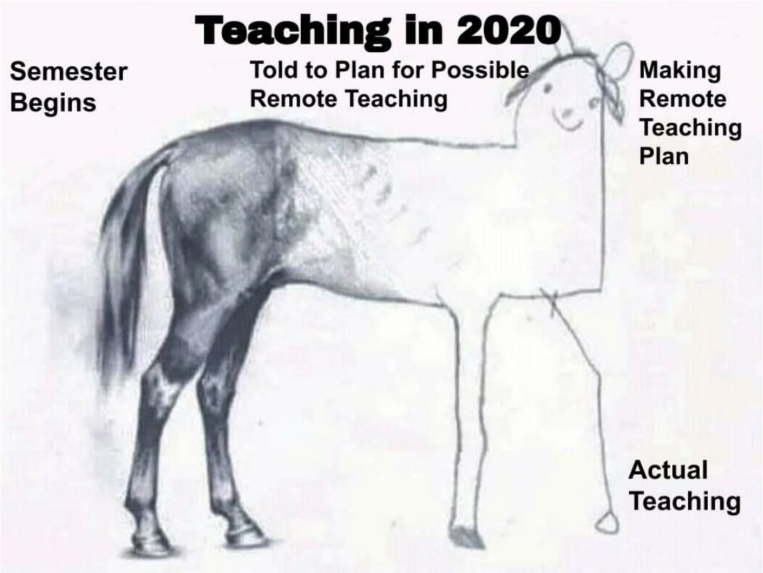
STA2201H Methods of Applied Statistics II

Monica Alexander

Week 10: Time Series Models


Teaching in 2020
Semester Begins
Told to Plan for Possible Remote Teaching
Making Remote Teaching Plan
Actual Teaching

Clarification from lab

Last week we were fitting models to lip cancer by region in GDR:

$$
\begin{aligned}
y_i &\sim \text{Poisson}\ (\theta_i \cdot e_i) \\
\log \theta_i &= \alpha_i + \beta(x_i^c)
\end{aligned}
$$

Assuming we want a different α_i for each region i (i.e. each region gets their own intercept), there are two options:

- Model α_i's non-hierarchically. This is just like adding region in as a factor: have a series of fixed effects on region. So the priors on the α_is are like any other standard regression coefficient, e.g. $\alpha \sim N(0, 1)$
- Model α_i's hierarchically so that $\alpha_i \sim N(\mu, \sigma_\mu^2)$. Now the α_is share **hyperparameters** and are drawn from the same distribution (c.f. where they are all from different distributions).

Reading for this week

Congdon (2006). Bayesian statistical modeling. Chapter 8

Goals of time series modeling

We observe outcome of interest at particular time points t, y_t.

- y_t may have additional indexes e.g. y_{st} (e.g. deaths in state s year t)
- y_t may be related to covariates X_t
- y_t may have missing obervations in the period

Some potential goals:

- forecasting
- back projecting
- reconstruction missing points
- smoothing

Goals of time series modeling

What you might be used to: Box-Jenkins approach.

Focus on the outcome:

- Start with y_t
- Remove anything systematic (trend, seasonality)
- Find an appropriate ARIMA specification
- Stationarity or death (differencing, transformations etc)

Perspective for this lecture

Strutural time series

Think about the outcome as:

$$y_t = \text{systematic part} + \text{fluctuations}$$

- The systematic part is potentially Trend + Seasonal Effects + Regression Term
- The errors/ fluctuations are likely to be autocorrelated because we're dealing with time
- We could model the systematic effects is by a set of fixed coefficients
- Or we could model them to vary over time, allowing for forecasts to place more weight on recent observations
- Intuitively: can model time dependency in outcome through time dependency in other parts
- We care less about stationarity, although still important for model specification and projections

Road map

- Simple AR(1) for y_t
  - how to run in Stan
  - how to forecast
- What if y_t have measurement error?
- What if we have missing observations?
- What if the mean is non-zero?

Example: foster care populations

- Linear trend models
- Random walk models
- Hierarchical extensions

Autocorrelation in response

Autocorrelation in response

- Given time limitations, we focus on main ideas and one specific (and simple) time series model AR(1)
- If you understand the simple model, you can get going with a larger class of models
- Start with simulated data

Simulated data

- I'm using functions from `distortr` package https://github.com/MJAlexander/distortr, something I wrote a while back
- Contains some potentially useful functions to simulate data to help think about measurement error, missing data, projections etc
- Might be useful if you're ever thinking about estimation and projection of time series that have a natural hierarchical structure
- Might also not be useful, who knows
- Can compare ARMA smoothers with P-splines (next week, hopefully) and Gaussian Processes

AR(1) process

A zero-mean autoregressive process y_t of order 1, referred to here as an $AR(1)$ process $y_t \sim AR(1)$ for $t = 0, \pm 1, \pm 2, \ldots$ is given by

