Demographic Forecasting

Lecture 3: parametric approaches

Ugofilippo Basellini
basellini@demogr.mpg.de
Max Planck Institute for Demographic Research

March 6, 2024

Course overview

- Lecture 1: direct extrapolation by (generalized) linear models
- Lecture 2: direct extrapolation by time-series methods
- Lecture 3: parametric approaches
- Lecture 4: Lee-Carter method

Parametric methods

Some advantages (see, e.g., Congdon 1993):

- *Smoothness*
- *Parsimony*
- *Interpolation*
- *Comparison*
- **Trends and forecasting**

Parametric methods

- Objective: obtain best fit with the smallest number of parameters
- Trade-off:
  - more parameters, better fit
  - more parameters, less statistical stability (overparameterization)

Fertility parametric methods

- Hadwiger (1940): $f_x = \frac{ab}{c}\left(\frac{c}{x}\right)^{3/2} e^{-b^2\left(\frac{c}{x}+\frac{x}{c}-2\right)}$
- Chandola et al. (1999):

$$
\begin{aligned}
f_x = & \alpha m \frac{b_1}{c_1}\left(\frac{c_1}{x}\right)^{3/2} e^{-b_1^2\left(\frac{c_1}{x}+\frac{x}{c_1}-2\right)} \\
& + (1-m)\frac{b_2}{c_2}\left(\frac{c_2}{x}\right)^{3/2} e^{-b_2^2\left(\frac{c_2}{x}+\frac{x}{c_2}-2\right)}
\end{aligned}
$$

- Peristera and Kostaki (2007):

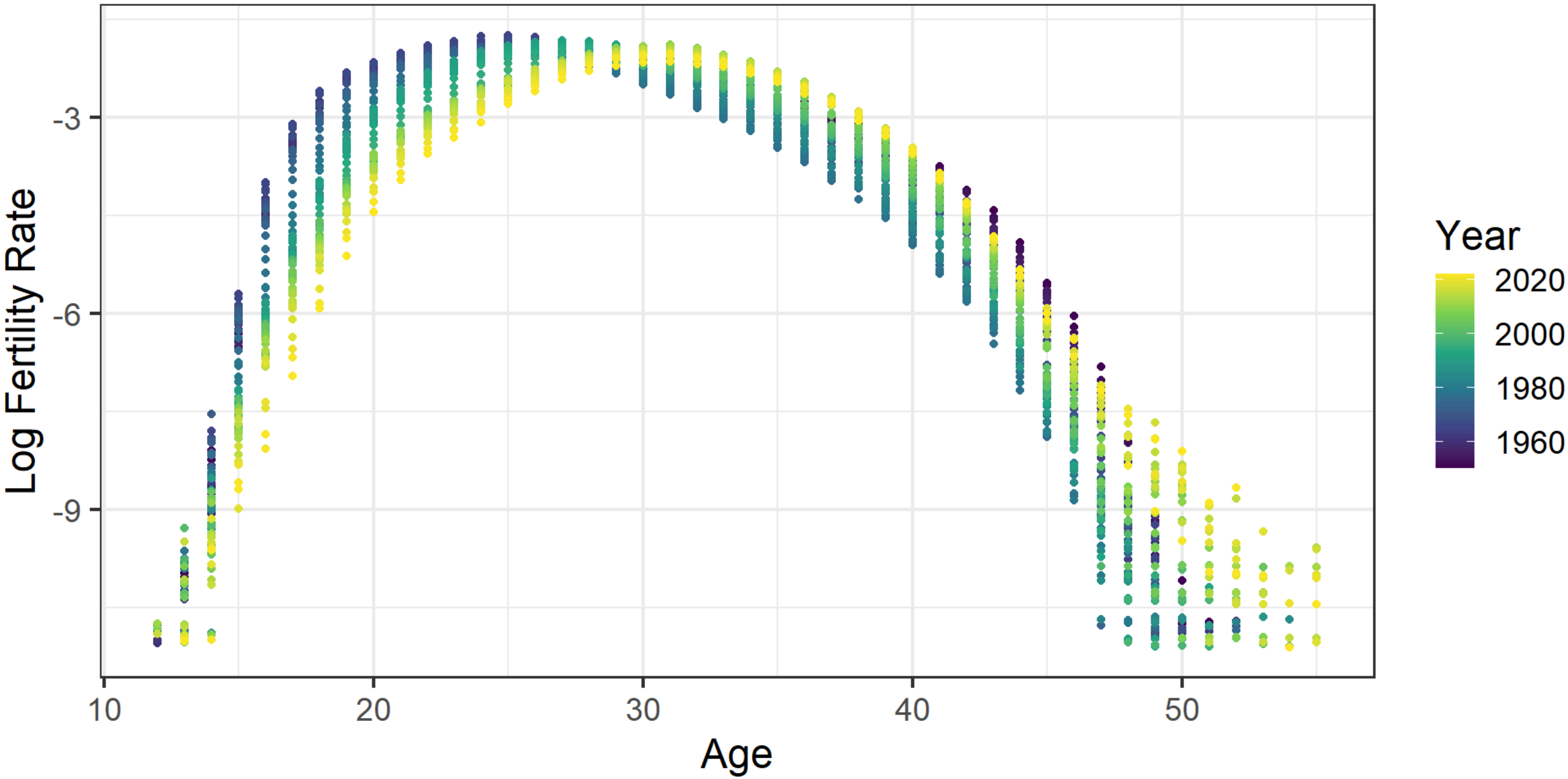
$$
f_x = c_1 e^{-\left(\frac{x-\mu}{\sigma_x}\right)^2}
$$

with $\sigma_x = \sigma_{1x}$ for $x \leq \mu$ and $\sigma_x = \sigma_{2x}$ for $x > \mu$

Mortality parametric methods

- Adult mortality (typically $x \geq 30$):
  - Gompertz (1825): $m_x = e^{a+bx}$
  - Makeham (1860): $m_x = c + e^{a+bx}$
  - Perks (1932): $m_x = c + \frac{e^{a+bx}}{1+e^{\alpha+bx}}$
- Overall mortality:
  - Thiele (1871): $m_x = a_1 e^{-b_1 x} + a_2 e^{-\frac{1}{2} b_2^2 (x-c)^2} + a_3 e^{b_3 x}$
  - Siler (1979): $m_x = a_1 e^{-b_1 x} + a_2 + a_3 e^{b_3 x}$ (for animals, but used in demography - see, e.g., Canudas-Romo and Schoen (2005))
  - Heligman and Pollard (1980): $\frac{q_x}{1-q_x} = A^{(x+B)^C} + De^{-E(\ln(x)-\ln(F))^2} + GH^x$

A simple parametric model for fertility



A simple parametric model for fertility

It looks like a simple log-quadratic model could fit the age-pattern of fertility rather well:

$$\ln(f_{x,t}) = \beta_{0,t} + \beta_{1,t}x + \beta_{2,t}x^2$$

i.e. we could fit a separate model for all years t and derive time-series for the model's parameters.

Exercise

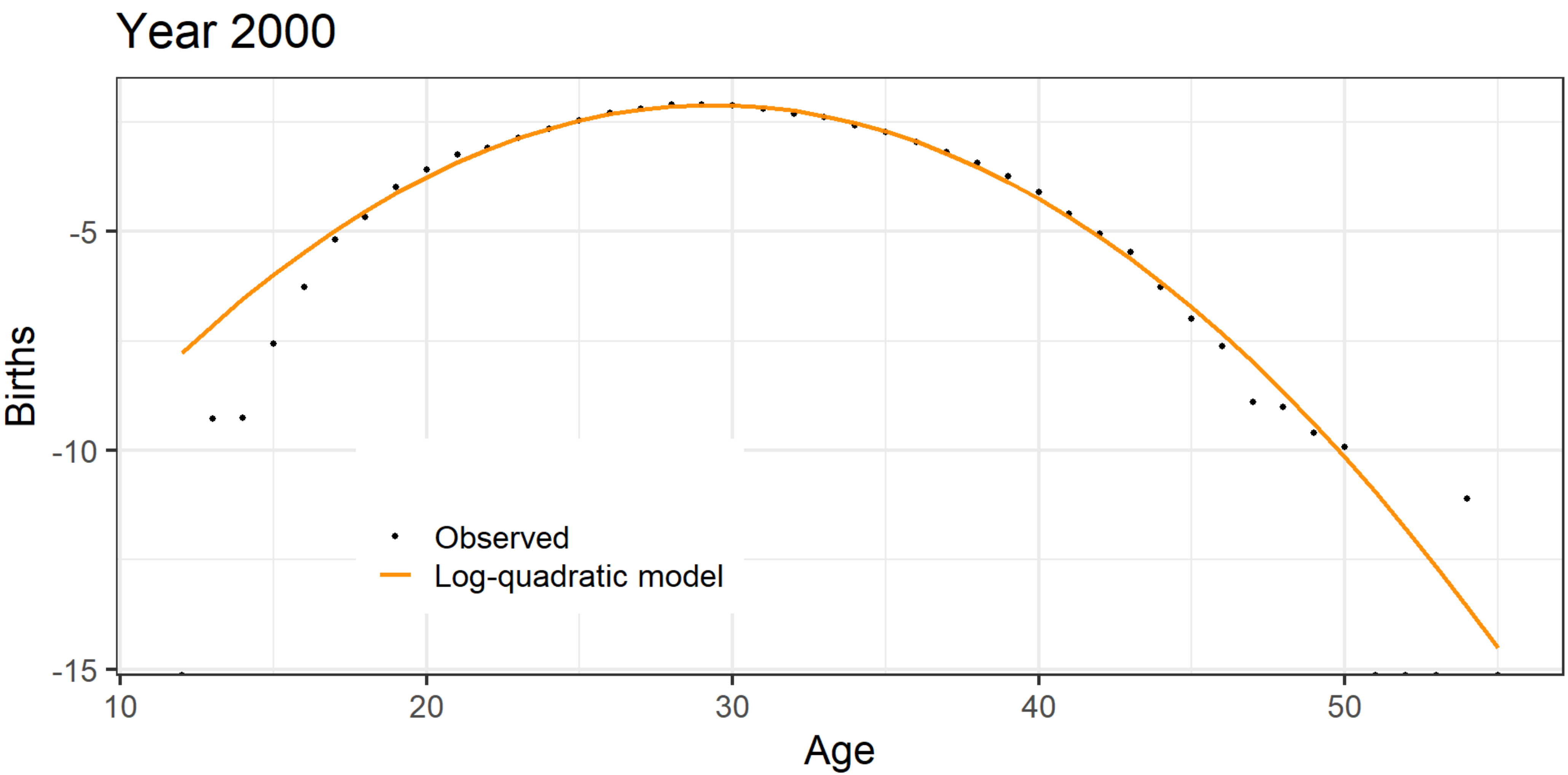
Exercise

Open your `R` session. Load the `FertSWE.Rdata` dataset, and consider only data from 1950 onward. Further, focus on the year 2000, and fit a generalized linear model for births with exposures as an offset using age and age-squared as covariates. Plot the fitted values against the observed log rates.

