Class 2: State-space and change point models

Andrew Parnell
andrew.parnell@ucd.ie

Learning outcomes

- Learn the basics of parameter and state estimation for simple state space models
- Fit different types of change point models

Introduction to state space models

- State space models are a very general family of models that are used when we have a noisy time series of observations that are stochastically related to a hidden time series which is what we are really interested in
- The example we will use in this lecture is for palaeoclimate reconstruction when we observe pollen but are really interested in climate
- All state space models have two parts:
  - The first part is called the *state equation* which links the observations to a latent *stochatic process*
  - The second part of the model is called the *evolution equation* which determines how the latent stochastic process changes over time

A simple linear state space model

- We define y_t in the usual way, but write x_t for the hidden stochastic process
- For a simple linear state space model we have a *state equation* of:

$$y_t = \alpha_y + \beta_y x_t + \epsilon_t,\ \epsilon_t \sim N(0, \sigma_y^2)$$

- The *evolution equation* could be a random walk:

$$x_t = x_{t-1} + \gamma_t,\ \gamma_t \sim N(0, \sigma_x^2)$$

- The usual aim when fitting these models is to either estimate x_t, or the parameters $(\alpha_y, \beta_y, \sigma_y, \sigma_x)$, or to predict future values of x_t
- This type of model is sometimes known as the *Kalman Filter*

JAGS code for a linear state space model

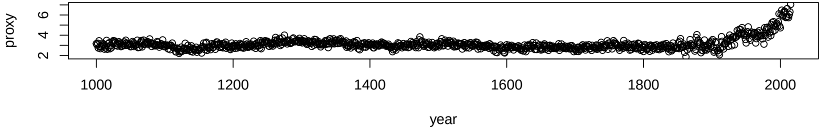
```
model_code = '
model
{
  # Likelihood
  for (t in 1:T) {
    y[t] ~ dnorm(alpha_y + beta_y * x[t], sigma_y^-2)
  }
  x[1] ~ dnorm(0, 100^-2)
  for (t in 2:T) {
    x[t] ~ dnorm(x[t-1], sigma_x^-2)
  }

  # Priors
  sigma_y ~ dunif(0, 100)
  sigma_x ~ dunif(0, 100)
}'
```

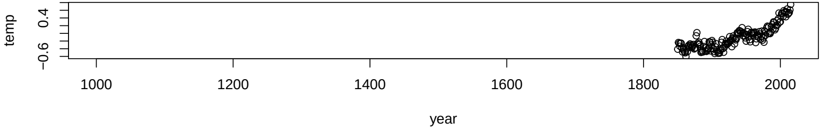
Priors for state space models

- You need to be very careful with state space models as it's very easy to create models which are ill-defined and crash
- For example, in the Kalman filter model you can switch the sign of x_t and β_y and still end up with the same model
- It's advisable to either fix some of the parameters, or use extra data to calibrate the parameters of the state space model

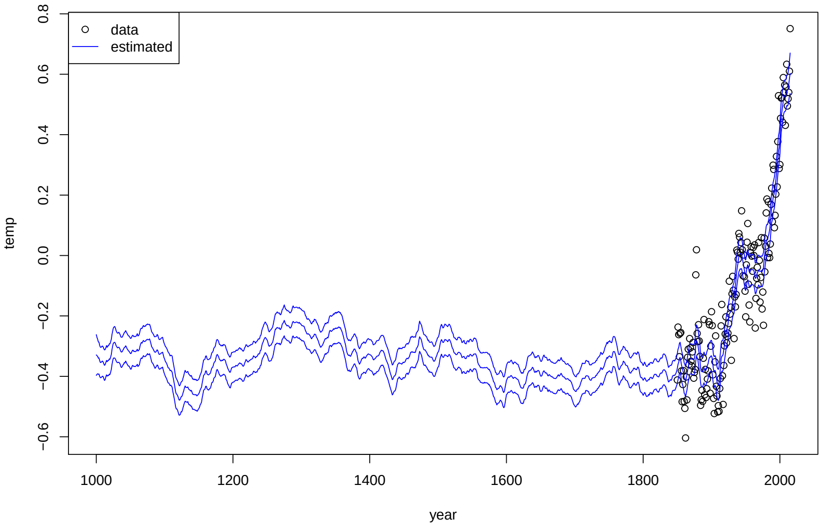
Example: palaeoclimate reconstruction

```r
palaeo = read.csv('../../data/palaeo.csv')
par(mfrow=c(2,1))
with(palaeo,plot(year, proxy))
with(palaeo,plot(year, temp))
```





Palaeoclimate reconstruction results



More advanced state space models

- State space models can get much more advanced
- We can make the state equation richer by making the relationship between the response and the latent time series more complex
- We can make the evolution equation richer by including a more complex time series model, e.g. an OU process
- We can extend the model if the response is multivariate, or allow the latent time series to be multivariate
- In fact the model will often fit better if you have multivariate observations or stricter requirements about the time series applied to x_t

Change point models

Introduction to change point models

- Another method commonly used for both discrete and continuous time stochastic processes is that of change point modelling
- The goal is to find one or more *change points*; times at which the time series changes in some structural way
- We will study two versions of change point models; *discontinuous*, where there can be instantaneous jumps in the mean, and *continuous* where there can be a jump in the rate of change of the mean, but subsections must link together