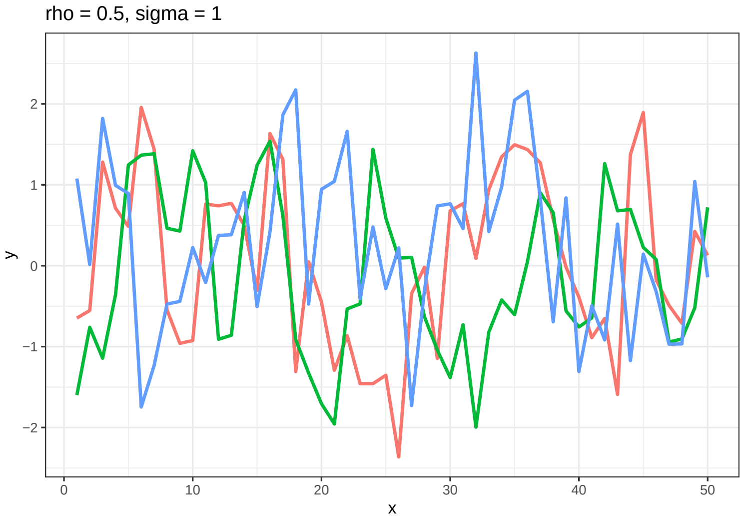
$$y_t = \rho y_{t-1} + \varepsilon_t$$
$$\varepsilon_t | \sigma \sim N\left(0, \sigma^2\right), \text{ independent}$$

- An $AR(1)$ process with normally distributed innovations ε_t and we assume that ε_t is indep. of y_{t-k} for $k > 0$

AR(1)

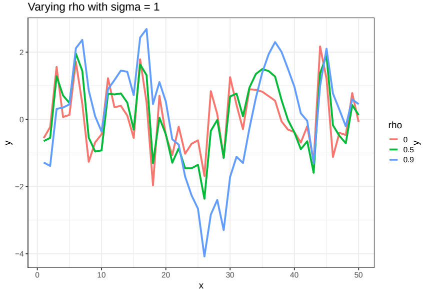
- An $AR(1)$ process is an example of a stochastic process: a sequence of random variables indexed by time.

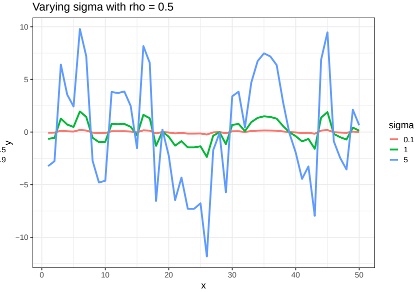
Three different simulations:



AR(1)

Interpretation of ρ? Interpretation of σ?





AR(1)

$$y_t = \rho y_{t-1} + \varepsilon_t$$
$$\varepsilon_t | \sigma \sim N\left(0, \sigma^2\right), \text{ independent}$$

- For fixed ρ, σ controls magnitude of series
- ρ determines strength of autocorrelation

Stationarity for time series processes

A time series process is weakly (or second order) stationary if

- Mean $E(y_t)$ is constant with time t
- Covariance function $\gamma_{t,t+k} = \text{Cov}\left(y_t, y_{t+k}\right)$ for any time t and time lag k depends on lag k only (is constant with time t).
- An AR(1) process is stationary if and only if $|\rho| < 1$
- If the AR(1) is stationary

$$\text{Var}(y_t) = \rho^2 \,\text{Var}(y_{t-1}) + \text{Var}(\varepsilon_t)$$

which implies stationary variance

$$\text{Var}\left(y_t\right) = \sigma^2 / \left(1 - \rho^2\right)$$

Stationarity

More general form, for $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$

$$\boldsymbol{y}|\rho, \sigma \sim N_n(\mathbf{0}, \Sigma)$$

with $\Sigma_{t,s} = \mathrm{Cov}\left(y_t, y_s|\rho, \sigma\right) = \sigma^2 / \left(1 - \rho^2\right) \cdot \rho^{|t-s|}$.

Fit and forecast in a Bayesian setting

Suppose we have time series $y_1, \ldots, y_n$ and want to fit a Bayesian zero-mean AR(1) model to it, to construct forecasts.

Proposed model

$$y_t \sim AR(1)$$
$$\rho \sim U(-1, 1)$$
$$\sigma \sim N_+(0, 1)$$

How to fit in Stan? What's the likelihood of the y_i's?

How to fit in Stan?

We wrote that $\boldsymbol{y}|\rho,\sigma \sim N_n(\mathbf{0},\Sigma)$, so could fit based on that. But this is slow! Generally good to avoid Multivariate normals is possible.

Faster option: decompose the likelihood function

$$p(\boldsymbol{y}) = p\left(y_1\right) p\left(y_2|y_1\right) p\left(y_3|y_2,y_1\right) \cdot \ldots \cdot p\left(y_n|y_{n-1},\ldots,y_1\right)$$

where

$$y_t = \rho y_{t-1} + \varepsilon_t$$

$$y_t|y_{t-1}\rho,\sigma \sim N\left(\rho y_{t-1},\sigma^2\right)$$

$$p\left(y_t|y_{t-1},\ldots,y_1,\rho,\sigma\right) = p\left(y_t|y_{t-1},\rho,\sigma\right)$$

Model block

```
model {

   y[2:N] ~ normal(rho * y[1:(N - 1)], sigma);

  // equivalent, but slower
  //for (n in 2:N)
    //  y[n] ~ normal(rho * y[n-1], sigma);
}
```

AR(1) in Stan

Fine, but what happened to y_1?