One possible solution

```
## cleaning the workspace
rm(list=ls(all=TRUE))
## packages
library(tidyverse)
## loading the data
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
load("data/FertSWE.Rdata")
## subset
my.df <- FERT.SWE %>% filter(Year>=1950)
## extracting data
y <- my.df %>% filter(Year==2000) %>% select(Births) %>% pull()
e <- my.df %>% filter(Year==2000) %>% select(Exposures) %>% pull()
lmx <- log(y/e)
x <- unique(my.df$Age)
m <- length(x)
plot(x,log(y/e))
## fitting GLM
x.sq <- x^2
glm1 <- glm(y~x+x.sq,offset = log(e),family=poisson())
```

One possible solution



Exercise

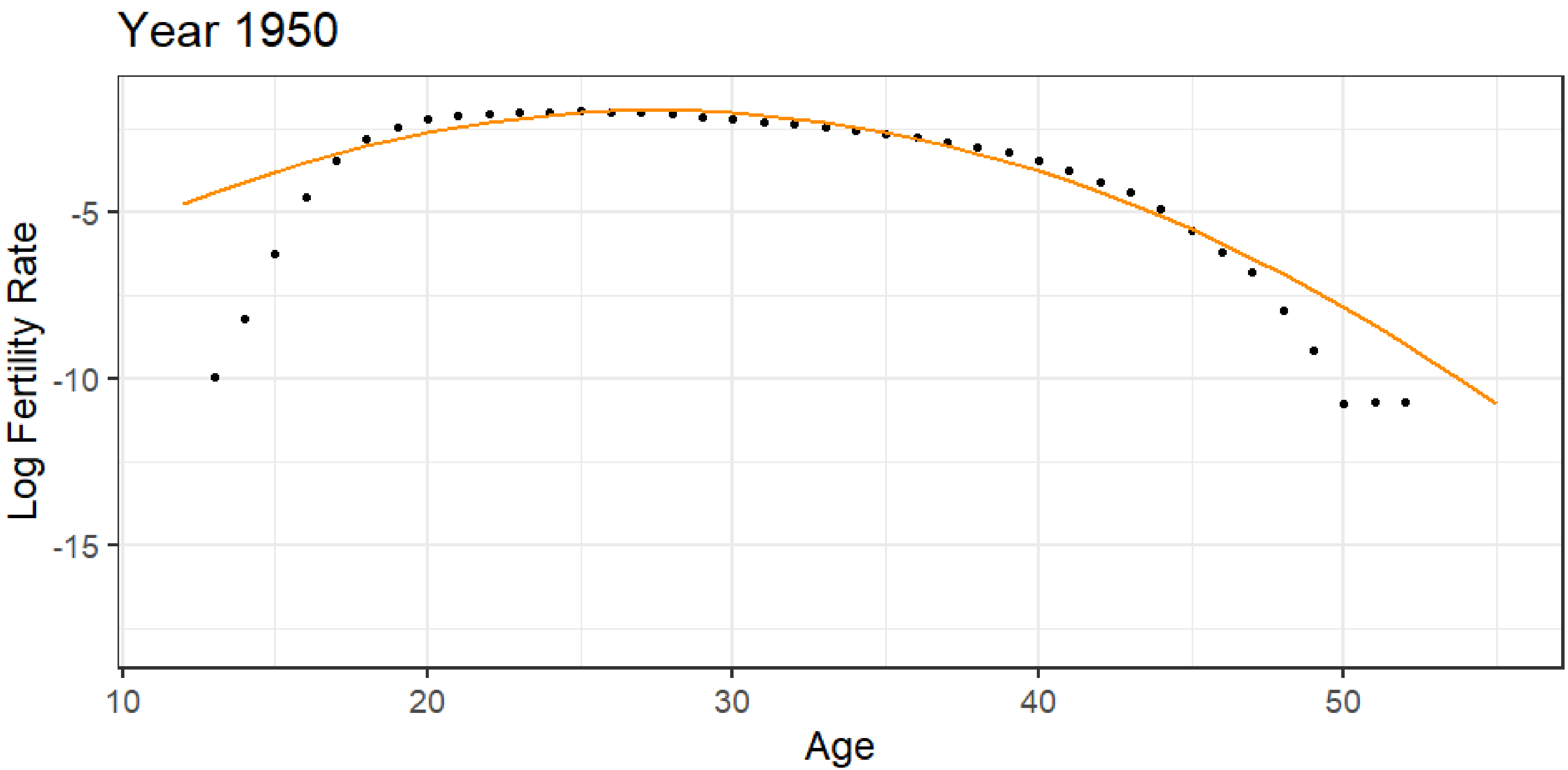
Exercise

Repeat this for all years the dataset (1950-2022), and plot the three time series of the estimated parameters over time.

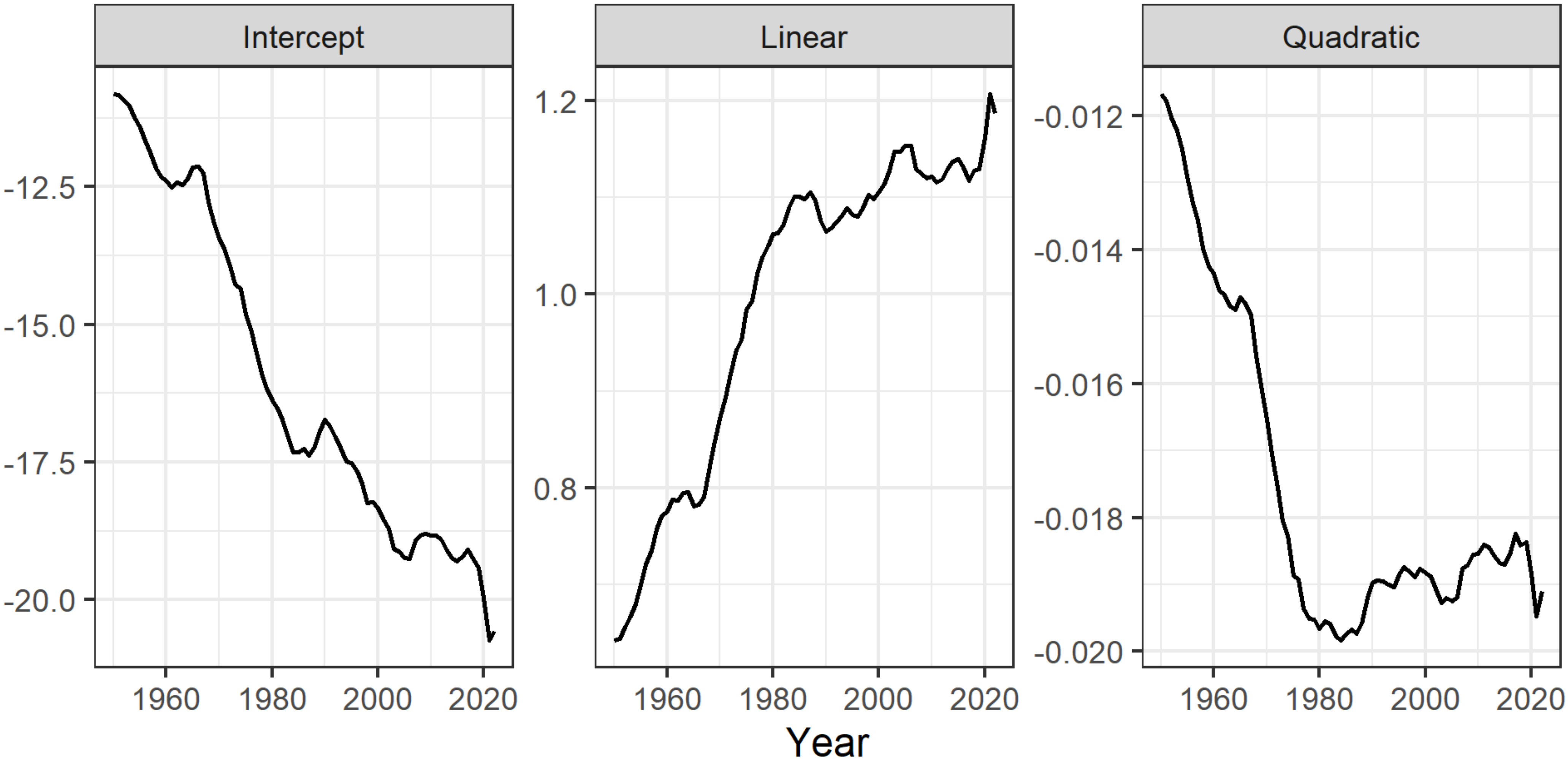
One possible solution

```r
## cleaning the workspace
rm(list=ls(all=TRUE))
## packages
library(tidyverse)
## loading the data
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
load("data/FertSWE.Rdata")
## subset
my.df <- FERT.SWE %>% filter(Year>=1950)
## extracting data
x <- unique(my.df$Age)
t <- unique(my.df$Year)
n <- length(t)
m <- length(x)
## matrices
BIRTHS <- matrix(my.df$Births,m,n)
EXPOS <- matrix(my.df$Exposures,m,n)
RATES <- matrix(my.df$Rates,m,n)
LRATES <- matrix(my.df$logRates,m,n)
```

