Time Series Modelling and Kalman Filters

Chris Williams

School of Informatics, University of Edinburgh

November 2010

Outline

- Stochastic processes
- AR, MA and ARMA models
- The Fourier view
- Parameter estimation for ARMA models
- Linear-Gaussian HMMs (Kalman filtering)
- Reading: Handout on Time Series Modelling: AR, MA, ARMA and All That
- Reading: Bishop 13.3 (but not 13.3.2, 13.3.3)

Example time series

- FTSE 100
- Meteorology: temperature, pressure ...
- Seismology
- Electrocardiogram (ECG)
- ...

Stochastic Processes

- A stochastic process is a family of random variables $X(t),\ t \in T$ indexed by a parameter t in an index set T
- We will consider *discrete-time* stochastic processes where $T = \mathbb{Z}$ (the integers)
- A time series is said to be *strictly stationary* if the joint distribution of $X(t_1), \ldots, X(t_n)$ is the same as the joint distribution of $X(t_1 + \tau), \ldots, X(t_n + \tau)$ for all $t_1, \ldots, t_n, \tau$
- A time series is said to be *weakly stationary* if its mean is constant and its autocovariance function depends only on the lag, i.e.

$$E[X(t)] = \mu \qquad \forall\ t$$
$$\mathrm{Cov}[X(t)X(t+\tau)] = \gamma(\tau)$$

- A Gaussian process is a family of random variables, any finite number of which have a joint Gaussian distribution
- The ARMA models we will study are stationary Gaussian processes

Autoregressive (AR) Models

- Example AR(1)

$$x_t = \alpha x_{t-1} + w_t$$

where $w_t \sim N(0, \sigma^2)$
- By repeated substitution we get

$$x_t = w_t + \alpha w_{t-1} + \alpha^2 w_{t-2} + \dots$$

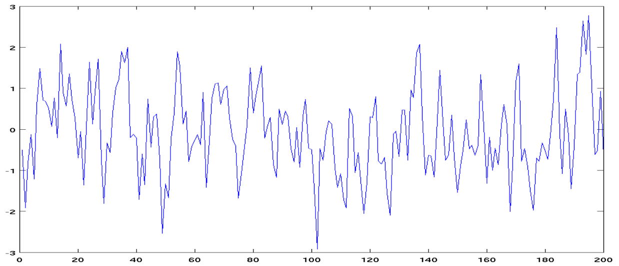
- Hence $E[X(t)] = 0$, and if $|\alpha| < 1$ the process is stationary with

$$\mathrm{Var}[X(t)] = (1 + \alpha^2 + \alpha^4 + \dots)\sigma^2 \qquad = \frac{\sigma^2}{1-\alpha^2}$$

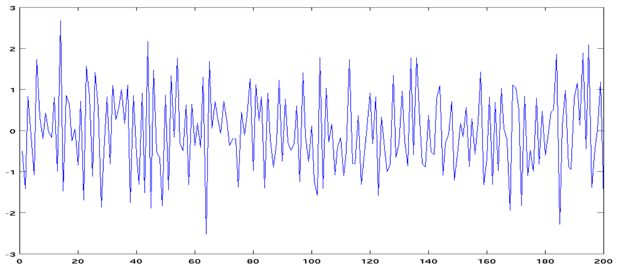
- Similarly

$$\mathrm{Cov}[X(t)X(t-k)] = \alpha^k \mathrm{Var}[X(t-k)] = \frac{\alpha^k \sigma^2}{1-\alpha^2}$$

$\alpha = 0.5$



$\alpha = -0.5$



- The general case is an AR(p) process

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + w_t$$

- Notice how x_t is obtainied by a (linear) regression from $x_{t-1}, \dots, x_{t-p}$, hence an *autoregressive* process
- Introduce the *backward shift* operator B, so that $Bx_t = x_{t-1}$, $B^2 x_t = x_{t-2}$ etc
- Then AR(p) model can be written as

$$\phi(B) x_t = w_t$$

where $\phi(B) = (1 - \alpha_1 B \dots - \alpha_p B^p)$
- The condition for stationarity is that all the roots of $\phi(B)$ lie outside the unit circle

Yule-Walker Equations

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + w_t$$

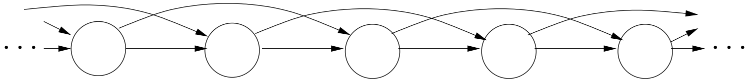
$$x_t x_{t-k} = \sum_{i=1}^{p} \alpha_i x_{t-i} x_{t-k} + w_t x_{t-k}$$

- Taking expectations (and exploiting stationarity) we obtain

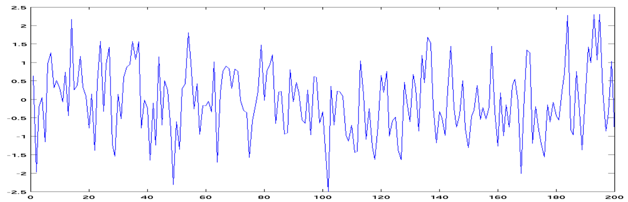
$$\gamma_k = \sum_{i=1}^{p} \alpha_i \gamma_{k-i} \qquad k = 1, \ ,2, \dots$$