- Could just not model, condition on y_1, so leave out of data (what is done in Stan manual!)
- Loss of data in likelihood, hence less preferable (but ok if you are working with long time series).

Other option: use stationary distribution for y_1:

$$y_1 \sim N\left(0, \sigma^2 / \left(1 - \rho^2\right)\right)$$

Fitting to simulated data with $\rho = 0.5$ and $\sigma = 0.1$

```
y <- GetAR(nyears = 100, rho = 0.5, sigma = 0.1)
N <- length(y)

mod1 <- stan(data = list(y = y, N = N), file = "ar1_1.stan", iter = 100
```

```
##          mean   Rhat
## rho   0.4878 1.0480
## sigma 0.0961 0.9847
```

How to get projections?

- Given and posterior sample $\rho^{(s)}$ and $\sigma^{(s)}$ one can forecast trajectory $y_{n+p}^{(s)}$ with $p \geq 1$ as

$$y_{n+p}^{(s)} | y_{n+p-1}^{(s)}, \rho^{(s)}, \sigma^{(s)} \sim N\left(\rho^{(s)} y_{n+p-1}^{(s)}, \left(\sigma^{(s)}\right)^2\right)$$

where $y_n^{(s)} = y_n$. Once we have set of posterior samples, $y_{n+p}^{(1)}, y_{n+p}^{(2)}, \ldots, y_{n+p}^{(S)}$ point forecasts and 95% CIs can be constructed.

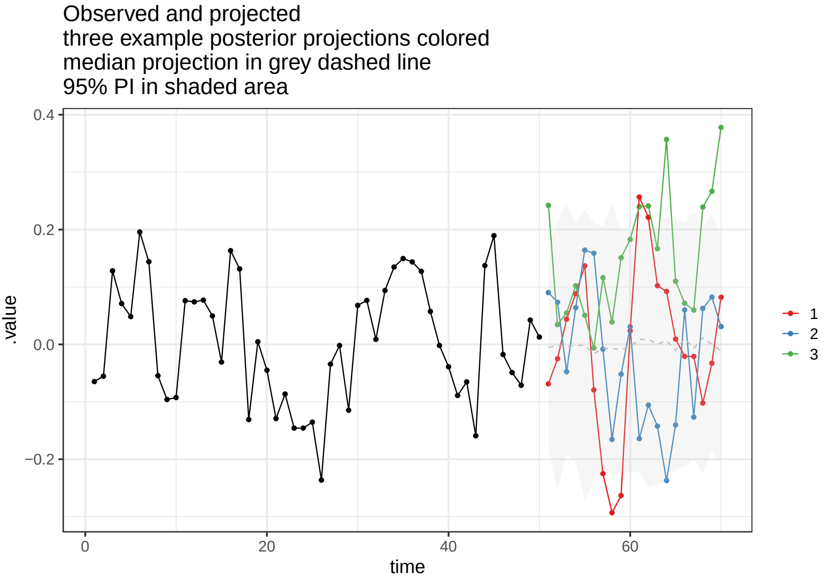
- Can do in R or in Stan
- Note: can also back-project in the same way

Projections in Stan using the generated quantities block

The full model (also see git repo)

```
data {
  int<lower=0> N;
  int<lower=0> P;
  vector[N] y;
}
parameters {
  real<lower = -1, upper = 1> rho;
  real<lower=0> sigma;
}
model {
   //likelihood
   y[1] ~ normal(0, sigma/sqrt((1-rho^2)));
   y[2:N] ~ normal(rho * y[1:(N - 1)], sigma);

   //priors
   rho ~ uniform(-1, 1);
   sigma ~ normal(0,1);
}
generated quantities {
  //project forward P years
  vector[P] y_p;
  y_p[1] = normal_rng(rho*y[N], sigma);
  for( i in 2:P){
    y_p[i] = normal_rng(rho*y_p[i-1], sigma);
  }
}
```

Results



Observed and projected
three example posterior projections colored
median projection in grey dashed line
95% PI in shaded area
0.4
0.2
0.0
−0.2
.value
0
20
40
60
time
1
2
3

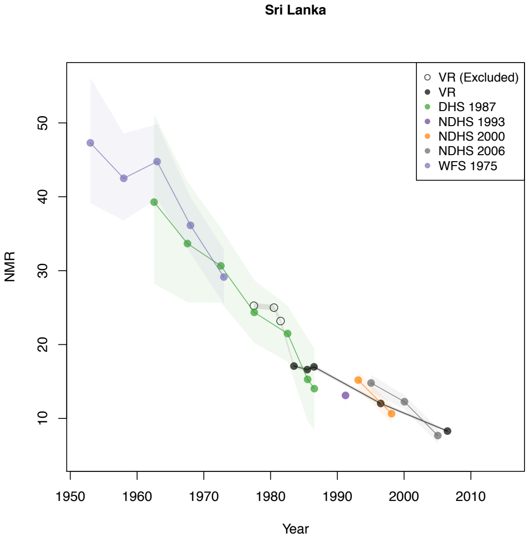
Measurement error

Measurement error

- So far we have taken y_t at face value
- Now suppose we observe an AR(1) series with observational error i.e. $y_t = \mu_t + e_t$ with $\mu_t \sim AR(1)$ and error $e_t \sim N\left(0, \delta_t^2\right)$
- could estimate $\delta = \delta_t$ if measurement error unknown
- sometimes measurement error is known e.g. for surveys

