Lab 11

Yaqi Shi, 1003813180

2023-04-03

Overview

In this lab you'll be fitting a second-order P-Splines regression model to foster care entries by state in the US, projecting out to 2030.

```
library(tidyverse)
library(here)
library(rstan)
library(tidybayes)
source("getsplines.R")
```

Here's the data

```
d <- read.csv("fc_entries.csv")
```

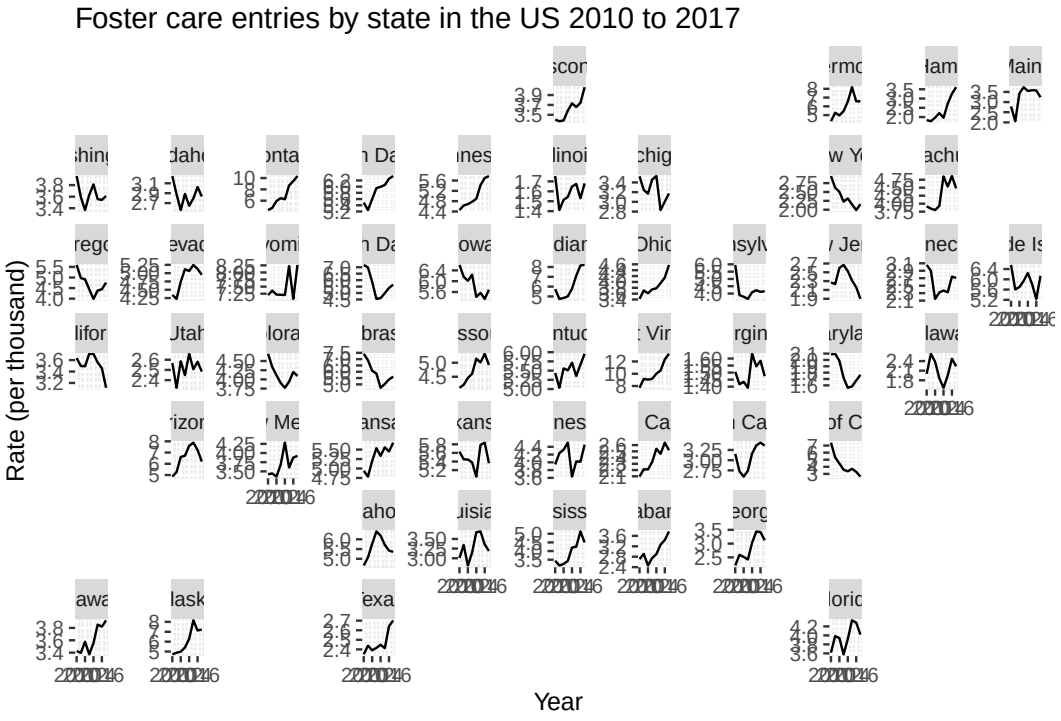
Question 1

Make a plot highlighting trends over time by state. Might be a good opportunity to use `geofacet`. Describe what you see in a couple of sentences.

Answer

Here is the plot using 'geofacet'that describes the state level trend over time:

```
library(geofacet)

d %>%
  ggplot(aes(year, ent_pc))+
  geom_line()+
  facet_geo(~state, scale="free_y")+
  labs(title = "Foster care entries by state in the US 2010 to 2017",
    x = "Year",
    y = "Rate (per thousand)")
```



Comment:

In this plot, we can see that the trend in each state is very different. Some stated experienced a increase from 2010 to 2017 (like Ohio). Some states fluctuates and having a decreasing trend in general(like Iowa). Some of the states first experienced a decrease then increased back (like Colorado). In general, more states experienced an increase in rates from 2010 to 2017. Each individual states has its own feature, thus we would consider model it separately (using state as a hierarchy).

Question 2

Fit a hierarchical second-order P-Splines regression model to estimate the (logged) entries per capita over the period 2010-2017. The model you want to fit is

$$
\begin{aligned}
y_{st} &\sim N(\log \lambda_{st}, \sigma^2_{y,s}) \\
\log \lambda_{st} &= \alpha_k B_k(t) \\
\Delta^2 \alpha_k &\sim N(0, \sigma^2_{\alpha,s}) \\
\log \sigma_{\alpha,s} &\sim N(\mu_\sigma, \tau^2)
\end{aligned}
$$

Where $y_{s,t}$ is the logged entries per capita for state s in year t. Use cubic splines that have knots 2.5 years apart and are a constant shape at the boundaries. Put standard normal priors on standard deviations and hyperparameters.