Dscontinuous change point regression models

- We will write the overall model as:

$$y(t) \sim N(\mu(t), \sigma^2)$$

- For the discontinuous change point regression (DCPR) model with one change point

$$\mu(t) = \begin{cases} \alpha_1 & \text{if } t < t_1 \\ \alpha_2 & \text{if } t \geq t_1 \end{cases}$$

- Here, α_1 and α_2 are the mean before and after the change point respectively, and t_1 is a parameter which gives the time of the change in the mean
- In JAGS we use the `step` function to determine which side of the change point a data point is currently on

JAGS code

```
model_code_DCPR_1="
model
{
  # Likelihood
  for(i in 1:T) {
    y[i] ~ dnorm(mu[i], sigma^-2)
    mu[i] <- alpha[J[i]]
    # This is the clever bit - only pick out the right
      change point when above t_1
    J[i] <- 1 + step(t[i] - t_1)
  }

  # Priors
  alpha[1] ~ dnorm(0, 10^-2)
  alpha[2] ~ dnorm(0, 10^-2)
  t_1 ~ dunif(t_min, t_max)

  sigma ~ dunif(0, 100)
}
"
```

Continuous change point regression models

- The continuous change point regression model (CCPR) forces the segments to join together
- The mean for this version is:

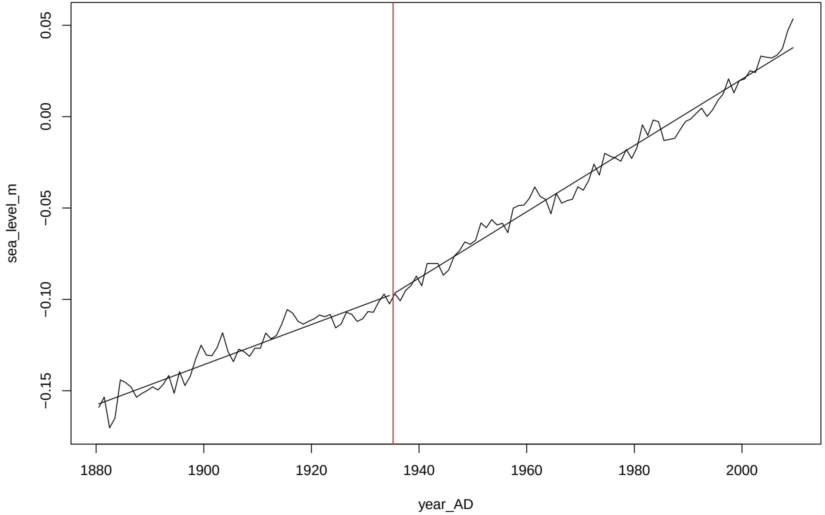
$$\mu(t) = \begin{cases} \alpha + \beta_1(t - t_1) & \text{if } t < t_1 \\ \alpha + \beta_2(t - t_1) & \text{if } t \geq t_1 \end{cases}$$

- In this version β_1 and β_2 are the rates of change before and after the change point, α is the mean value of y at the change point

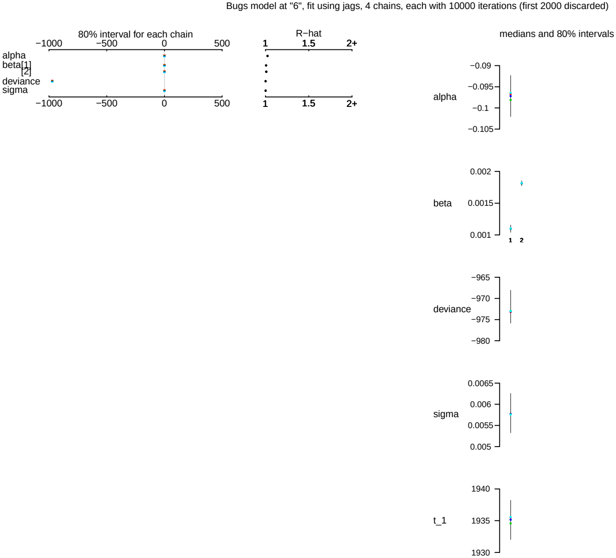
JAGS code for CCPR

```
model_code_CCPR_1="
model
{
  # Likelihood
  for(i in 1:T) {
    y[i] ~ dnorm(mu[i], sigma^-2)
    mu[i] <- alpha + beta[J[i]]*(t[i]-t_1)
    J[i] <- 1 + step(t[i] - t_1)
  }

  # Priors
  alpha ~ dnorm(0, 10^-2)
  beta[1] ~ dnorm(0, 10^-2)
  beta[2] ~ dnorm(0, 10^-2)
  t_1 ~ dunif(t_min, t_max)

  sigma ~ dunif(0, 100)
}
"
```

Example: change points of tide gauge data



Plots of the change-point parameters



Multiple change-points

- We don't have to stop at just one change point, though the model gets a bit more complicated for 2, 3, ... change-points
- Often run into convergence problems with multiple change points. Usually we would *sort the change points* so that e.g. $t_1 < t_2 < \ldots < t_k$
- Usually fit 1CP model, 2CP model, etc, and choose via AIC/DIC/WAIC, etc
- Impossible to fit these models in Stan!

Summary

- We have seen how to fit basic Bayesian state space models and observed some of their pitfalls
- We have covered discontinuous and continuous change point models, and shown how they apply to some data sets