: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 3: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 3: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 3: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 3: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 3: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 3: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 3: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 3: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 3: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.132 seconds (Warm-up)
Chain 3:                0.124 seconds (Sampling)
Chain 3:                0.256 seconds (Total)
Chain 3:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
Chain 4:
Chain 4: Gradient evaluation took 4e-06 seconds
```

```
Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.04 seconds.
Chain 4: Adjust your expectations accordingly!
Chain 4:
Chain 4:
Chain 4: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 4: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 4: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 4: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 4: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 4: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 4: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 4: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 4: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 4: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 4: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 4: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 4:
Chain 4:  Elapsed Time: 0.12 seconds (Warm-up)
Chain 4:                0.11 seconds (Sampling)
Chain 4:                0.23 seconds (Total)
Chain 4:
```

```r
res1 <- mod1 |>
  gather_draws(mu[t]) |>
  median_qi() |>
  mutate(year = years[t])

res1_new <- mod1 |>
  gather_draws(projected_mu[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 = res1, aes(year, .value)) +
  geom_ribbon(data = res1, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
```

```
  geom_line(data = res1_new, aes(year, .value), col = 'lightseagreen') +
  geom_ribbon(data = res1_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2,
  theme_bw()+
  labs(title = "Ratio of neonatal to under-five child mortality (logit), Sri Lanka",
       y = "logit ratio", subtitle = "Random walk(1) fit in black, projection in light gre
```



Question 4

```
mod2 <- stan(data = stan_data,
             iter = 3000,
             file = here("stan","SecondOrderRW.stan"))
```

```
Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
using C compiler: ‘Apple clang version 14.0.3 (clang-1403.0.22.14.1)’
using SDK: ‘MacOSX13.3.sdk’
clang -arch arm64 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG   -I"/Libra
In file included from <built-in>:1:
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/St
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
```

```
namespace Eigen {
^
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
namespace Eigen {
               ^
               ;
In file included from <built-in>:1:
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/St
In file included from /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/Rc
/Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/library/RcppEigen/include/Eigen/
#include <complex>
         ^~~~~~~~~
3 errors generated.
make: *** [foo.o] Error 1

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 3.2e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.32 seconds.
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 1: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 1: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 1: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 1: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 1: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 1: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 1: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 1: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 1: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 1: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 1: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.536 seconds (Warm-up)
Chain 1:                0.509 seconds (Sampling)
Chain 1:                1.045 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 8e-06 seconds
```

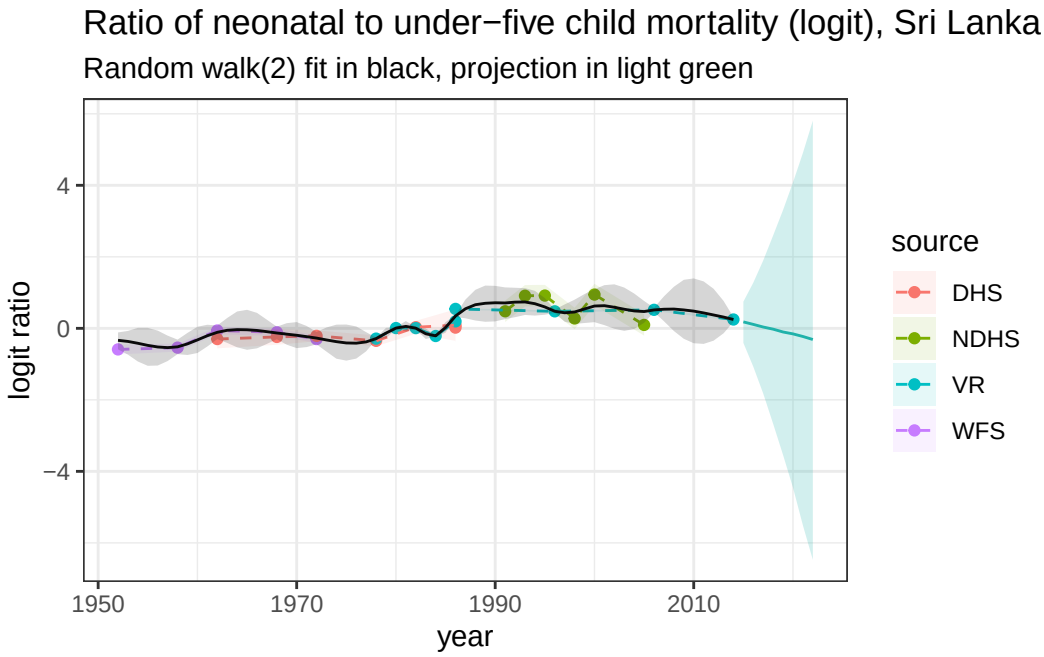
```
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 2: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 2: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 2: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 2: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 2: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 2: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 2: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 2: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 2: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 2: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 2: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.535 seconds (Warm-up)
Chain 2:                0.49 seconds (Sampling)
Chain 2:                1.025 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
Chain 3:
Chain 3: Gradient evaluation took 6e-06 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.06 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:    1 / 3000 [  0%]  (Warmup)
Chain 3: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 3: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 3: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 3: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 3: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 3: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 3: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 3: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 3: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 3: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 3: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.539 seconds (Warm-up)
```