Motivating example

Data available on neonatal mortality in Sri Lanka



Observations are subject to measurement error

$$y_t = \mu_t + e_t$$
$$\mu_t \sim AR(1)$$
$$e_t|\delta \sim N\left(0, \delta^2\right)$$

How to model?

- Use the AR(1) set-up as before for μ_t (as opposed to for y_t) and add data model
- Model for y's is data model, model for μ's is process model
- Interpretation: observations are 'truth' plus some error, we are interested in truth.

Model with measurement error

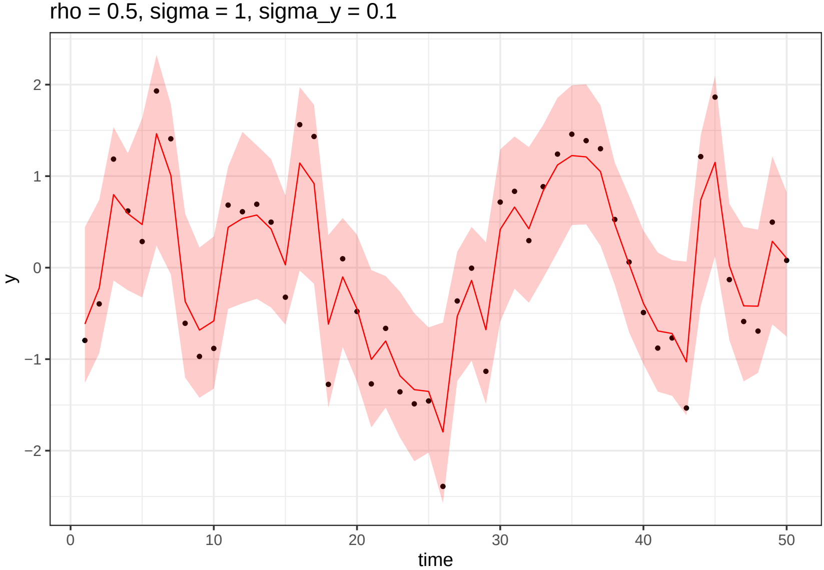
```
data {
  int<lower=0> N;
  vector[N] y;
}
parameters {
  real<lower = -1, upper = 1> rho;
  vector[N] mu;
  real<lower=0> sigma;
  real<lower=0> sigma_y;
}
model {

   y ~ normal(mu, sigma_y);
   mu[1] ~ normal(0, sigma/sqrt((1-rho^2)));
   mu[2:N] ~ normal(rho * mu[1:(N - 1)], sigma);

   //priors
   rho ~ uniform(-1,1);
```

Simulate some data

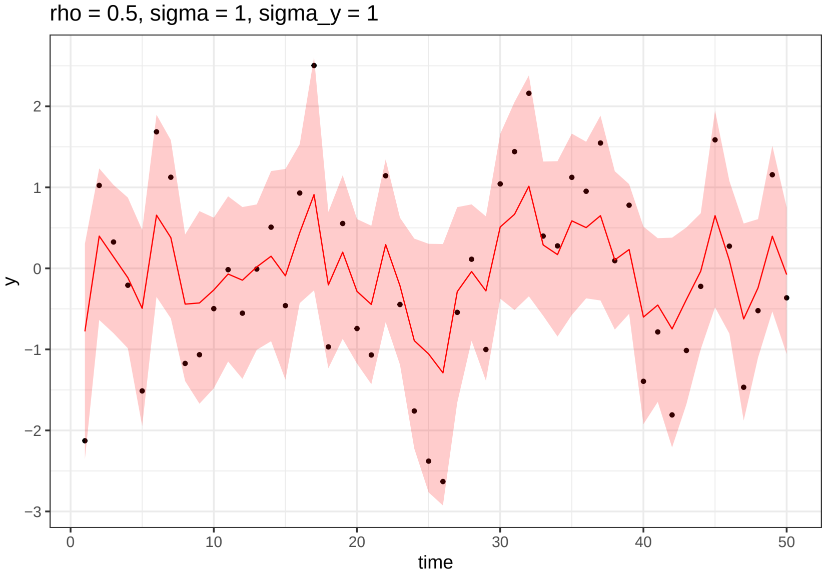
```r
nyears <- 50
rho <- 0.5
sigma <- 1
sigma.y <- 0.1

mu.t <- GetAR(nyears, rho, sigma)
set.seed(12)
y.t <- mu.t + rnorm(nyears, 0, sigma.y)
```

Results

The data are in black, the fit is in red. Note the μ's are plotted ("truth")