One possible solution



One possible solution



Exercise

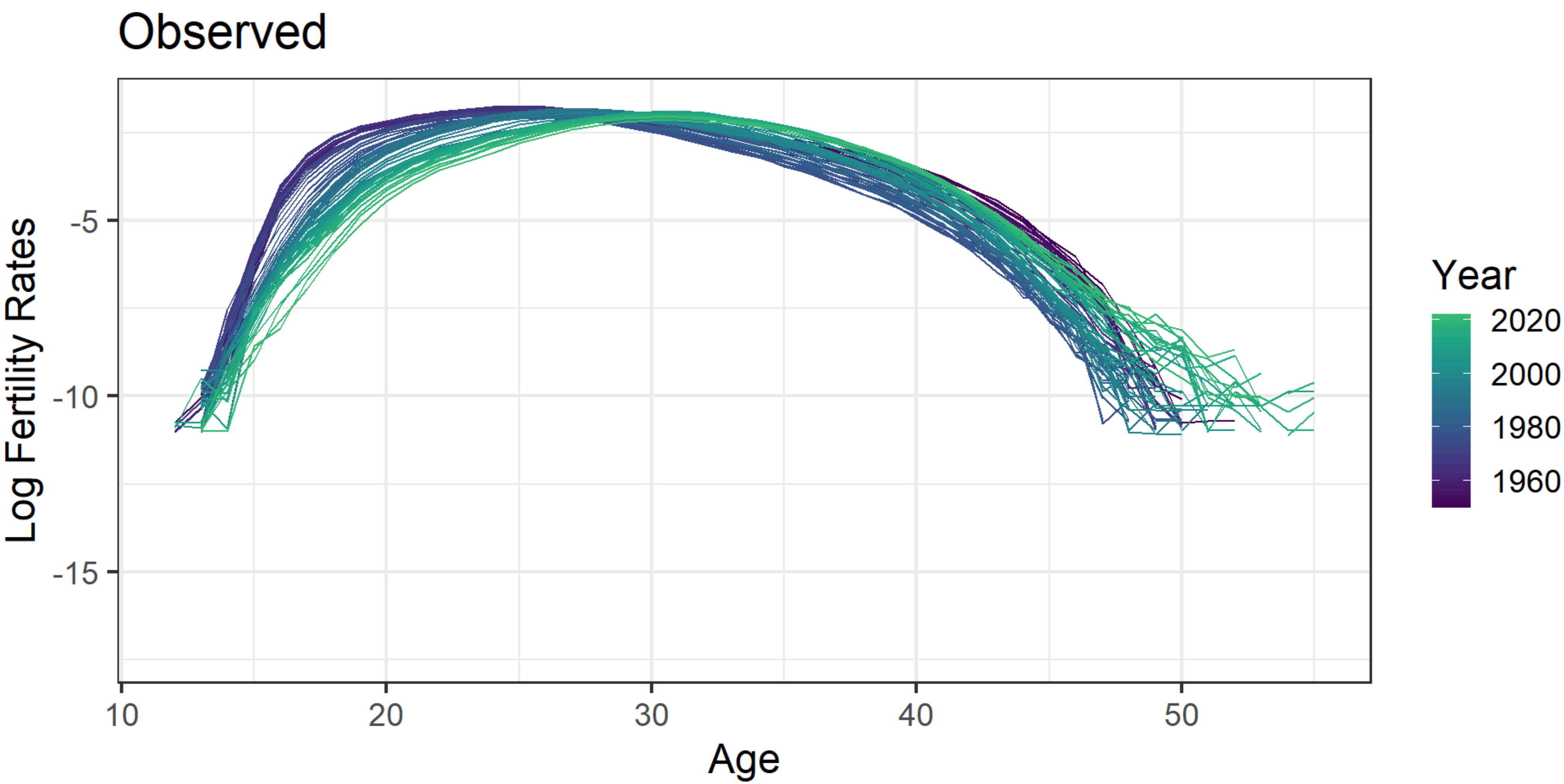
Exercise

Now forecast the three time-series using the most appropriate ARIMA(p,d,q) model, and derive the forecast age-pattern of fertility in 2050.

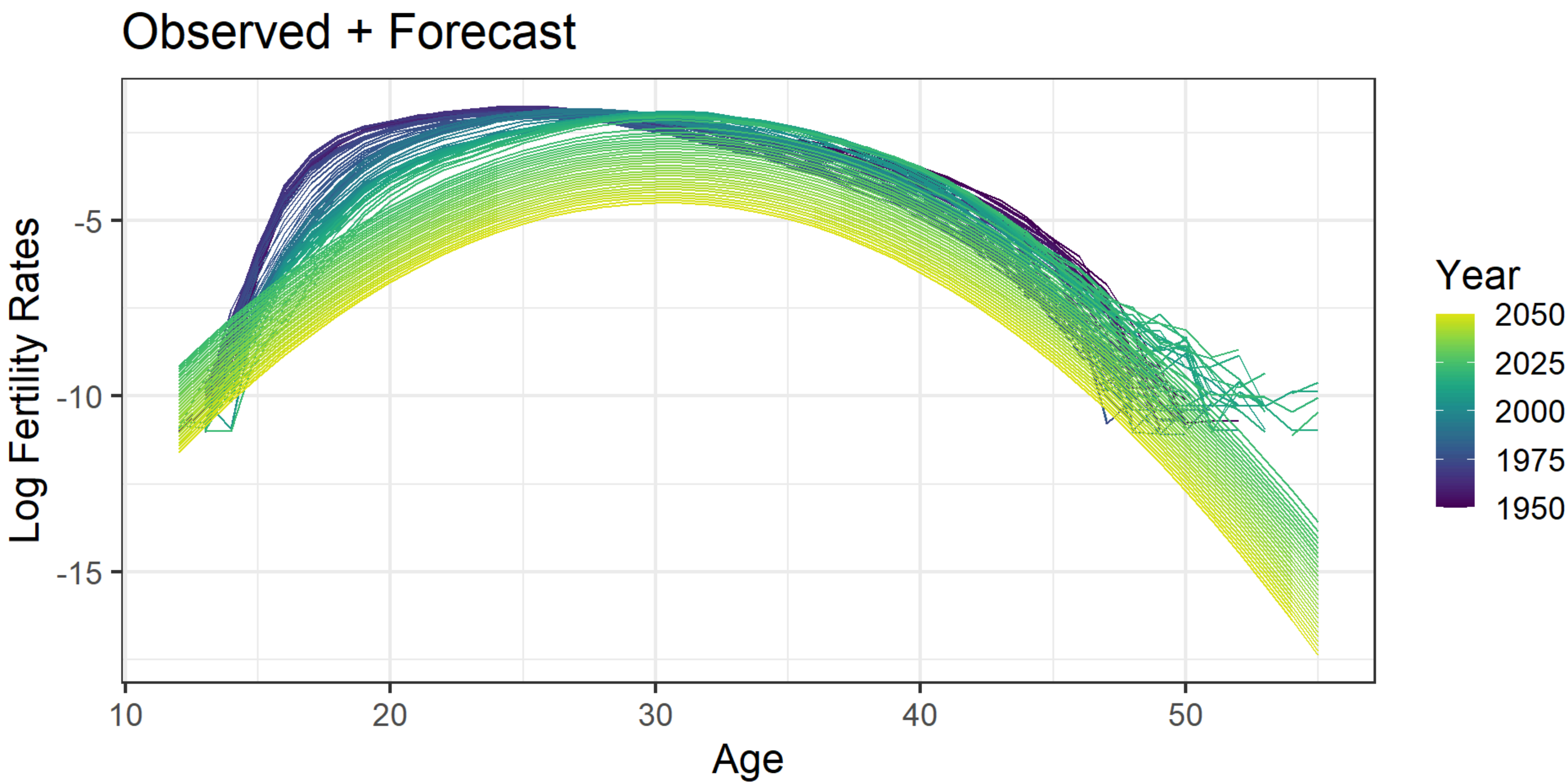
One possible solution

```
## forecast package
library(forecast)
## extracting parameters
y1 <- COEFS[,1]
y2 <- COEFS[,2]
y3 <- COEFS[,3]
## fitting ARIMA models
mod1 <- auto.arima(y1)
mod2 <- auto.arima(y2)
mod3 <- auto.arima(y3)
## forecast
y.fore1 <- forecast(mod1,h=nF)
y.fore2 <- forecast(mod2,h=nF)
y.fore3 <- forecast(mod3,h=nF)
plot(y.fore1)
plot(y.fore2)
plot(y.fore3)
## forecast rates
ETA.fore <- matrix(NA,m,nF)
for (i in 1:nF){
  ETA.fore[,i] <- y.fore1$mean[i]+y.fore2$mean[i]*x +y.fore3$mean[i]*x.sq
}
## plotting
my.cols <- viridis(n+nF)
matplot(x,LRATES,t="l",col=my.cols[1:n],lty=1,ylim=range(LRATES,ETA.fore,finite=T))
matlines(x,ETA.fore,col=my.cols[1:nF+n],lty=1)
```