Answer

Here is the hierarchical second-order P-Splines regression model: (details of the model is in the stan file)

```
# Prepare the input for stan
years <- unique(d$year)
N <- length(years)

y <- log(d%>% select(state, year, ent_pc) %>%
  pivot_wider(names_from = "state", values_from = "ent_pc") %>%
```

```
  select(-year) %>%
  as.matrix())

res <- getsplines(years, 2.5)
B <- res$B.ik
K <- ncol(B)
stan_data <- list(N=N, y = y, K=K, S = length(unique(d$state)), B=B)
```

```
# Fit the model
mod <- stan(data = stan_data, file = "lab11.stan")
```

Question 3

Project forward entries per capita to 2030. Pick 4 states and plot the results (with 95% CIs). Note the code to do this in R is in the lecture slides.

Answer

Project forward entries per capita to 2030:

The four state I pick are California, Hawaii, Ohio and Tennessee

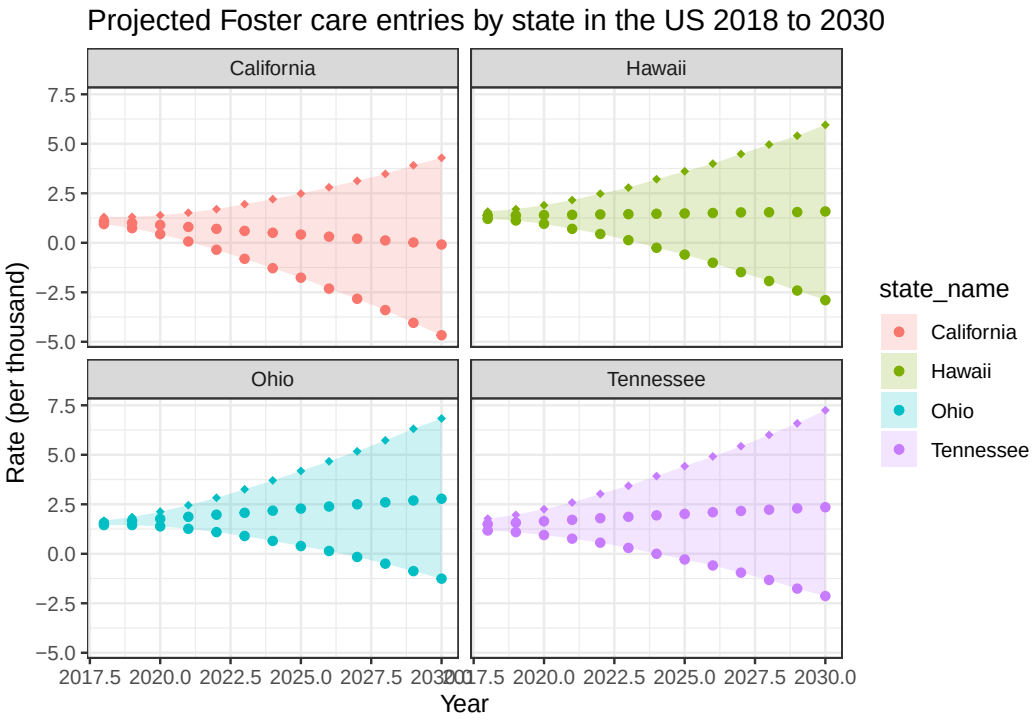
```
proj_years <- 2018:2030
# Note: B.ik are splines for in-sample period
# has dimensions i (number of years) x k (number of knots)
# need splines for whole period
B.ik_full <- getsplines(c(years, proj_years), 2.5)$B.ik
K <- ncol(B) # number of knots in sample
K_full <- ncol(B.ik_full) # number of knots over entire period
proj_steps <- K_full - K # number of projection steps

# get your posterior samples
alphas <- extract(mod)[["alpha"]]
sigmas <- extract(mod)[["sigma_alpha"]] # sigma_alpha
sigma_ys <- extract(mod)[["sigma_y"]]
nsims <- nrow(alphas)

# first, project the alphas
states <- unique(d$state)
alphas_proj <- array(NA, c(nsims, proj_steps, length(states)))
set.seed(1098)

# project the alphas
for(j in 1:length(states)){
  first_next_alpha <- rnorm(n = nsims, mean = 2*alphas[,K,j] - alphas[,K-1,j], sd = sigmas[,j])
  second_next_alpha <- rnorm(n = nsims, mean = 2*first_next_alpha - alphas[,K,j], sd = sigmas[,j])

  alphas_proj[,1,j] <- first_next_alpha
  alphas_proj[,2,j] <- second_next_alpha

  for(i in 3:proj_steps){
    alphas_proj[,i,j] <- rnorm(n = nsims,
    mean = 2*alphas_proj[,i-1,j] - alphas_proj[,i-2,j],
    sd = sigmas[,j])
  }
```

```
}

# now use these to get y's
y_proj <- array(NA, c(nsims, length(proj_years), length(states)))

for(i in 1:length(proj_years)){ # now over years
  for(j in 1:length(states)){
    all_alphas <- cbind(alphas[,,j], alphas_proj[,,j] )
    this_lambda <- all_alphas %*% as.matrix(B.ik_full[length(years)+i, ])
    y_proj[,i,j] <- rnorm(n = nsims, mean = this_lambda, sd = sigma_ys[,j])
  }
}