What happens if we have more measurement error



Missing data

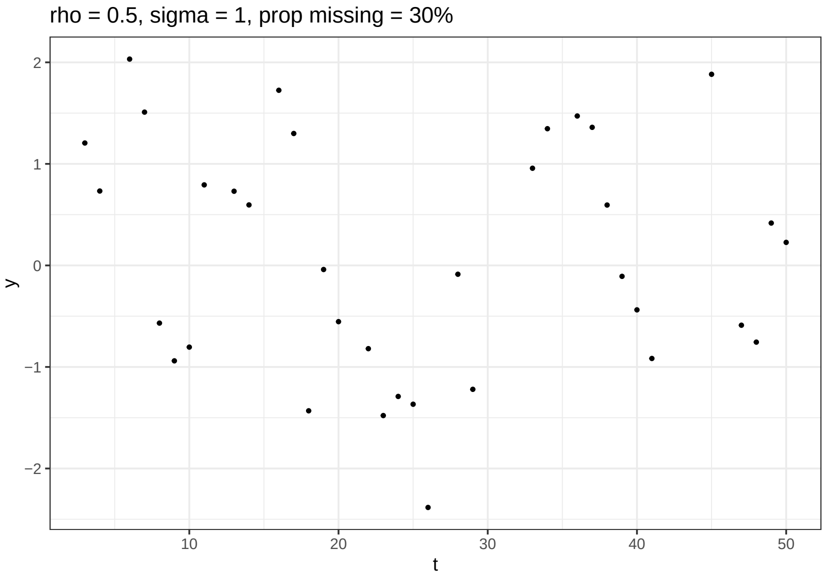
Missing data

- Now imagine we have observations y_t but some t's are missing
- e.g. if we observe $y_1, y_2, \ldots, y_n$ from time points $t_1, t_2, \ldots, t_n$ with $t_i \neq t$.
- As above, keep process model the same but change the data model
- Need to create an indexing vector $t[i]$ which tells you what t the ith observation refers to

Missing data Stan model

```
data {
  int<lower=0> N;
  int<lower=0> N_obs;
  vector[N_obs] y;
  int t_i[N_obs];
}
parameters {
  real<lower = -1, upper = 1> rho;
  vector[N] mu;
  real<lower=0> sigma;
  real<lower=0> sigma_y;
}
model {

   y ~ normal(mu[t_i], sigma_y);
   mu[1] ~ normal(0, sigma/sqrt((1-rho^2)));
   mu[2:N] ~ normal(rho * mu[1:(N - 1)], sigma);

   //priors
   rho ~ uniform(-1,1);
   sigma ~ normal(0,1);
   sigma_y ~ normal(0,1);
```

Missing data: simulation



Results

Missing data

Suppose we want to get μ_t given μ_{t-1} and μ_s, where $s > t$. What is the conditional mean and variance of μ_t?

It turns out that (notes on GitHub)

$$E\left(\mu_t | \mu_{t-1}, \mu_s\right) = \frac{1}{1-A}\left(\rho \cdot \left(1-\rho^{2(s-t)}\right) \cdot \mu_{t-1} + \rho^{s-t}\left(1-\rho^2\right) \cdot \mu_s\right)$$

$$\text{Var}\left(\mu_t | \mu_{t-1}, \mu_s\right) = \frac{\sigma^2}{1-\rho^2}\left(1 - \frac{\rho^2 - 2A + \rho^{2(s-t)}}{1-A}\right)$$

$$A = \rho^{2(s-t+1)}$$

- Conditional mean is weighted average of two points, where weights depend on how far away s is
- Variance increases with s

Non-zero means

Non-zero means

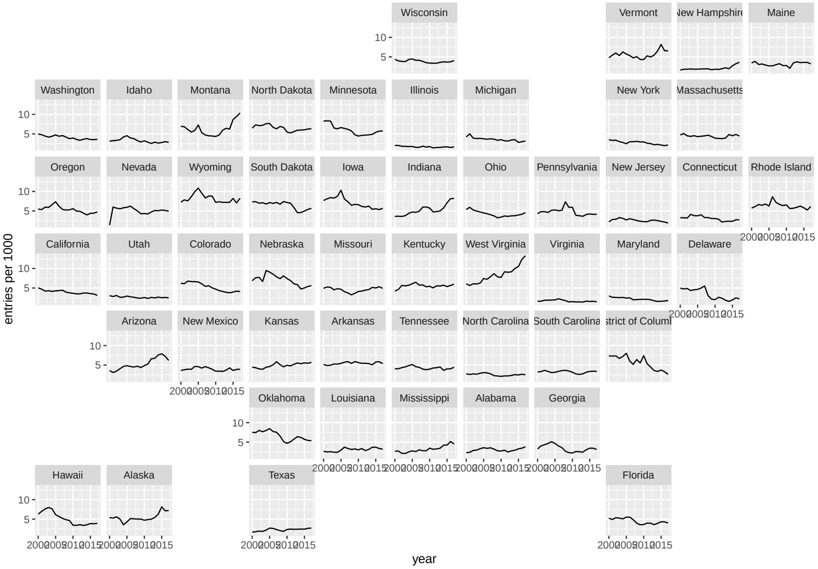
Suppose we have the following candidate model for y_t

$$y_t|\gamma_t, \delta \sim N\left(\gamma_t, \delta^2\right)$$
$$\gamma_t = \kappa_t + \mu_t, \text{ with } \mu_t \sim AR(1)$$