One possible solution



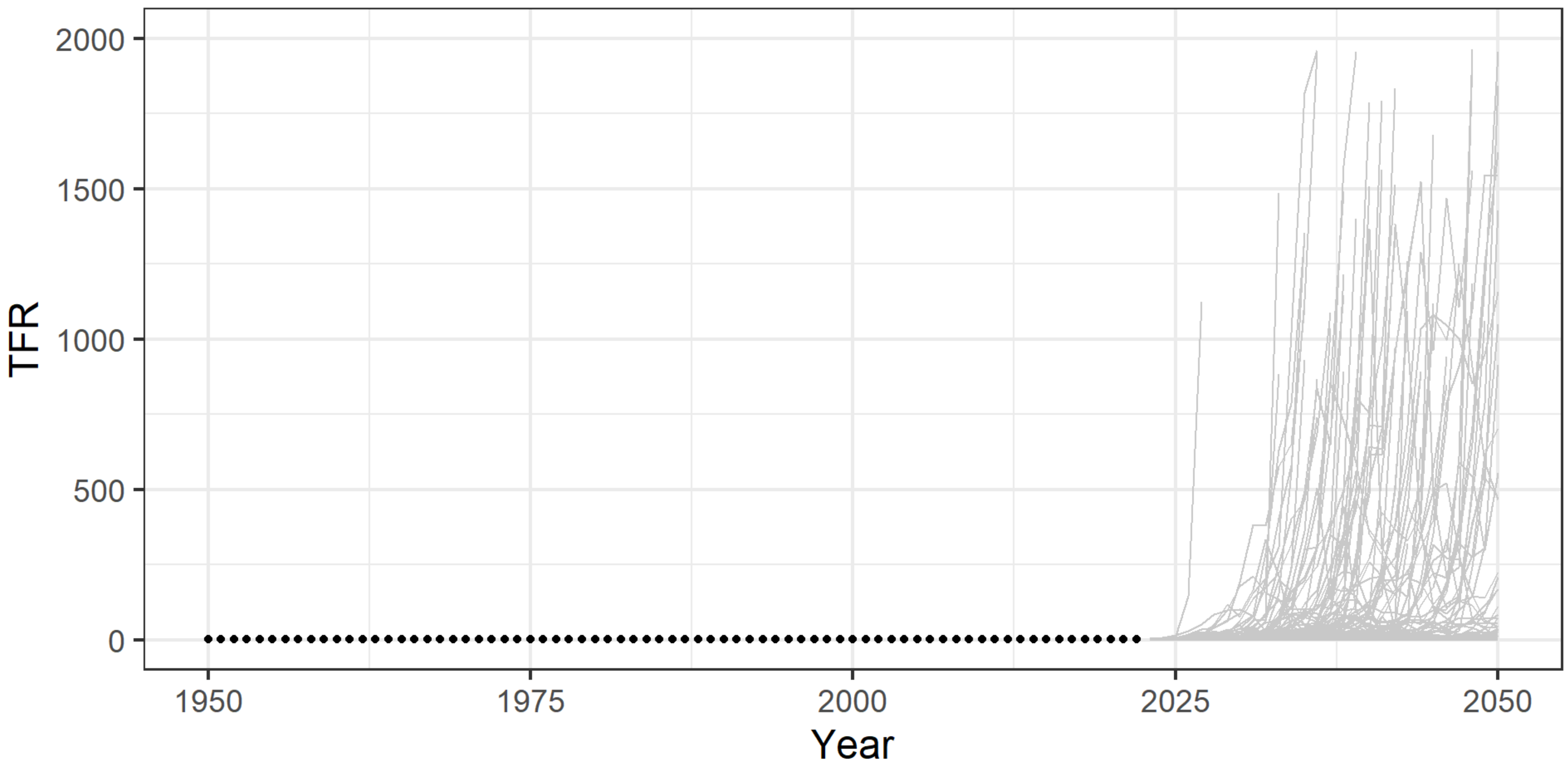
One possible solution



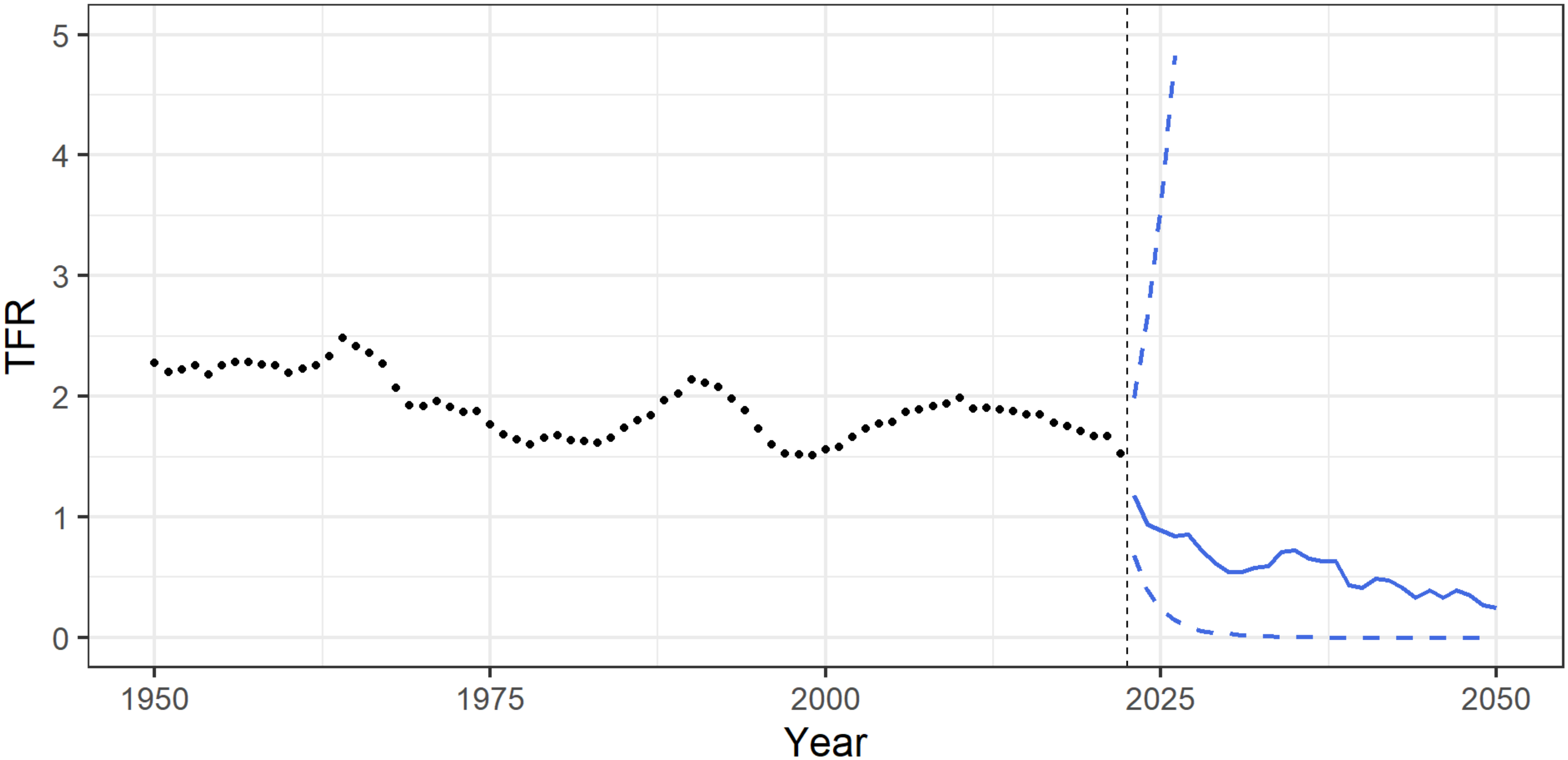
Uncertainty

- We can use the ARIMA simulations for future paths of the coefficients to derive prediction intervals for the age-pattern of fertility as well as for summary measures

TFR simulations



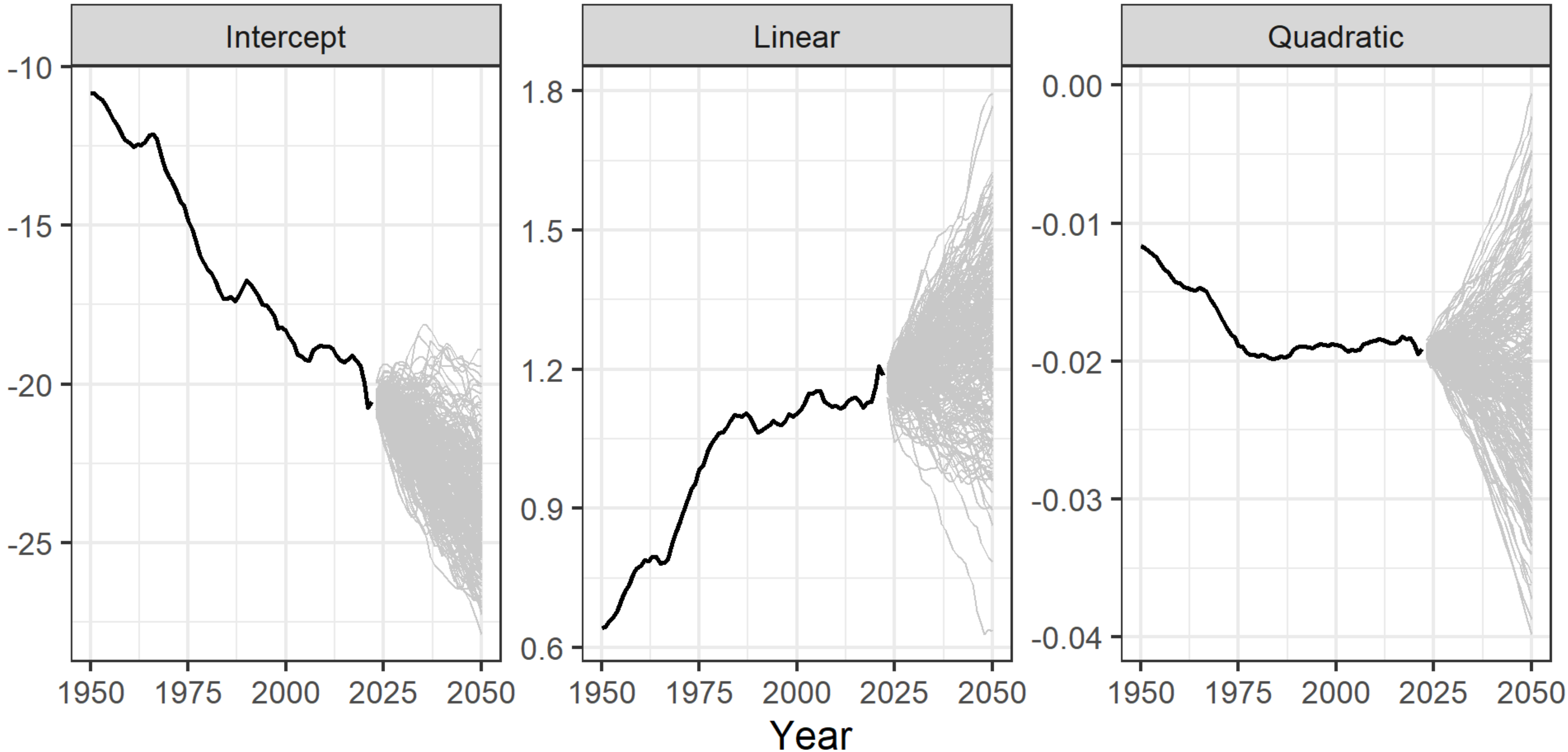
TFR 80% CI



PIs with parametric approach

- uncertainty appears to escalate quickly with forecasting horizon
- Why is that?

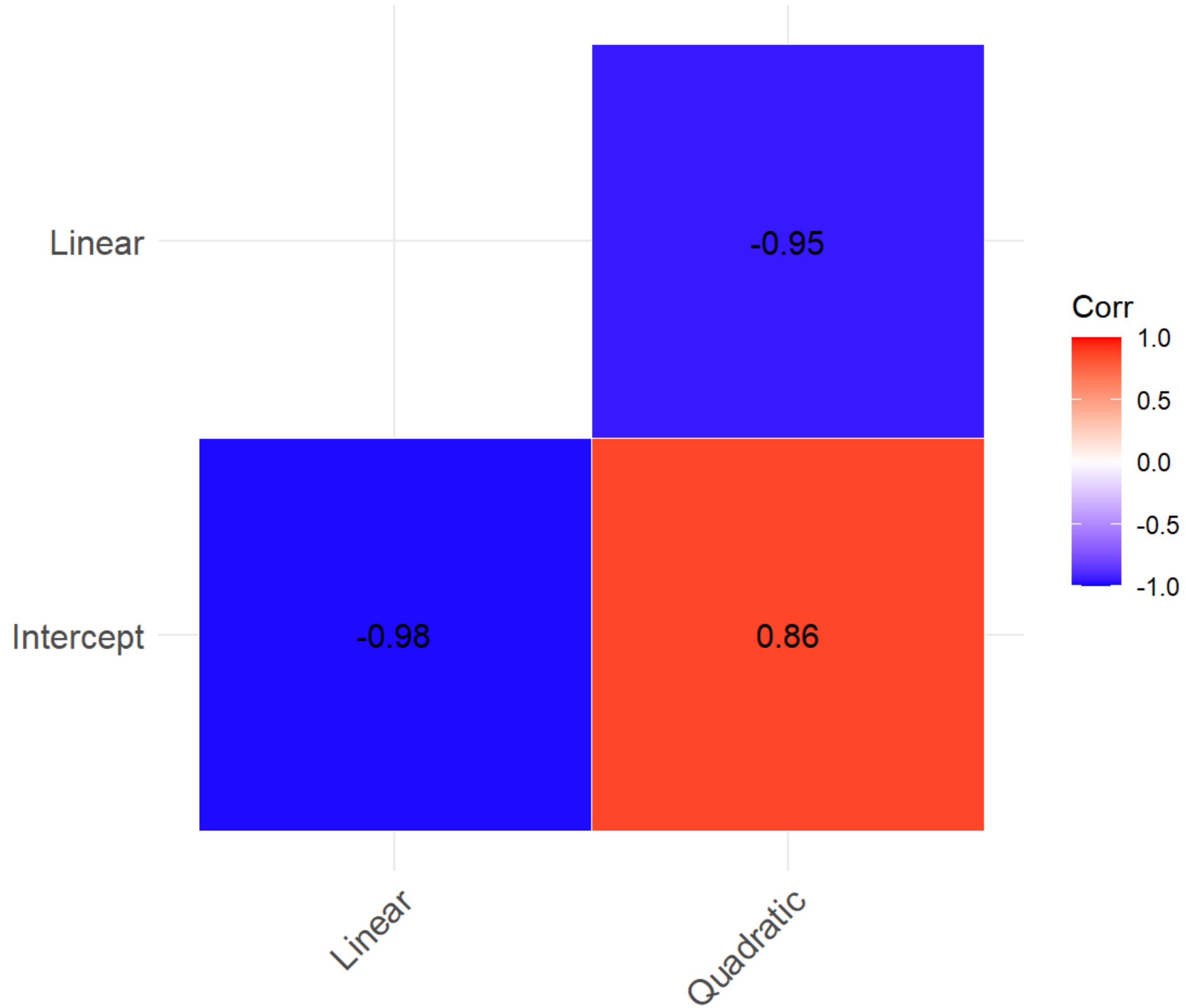
Simulated parameters



PIs with parametric approach

- large uncertainty in the forecast parameters
- where does this stem from?