# Get the desired state and the median, 95% CI
state_ind <- c(5, 12, 36, 43)
proj_years <- 2018:2030

upper <- function(x) {
  quantile(x, 0.975)
}

lower <- function(x) {
  quantile(x, 0.025)
}

df_all <- c()

for(i in 1:4) {
  state_name <- rep(states[state_ind[i]], 13)
  ind <- state_ind[i]
  res <- y_proj[,,ind]
  med <- apply(res, 2, median)
  low <- apply(res, 2, lower)
  upp <- apply(res, 2, upper)

  df <- cbind(state_name, proj_years, med, low, upp)
  df_all <- rbind(df_all, df)
}

df_all <- data.frame(df_all)
df_all$state_name <- as.factor(df_all$state_name)
df_all$proj_years <- as.numeric(df_all$proj_years)
df_all$med <- as.numeric(df_all$med)
df_all$low <- as.numeric(df_all$low)
df_all$upp <- as.numeric(df_all$upp)

head(df_all)
```

```
##   state_name proj_years       med         low      upp
## 1 California       2018 1.0987428  0.94504496 1.304946
## 2 California       2019 1.0023156  0.73823898 1.308224
## 3 California       2020 0.9001515  0.43490303 1.389032
## 4 California       2021 0.8008658  0.06431287 1.516716
## 5 California       2022 0.7032655 -0.35234483 1.698190
```

```
## 6 California       2023 0.5981748 -0.80994862 1.949290
```

```
df_all %>%
  ggplot(aes(x = proj_years))+
  geom_point(aes(y = med, color = state_name))+
  geom_point(aes(y = low, color = state_name))+
  geom_point(aes(y = upp, color = state_name), pch = 18)+
  geom_ribbon(aes(y = med, ymin = low, ymax = upp, fill = state_name), alpha = 0.2)+
  theme_bw()+
  facet_wrap(~state_name)+
  labs(title = "Projected Foster care entries by state in the US 2018 to 2030",
    x = "Year",
    y = "Rate (per thousand)")
```



Question 4 (bonus)

P-Splines are quite useful in structural time series models, when you are using a model of the form

$$f(y_t) = \text{systematic part} + \text{time-specific deviations}$$

where the systematic part is model with a set of covariates for example, and P-splines are used to smooth data-driven deviations over time. Consider adding covariates to the model you ran above. What are some potential issues that may happen in estimation? Can you think of an additional constraint to add to the model that would overcome these issues?

Answer

The below was my notes in class, sorry I didn't have time to finish this :(

random walk is not stationry, you can model trends with that.

my model wont know which trends from spiline or the covariates.

i need the deviations from the expected part, it sun up to zero.