```
Chain 3:                0.566 seconds (Sampling)
Chain 3:                1.105 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 / 3000 [  0%]  (Warmup)
Chain 4: Iteration:  300 / 3000 [ 10%]  (Warmup)
Chain 4: Iteration:  600 / 3000 [ 20%]  (Warmup)
Chain 4: Iteration:  900 / 3000 [ 30%]  (Warmup)
Chain 4: Iteration: 1200 / 3000 [ 40%]  (Warmup)
Chain 4: Iteration: 1500 / 3000 [ 50%]  (Warmup)
Chain 4: Iteration: 1501 / 3000 [ 50%]  (Sampling)
Chain 4: Iteration: 1800 / 3000 [ 60%]  (Sampling)
Chain 4: Iteration: 2100 / 3000 [ 70%]  (Sampling)
Chain 4: Iteration: 2400 / 3000 [ 80%]  (Sampling)
Chain 4: Iteration: 2700 / 3000 [ 90%]  (Sampling)
Chain 4: Iteration: 3000 / 3000 [100%]  (Sampling)
Chain 4:
Chain 4:  Elapsed Time: 0.54 seconds (Warm-up)
Chain 4:                0.498 seconds (Sampling)
Chain 4:                1.038 seconds (Total)
Chain 4:
```

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

res2_new <- mod2 |>
  gather_draws(projected_mu[p]) |>
  median_qi() |>
  mutate(year = years[nyears]+p)
```

```r
ggplot(lka, aes(year, logit_ratio)) +
  geom_point(aes( color = source)) +
```

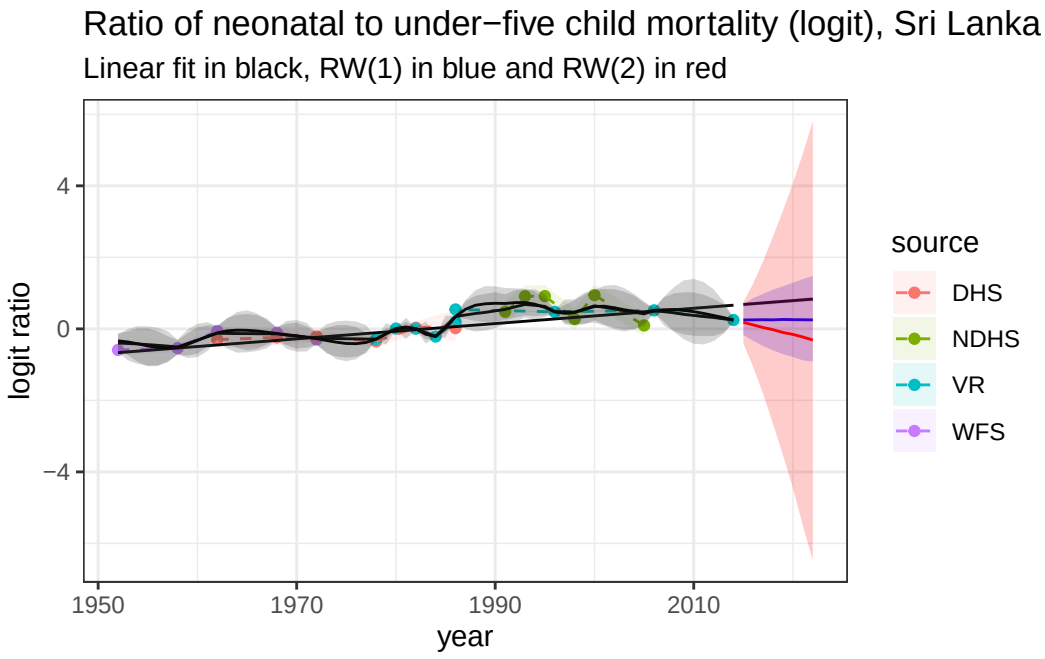
```
  geom_line(aes( color = source), lty = 2) +
  geom_ribbon(aes(ymin = logit_ratio - se,
                  ymax = logit_ratio + se,
                  fill =  source), alpha = 0.1) +
  theme_bw()+
  geom_line(data = res2, aes(year, .value)) +
  geom_ribbon(data = res2, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
  geom_line(data = res2_new, aes(year, .value), col = 'lightseagreen') +
  geom_ribbon(data = res2_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2,
  theme_bw()+
  labs(title = "Ratio of neonatal to under-five child mortality (logit), Sri Lanka",
       y = "logit ratio", subtitle = "Random walk(2) fit in black, projection in light gre
```



Question 5

```
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) +
```

```
geom_line(data = res, aes(year, .value)) +
geom_ribbon(data = res, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
geom_line(data = res_new, aes(year, .value)) +
geom_ribbon(data = res_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+

geom_line(data = res1, aes(year, .value)) +
geom_ribbon(data = res1, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
geom_line(data = res1_new, aes(year, .value), col = 'blue') +
geom_ribbon(data = res1_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2,
geom_line(data = res2, aes(year, .value)) +
geom_ribbon(data = res2, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2)+
geom_line(data = res2_new, aes(year, .value), col = 'red') +
geom_ribbon(data = res2_new, aes(y = .value, ymin = .lower, ymax = .upper), alpha = 0.2,
theme_bw()+
labs(title = "Ratio of neonatal to under-five child mortality (logit), Sri Lanka",
     y = "logit ratio", subtitle = "Linear fit in black, RW(1) in blue and RW(2) in red"
```



Question 6

The plot that we created in question 2 showed that mortality rates have been decreasing significantly over the years. The data with the logit ratio has not clearly been reflecting that however, having a many fluctuations, which resulted in what looks like an increasing trend.

Ideally our model should predict a decrease in mortality rate (equivalently in the log ratio), which the second order random walk model clearly shows the most. Hence that model should be preferred.