Parameters’ correlation



Parameters’ correlation

- The time-series of the three estimated parameters are highly correlated between each other
- Yet, we are treating them independently by fitting univariate time-series models
- It would be better to use multivariate time-series methods, or a methodology that is based on a single time-series, like the Lee-Carter method (see tomorrow)

Towards Lee-Carter I

- We could generalize the simple parametric model for fertility to allow for a linear time trend:

$$\ln(f_{x,t}) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 t$$

- Extrapolating the linear time trend can provide us with fertility forecasts

Exercise

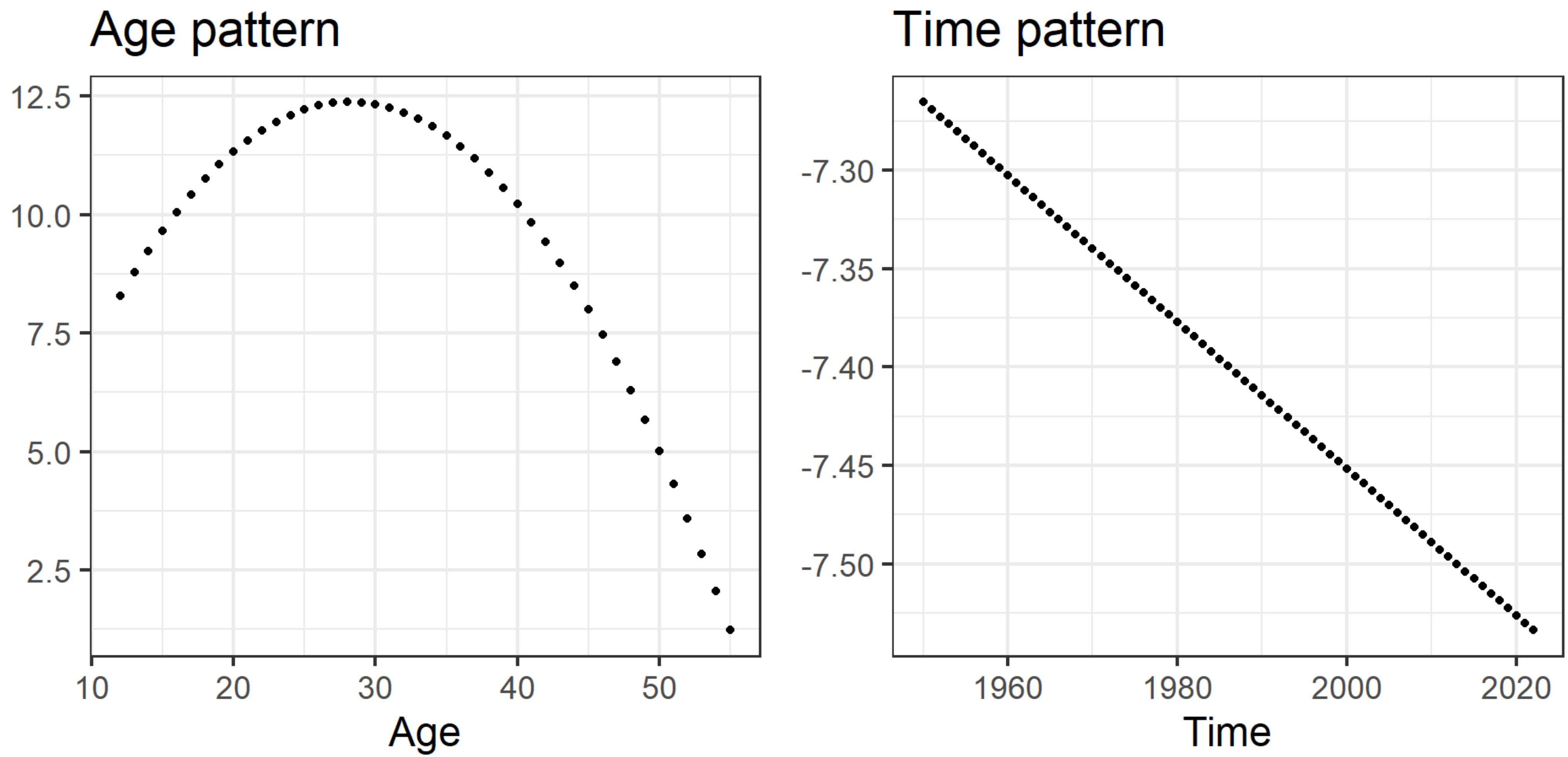
Exercise

Fit a single GLM model to the same data, which includes a quadratic shape for age and a linear trend for time. Extrapolate the linear time index to compute fertility forecasts up to 2050.

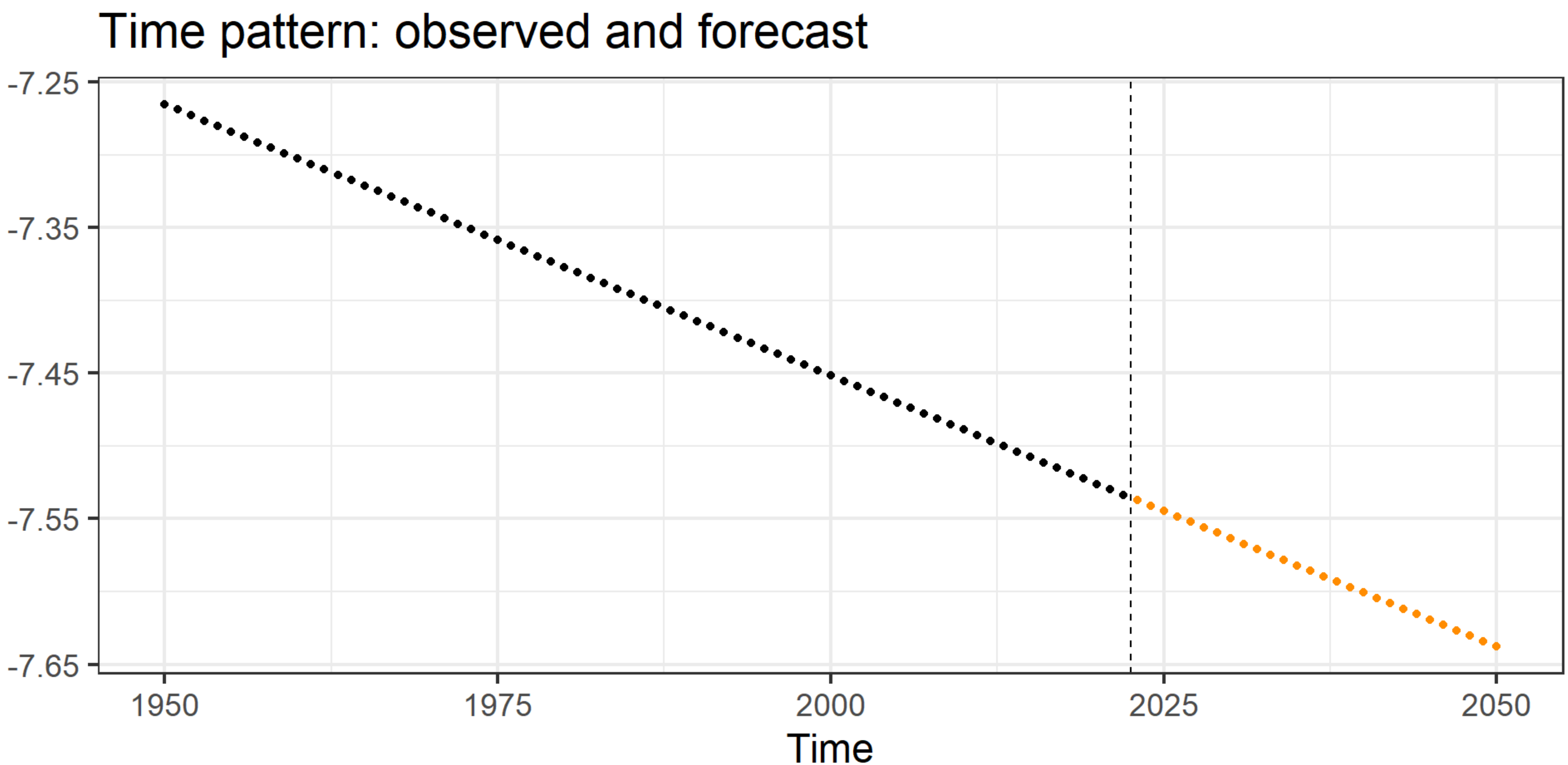
One possible solution

```
## including age squared in our datafrane
my.df <- FERT.SWE %>% filter(Year>=1950) %>%
  mutate(Age.sq=Age^2)
## fitting a single GLM model
glm2 <- glm(Births~Age+Age.sq+Year,family = poisson(),
            offset=log(Exposures),data=my.df)
summary(glm2)
## extracting age and time patterns
age.pattern <- coef(glm2)[2]*x + coef(glm2)[3]*x.sq
plot(x,age.pattern)
time.pattern <- coef(glm2)[4]*t
plot(t,time.pattern)
## extrapolating time pattern
t.all <- c(t,tF)
n.all <- length(t.all)
time.pattern.all <- coef(glm2)[4]*t.all
plot(t.all,time.pattern.all)
points(t,time.pattern,pch=16)
## fitting and forecast rates from single model
```