- μ_t is zero-mean AR(1) model
- κ_t could be
  - a constant α
  - related to covariate e.g. $\kappa_t = x_t\beta$
  - ...
- Fit as before but add in mean term
- Easy in theory, in practice model specification often hard

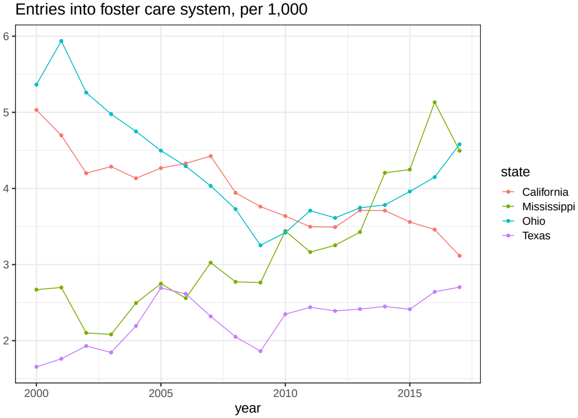
Example from this point

Goal: Project foster care populations by state in the US



Projecting foster care populations

- There's a number of different outcomes of interest, but let's look at entries into system (children aged 0-17)
- Let's use population of children as exposure variable, alternatively, think of modeling entries per captia
- Ignore issues of population age structure for now



Foster care populations

- Goal is projection, but understanding is important
  - why are things going up or down?
  - Are there driving factors that are modifiable or can be planned for?
- Uncertainty around projections is important

How to approach problem?

Data model

- y_{st} is number of entries into foster care system in state s and year t
- P_{st} is child population in same state and year

$$y_{st} \sim \text{Poisson}(\lambda_{st} P_{st})$$

λ_{st} is rate of entries, the outcome of interest. Model for λ_{st}?

Model for λ_{st}?

Start with no covariates (apart from time!)

Possibilities:

- Simplest would be

$$\log \lambda_{st} = \alpha + \beta t + \varepsilon_{st}$$

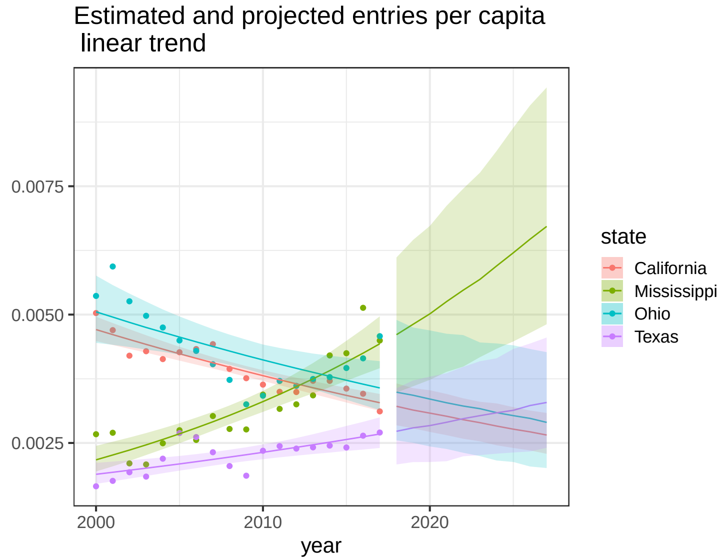
with $\varepsilon_{st} \sim N(0, \sigma^2)$

- What about autocorrelated errors

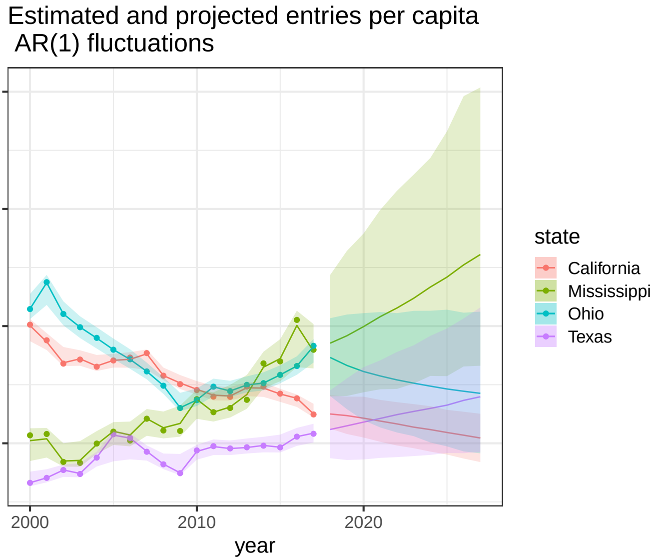
$$\log \lambda_{st} = \alpha + \beta t + \varepsilon_{st}$$

with $\varepsilon_{st} \sim AR(1)$

Linear in time



Linear in time with AR(1) errors



Moving away from non-linear trends

- Linear trend + AR(1) wasn't terrible, but probably want to put more weight on more recent observations
- Simplest option here is a random walk:

$$\log \lambda_{st} = \alpha_{st}$$

with $\alpha_{st} \sim N(\alpha_{s,t-1}, \sigma_s^2)$ or equivalently $\Delta\alpha_{st} \sim N(0, \sigma_s^2)$.