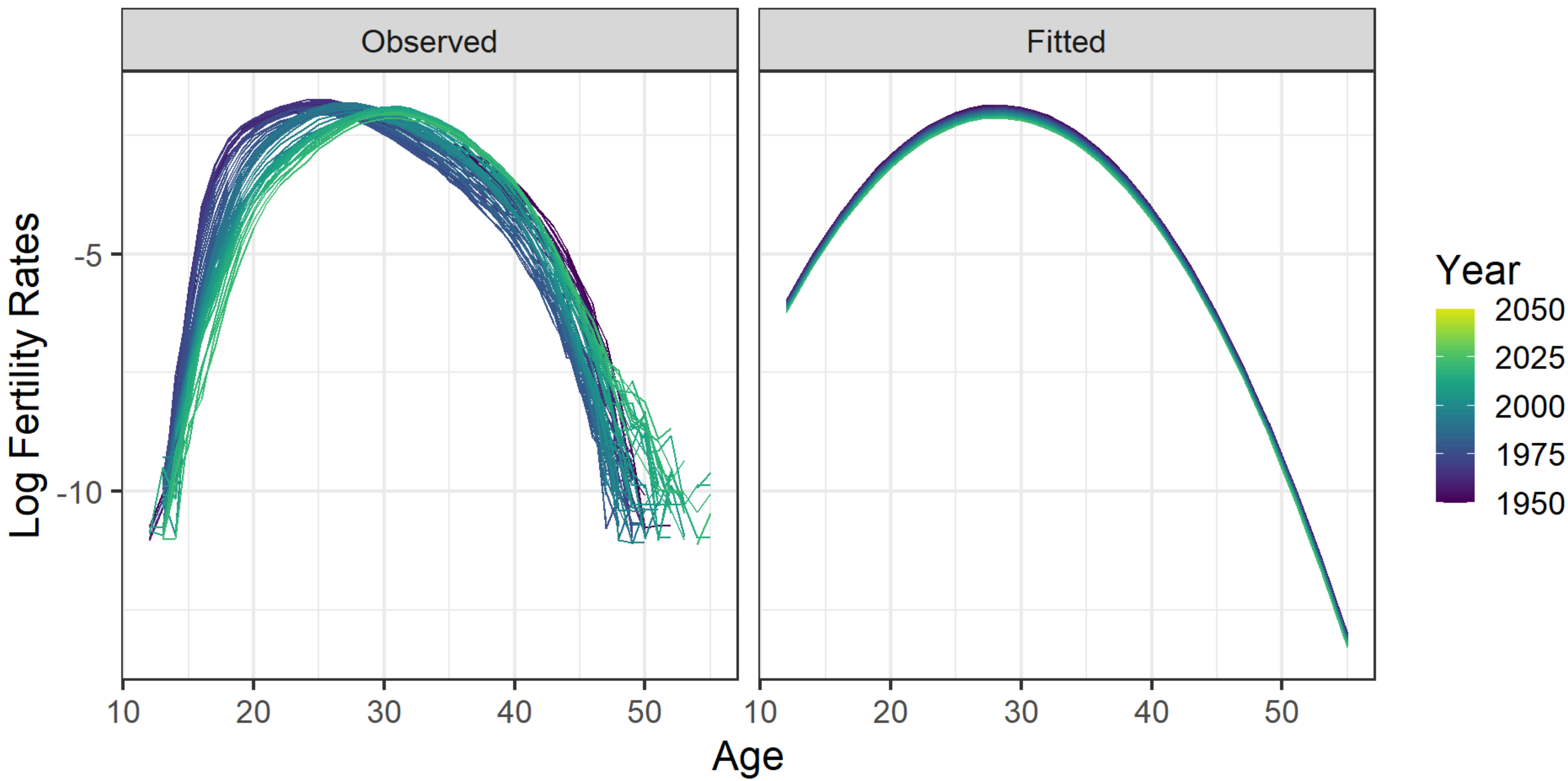
Log-linear time trend



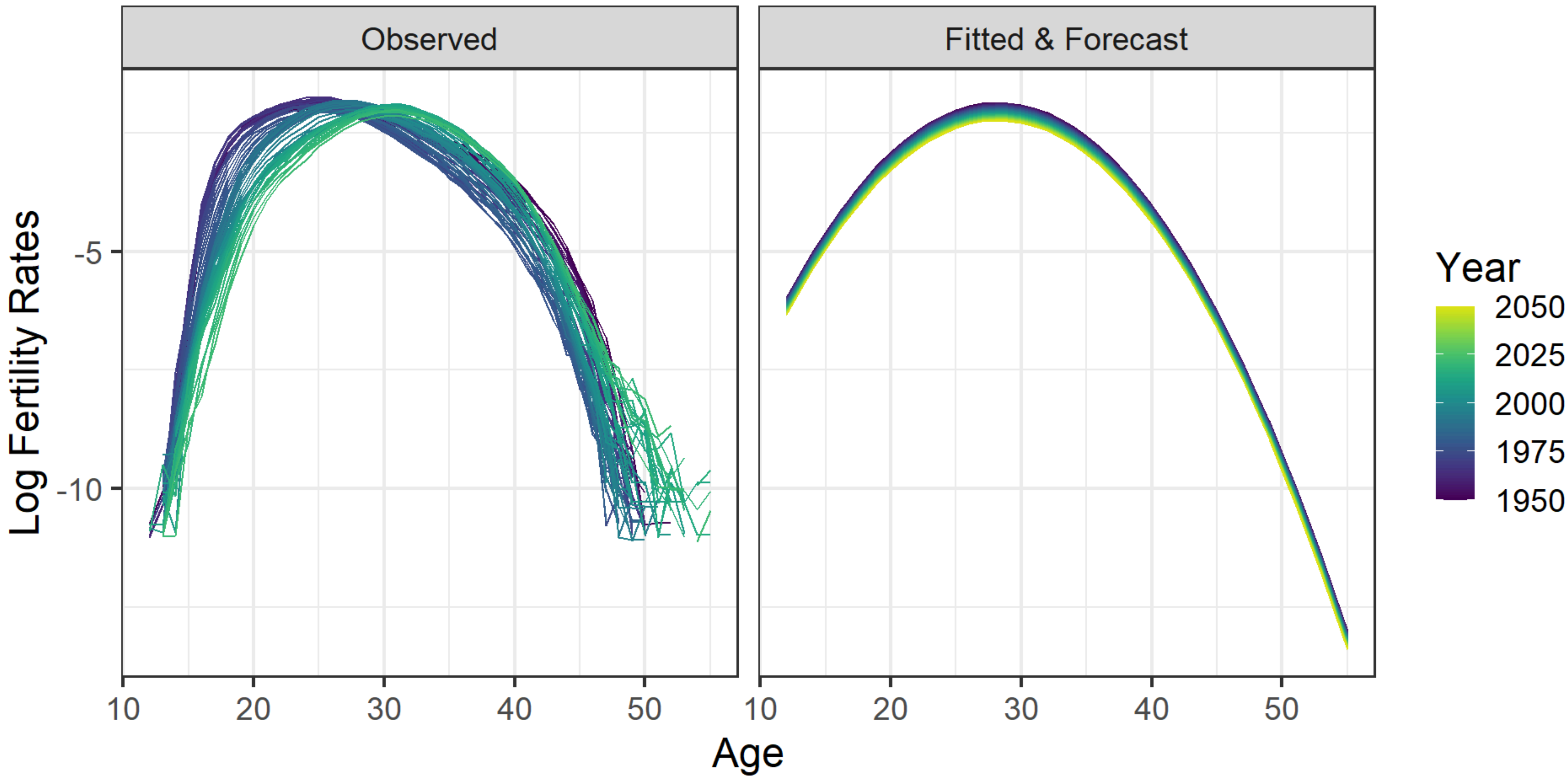
Log-linear time trend



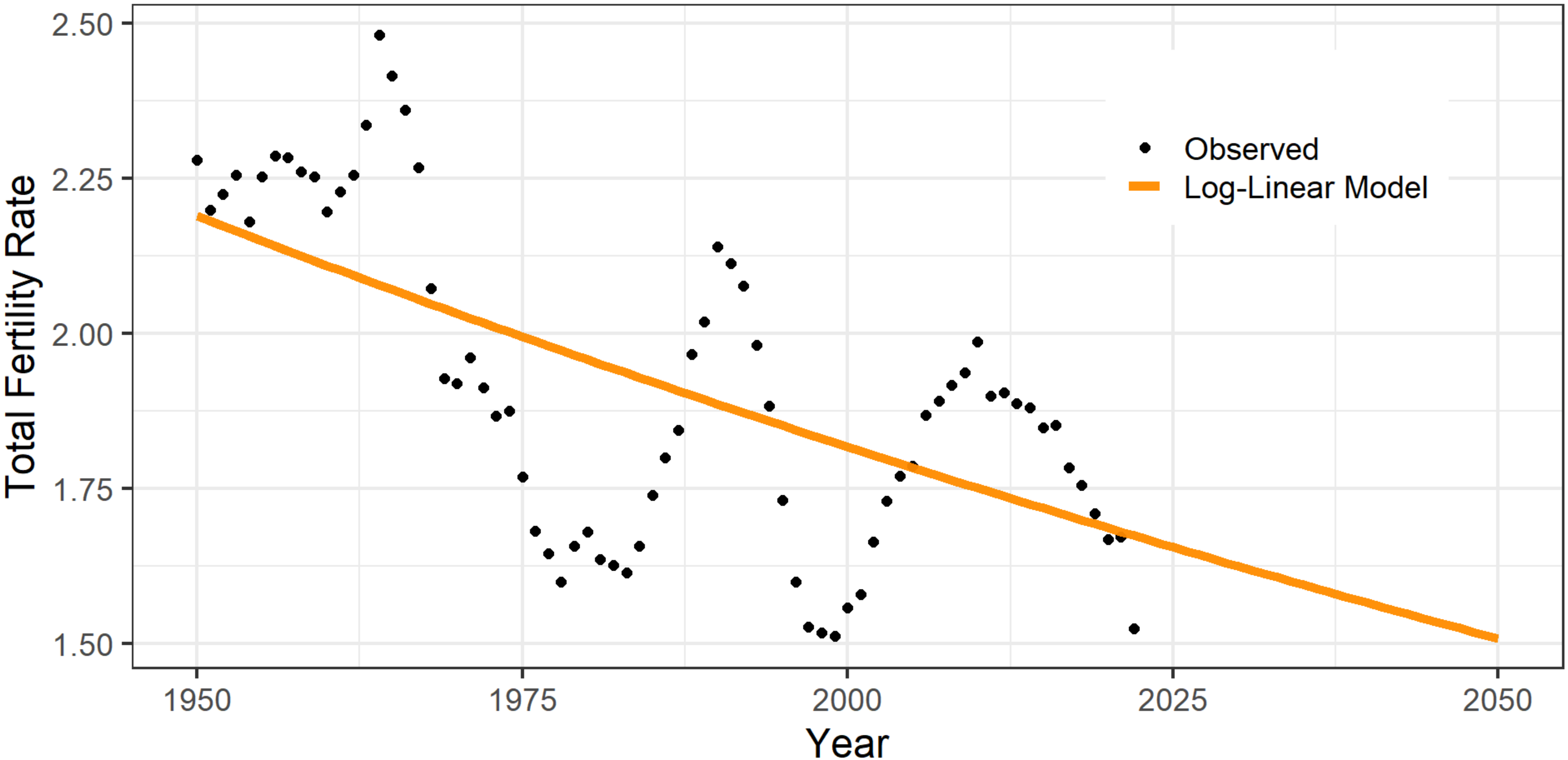
Log-linear time trend



Log-linear time trend



Log-linear time trend



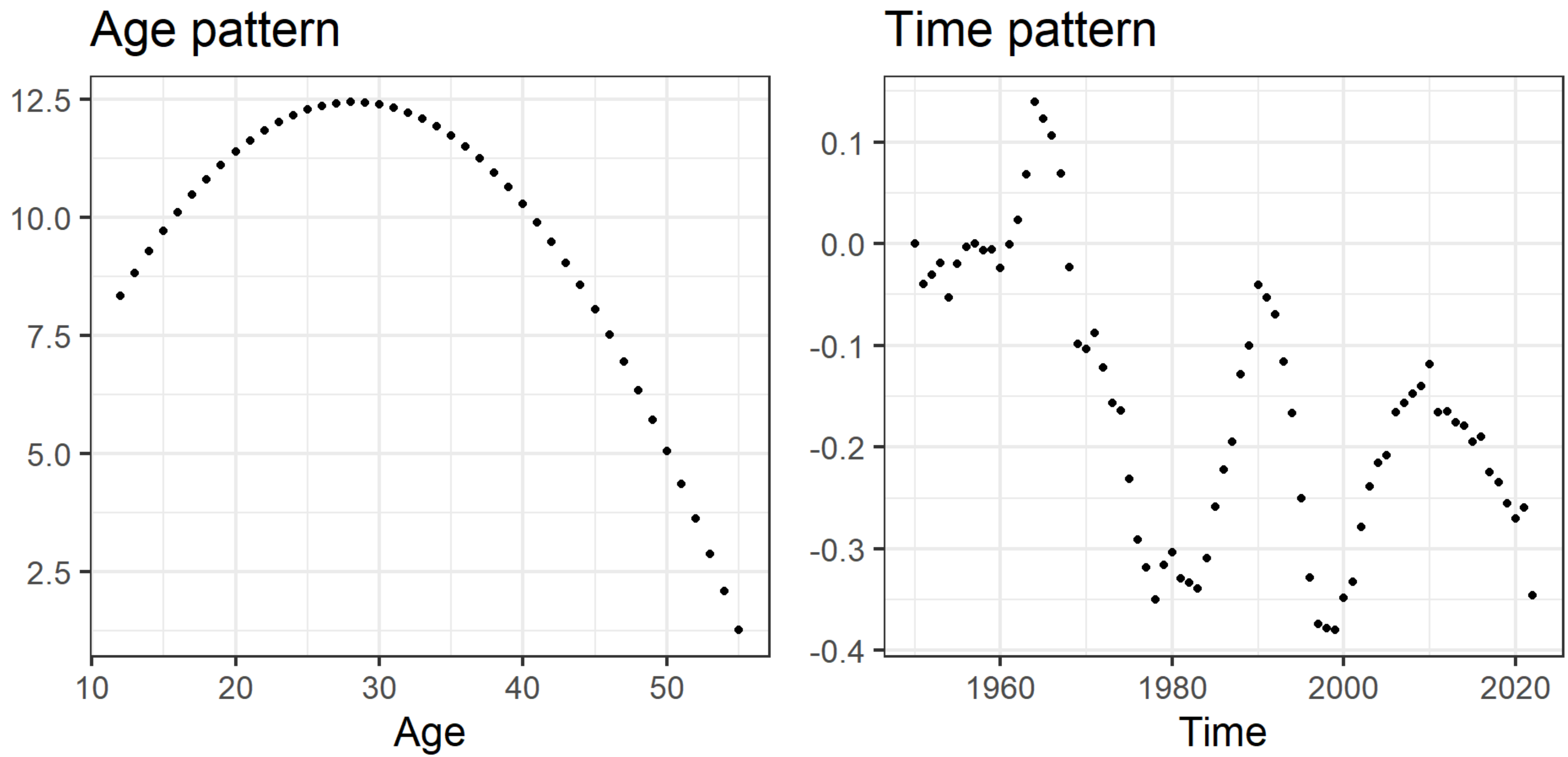
Towards Lee-Carter II

Alternatively, we could relax the linear time trend assumption and estimate one parameter for each year:

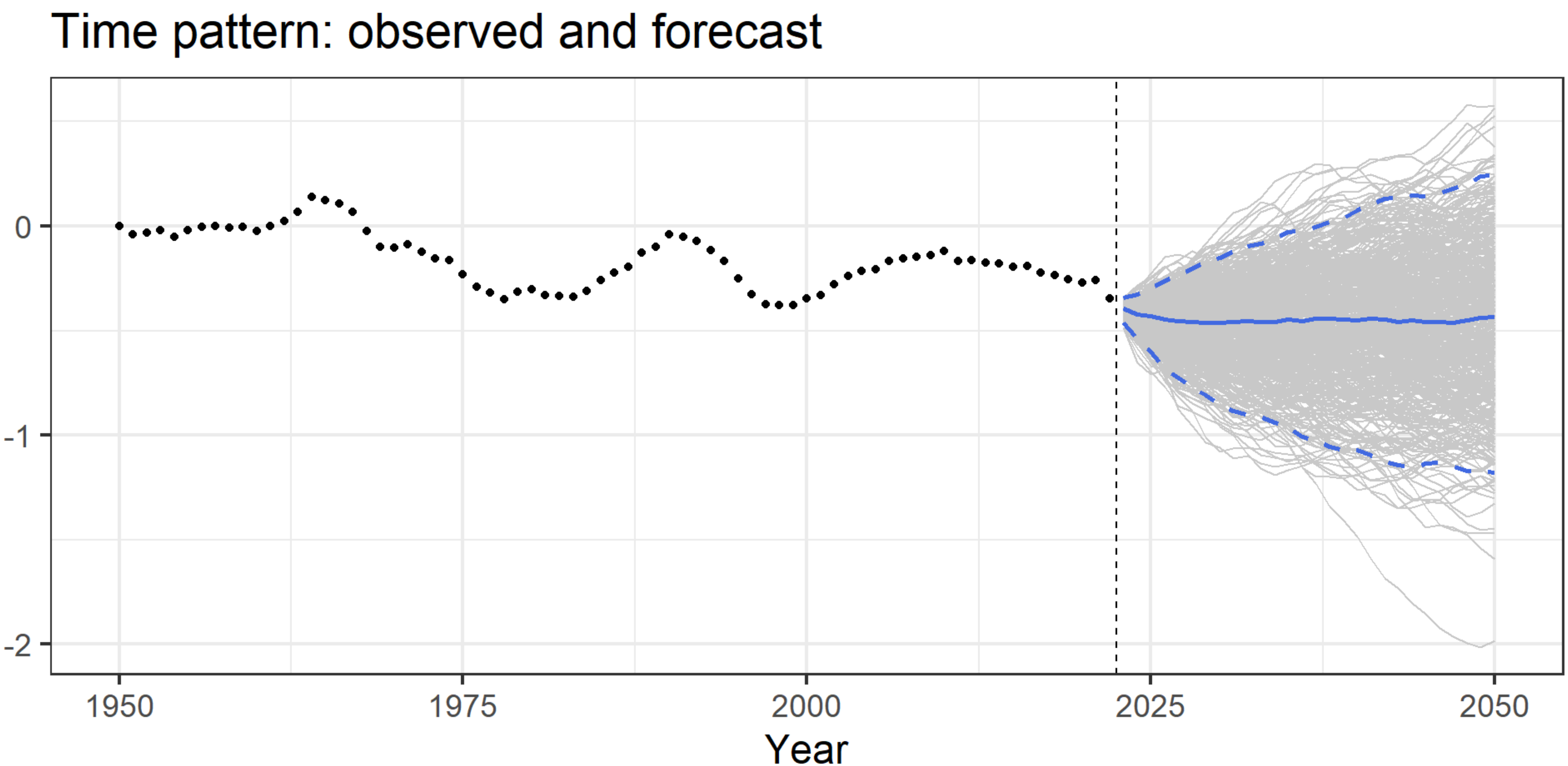
$$\begin{aligned} \ln(f_{x,t}) &= \beta_0 + \beta_1 x + \beta_2 x^2 + \sum_{i=2}^{n} \gamma_i \\ &= \beta_0 + \beta_1 x + \beta_2 x^2 + \kappa_t \end{aligned}$$

- Extrapolating the non-linear time trend (e.g. using an ARIMA model) can provide us with fertility forecasts