Random walk

Now we've lost stationary. The α's have the form

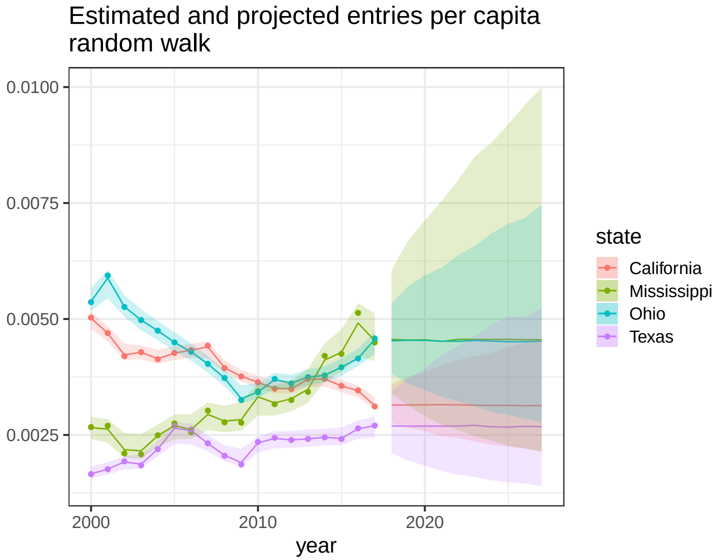
$$\alpha_t = \alpha_{t-1} + \varepsilon_t$$
$$\varepsilon_t | \sigma \sim N\left(0, \sigma^2\right)$$

Suppose $\alpha_1 = 0$, and that σ is known, then for $t > 0$, then

$$E\left(\alpha_t\right) = E\left(\alpha_{t-1}\right) + E\left(\varepsilon_t\right) = 0 \text{ and}$$
$$\mathrm{Var}\left(\alpha_t\right) = \mathrm{Var}\left(\alpha_{t-1}\right) + \mathrm{Var}\left(\varepsilon_t\right) = (t-1)\sigma^2$$

In practice what does this mean for our projections?

Random walk


Estimated and projected entries per capita
random walk
0.0100
0.0075
0.0050
0.0025
2000
2010
2020
year
state
California
Mississippi
Ohio
Texas

Random walk

- We've gone from our projections to caring about all years to just caring about the last year
- Projections in RW are based on the last observed level
- Uncertainty increases forever with time (c.f. stationary AR(1))

Higher-order random walks

We can increase the random walk's memory by moving to higher order random walks. E.g. a second-order random walk is

$$\log \lambda_{st} = \alpha_{st}$$

with

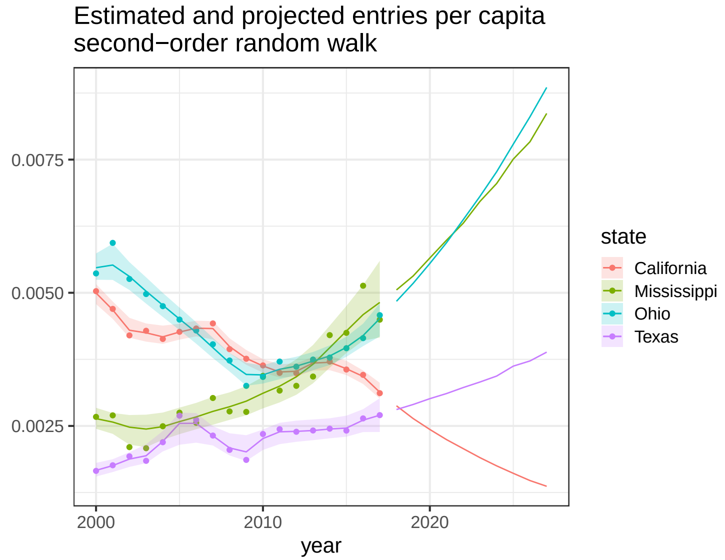
$\alpha_{st} - \alpha_{s,t-1} \sim N(\alpha_{s,t-1} - \alpha_{s,t-2}, \sigma_s^2)$ or equivalently

$\alpha_{st} \sim N(2\alpha_{s,t-1} - \alpha_{s,t-2}, \sigma_s^2)$ or equivalently