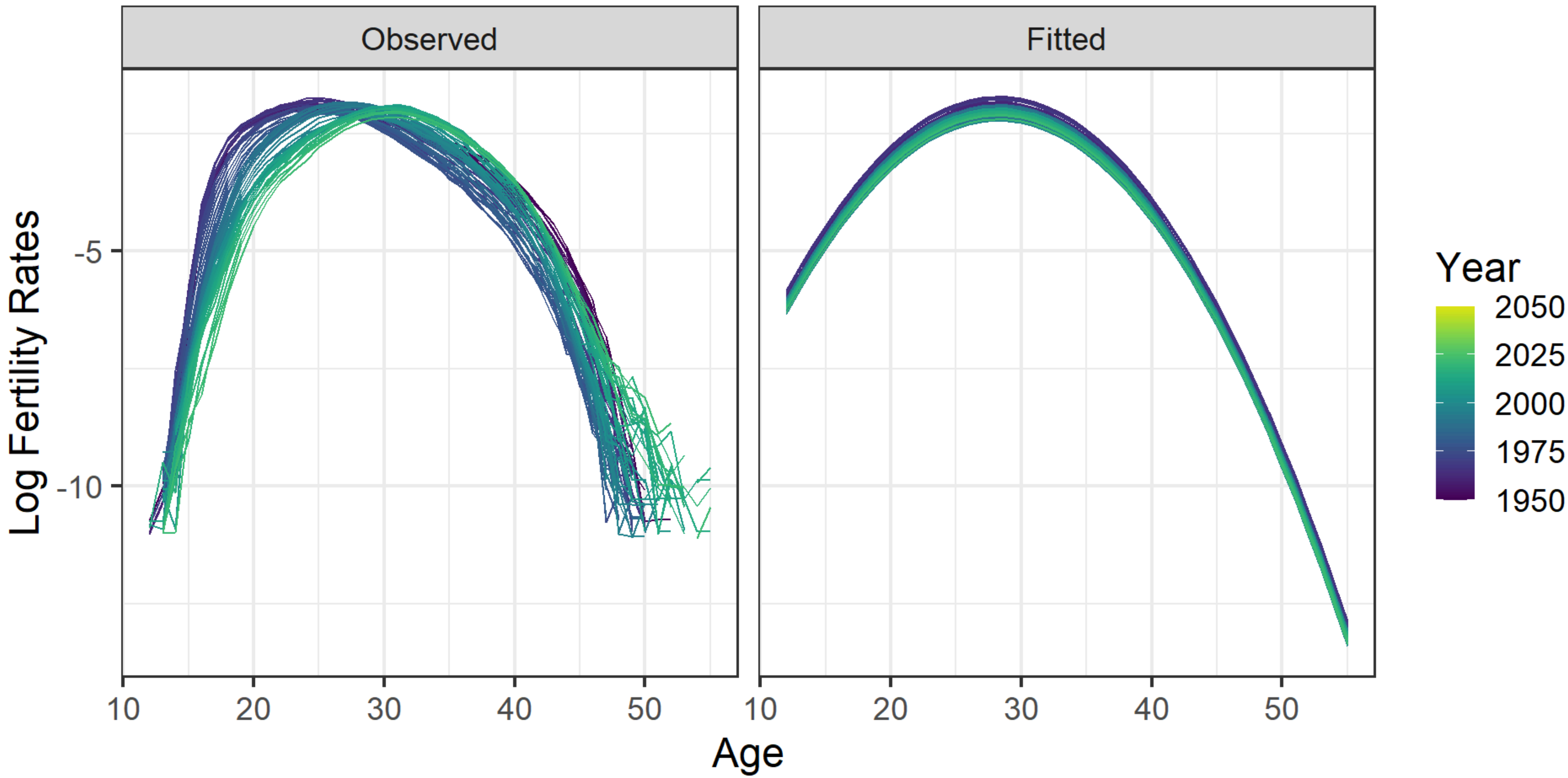
Flexible time index



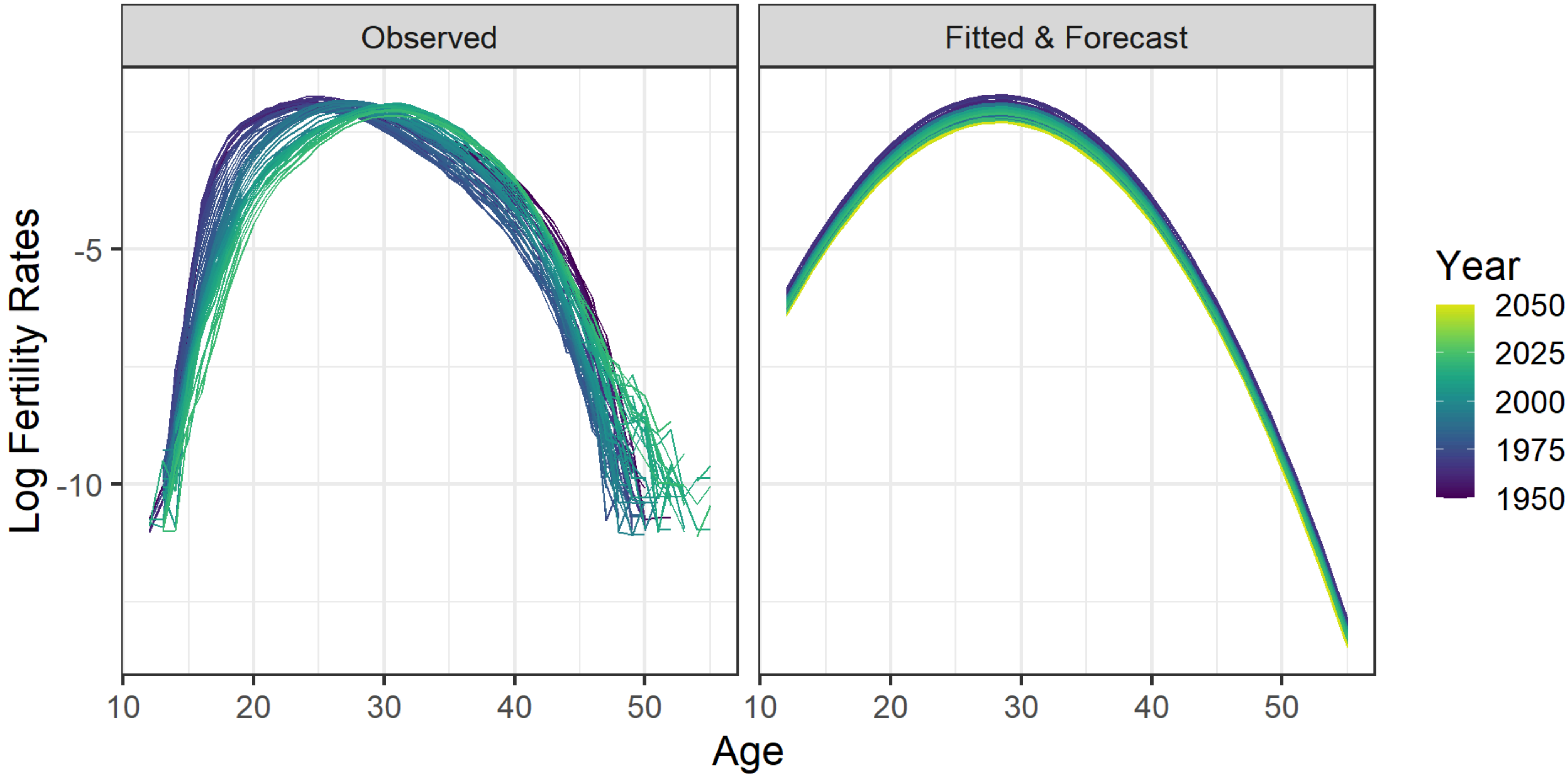
Flexible time index



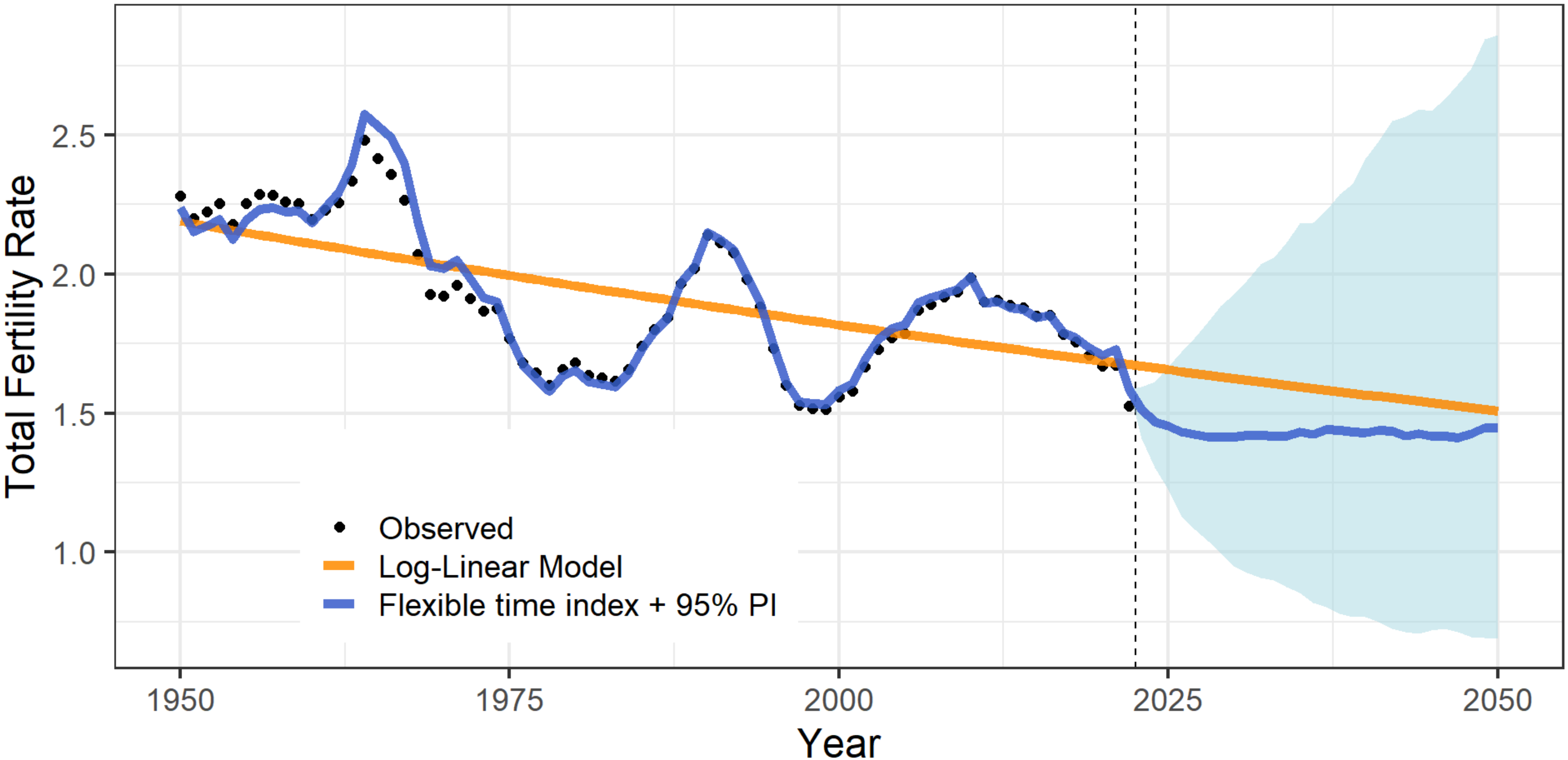
Flexible time index



Flexible time index



Flexible time index



Day 3 assignment

Assignment

6. Load the mortality data `MORTSWE.Rdata`, and focus on male mortality from 1950 onwards for ages $30 \leq x \leq 100$. Fit and forecast adult mortality up to 2050 using the parametric Gompertz model, fitting a model for each year independently [*hint*: this is a generalized linear model with deaths as response variable, exposures as an offset, and an intercept and age as covariates]. To forecast, fit the most appropriate ARIMA(p,d,q) models to the time-series of the two estimated parameters. Compute the 95% prediction intervals for life expectancy using simulations from the two ARIMA models.

7. Load the mortality data `MORTSWE.Rdata`, and focus on male mortality from 1950 onwards for ages $30 \leq x \leq 100$. Fit and forecast adult mortality up to 2050 using two different approaches:

- a Gompertz model with log-linear time trend, i.e. $\ln(m_{x,t}) = \beta_0 + \beta_1 x + \beta_2 t$
- a Gompertz model with flexible time index, i.e. $\ln(m_{x,t}) = \beta_0 + \beta_1 x + \kappa_t$

Plot the life expectancy forecasts of the two models (no need to derive PIs).

Hints: you can use the functions inside the `LifetableMX.R` code for constructing life tables and deriving estimates of life expectancy