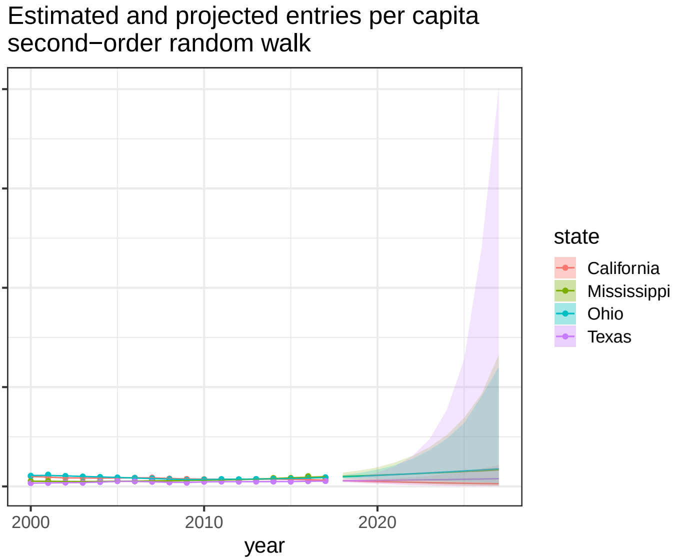
$\Delta^2 \alpha_{st} \sim N(0, \sigma_s^2)$.

If a first-order RW projects the level, what does a second-order RW project?

Second order RW



Oh no



Estimated and projected entries per capita
second−order random walk
0.20
0.15
0.10
0.05
0.00
2000
2010
2020
year
state
California
Mississippi
Ohio
Texas

Moving forward: hierarchical model

- Second order random walk gives ‘reasonable’ point estimates but unrealistic and unusable uncertainty intervals
- But we are working with hierarchical data: states within regions within the US
- Currently we are fitting a separate time series to each state
- Could model hierarchically such that information about the variability in the random walks (i.e. the σ^2 term) could be shared across states

Hierachical model for σ_s^2

A plausible set-up:

$$\log \lambda_{st} = \alpha_{st}$$

with

$\alpha_{st} \sim N(2\alpha_{s,t-1} - \alpha_{s,t-2}, \sigma_s^2)$

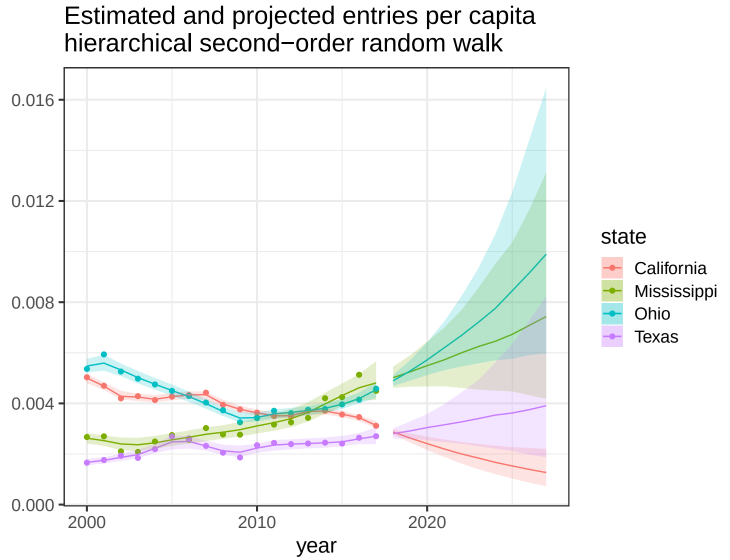
and

$\log \sigma_s \sim N(\mu_\sigma, \tau^2)$

with the usual prior on $\tau \sim N_+(0, 1)$.

- model the log of the σ's to ensure positive
- Make sure you can see that this is hierarchical. For reference, the non-hierarchical model just has $\sigma_s \sim N_+(0, 1)$

Looking better



Foster care: summary

A work in progress

- Second order RW shows promise in picking up characteristics of time series
- But of little use for understanding **why** changes are happening, and whether they are likely to happen in future

Moving forward: covariates

- a whole suite of candidate covariates
- could potentially model coefficients hierarchically
- could potentially model time varying effect of covariates
  - e.g. random walk on the effect of drug use over time

Summary

- AR(1) models easy to fit in Stan once you have the code (which you do now)
- Relatively straight forward to extend to more complicated structures
- Stationarity may or may not matter, but will effect uncertainty levels in projections
- Think about not only autocorrelation in the response, but also the structural components driving the response

Summary

Hierarchical take-aways:

- Up until today we have been putting hierarchical structures on regression coefficients (slopes, intercepts)
- Can also put hierarchical model on variance terms!
- Interpretation: the variability in a series in a particular state tells us something about the variability in another state
- Has the effect of shrinking the variance towards a global mean

Model checking?

- In general, you can't use LOO-CV to compare time series models in the same way we have been doing, because of the time dependence in the data

Possibilities:

- As usual: residual plots, where residual = observation - estimate
- Out-of-sample validation, e.g. leaving out data at random (if reconstruction of missing values is of interest) or the most recent observations (if forecasting is of interest).
- In-sample validation (depending on context): construct 1 (or more)-step ahead forecasts and compare observation to that forecast.
- There is a 'future' version of LOO discussed here: https://mc-stan.org/loo/articles/loo2-lfo.html

Lab and rest of semester

- No official lab today (you will magically get 10/10 for Lab 10)
- Will post rmd at some point walking through an example of Stan/brms workflow (hopefully help with dealing with output, model checking etc)
- Assignment 4 posted later this week
- No in-person exam, at this stage will be a take home (more details next week)