- Use p simultaneous equations to obtain the γ's from the α's. For inference, can solve a linear system to obtain the α's given estimates of the γ's
- Example: AR(1) process, $\gamma_k = \alpha^k \gamma_0$

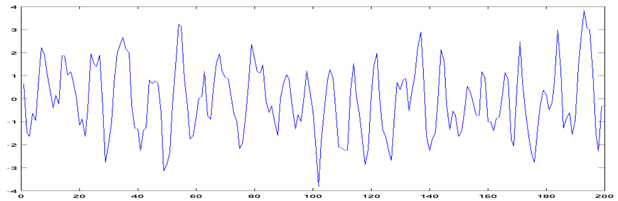
Graphical model illustrating an AR(2) process



AR2: $\alpha_1 = 0.2, \ \alpha_2 = 0.1$



AR2: $\alpha_1 = 1.0, \ \alpha_2 = -0.5$



Vector AR processes

$$\mathbf{x}_t = \sum_{i=1}^{p} A_i \mathbf{x}_{t-i} + G\mathbf{w}_t$$

where the A_is and G are square matrices

- We can in general consider modelling multivariate (as opposed to univariate) time series
- An AR(2) process can be written as a vector AR(1) process:

$$\begin{pmatrix} x_t \\ x_{t-1} \end{pmatrix} = \begin{pmatrix} \alpha_1 & \alpha_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ x_{t-2} \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} w_t \\ w_{t-1} \end{pmatrix}$$

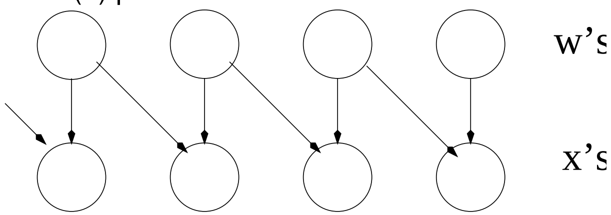
- In general an AR(p) process can be written as a vector AR(1) process with a p-dimensional state vector (cf ODEs)

Moving Average (MA) processes

$$x_t = \sum_{j=0}^{q} \beta_j w_{t-j} \qquad \text{(linear filter)}$$
$$= \theta(B) w_t$$

with scaling so that $\beta_0 = 1$ and $\theta(B) = 1 + \sum_{j=1}^{q} \beta_j B^j$

Example: MA(1) process



- We have $E[X(t)] = 0$, and

$$\mathrm{Var}[X(t)] = (1 + \beta_1^2 + \ldots + \beta_q^2)\sigma^2$$

$$\mathrm{Cov}[X(t)X(t-k)] = E[\sum_{j=0}^{q} \beta_j w_{t-j}, \sum_{i=0}^{q} \beta_i w_{t-k-i}]$$
$$= \begin{cases} \sigma^2 \sum_{j=0}^{q-k} \beta_{j+k}\beta_j & \text{for } k = 0, 1, \ldots, q \\ 0 & \text{for } k > q \end{cases}$$

- Note that covariance “cuts off” for $k > q$

ARMA(p,q) processes

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + \sum_{j=0}^{q} \beta_j w_{t-j}$$

$$\phi(B)x_t = \theta(B)w_t$$

- Writing an AR(p) process as a MA(∞) process

$$\begin{aligned} \phi(B)x_t &= w_t \\ x_t &= (1 - \alpha_1 B \ldots \ \alpha_p B^p)^{-1} w_t \\ &= (1 + \beta_1 B + \beta_2 B^2 \ldots) w_t \end{aligned}$$

- Similarly a MA(q) process can be written as a AR(∞) process
- Utility of ARMA(p,q) is potential parsimony

The Fourier View

- ARMA models are linear time-invariant systems. Hence sinusoids are their eigenfunctions (Fourier analysis)
- This means it is natural to consider the power spectrum of the ARMA process. The power spectrum $S(k)$ can be determined from the $\{\alpha\}$, $\{\beta\}$ coefficients
- Roughly speaking $S(k)$ is the amount of power allocated on average to the eigenfunction $e^{2\pi ikt}$
- This is a useful way to understand some properties of ARMA processes, but we will not pursue it further here
- If you want to know more, see e.g. Chatfield (1989) chapter 7 or Diggle (1990) chapter 4

Parameter Estimation

- Let the vector of observations $\mathbf{x} = (x(t_1), \ldots, x(t_n))^T$
- Estimate and subtract constant offset $\hat{\mu}$ if this is non zero
- ARMA models driven by Gaussian noise are Gaussian processes. Thus the likelihood $L(\mathbf{x}; \boldsymbol{\alpha}, \boldsymbol{\beta})$ is a multivariate Gaussian, and we can optimize this wrt the parameters (e.g. by gradient ascent)
- AR(p) models,

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + w_t$$

  can be viewed as the linear regression of x_t on the p previous time steps, $\boldsymbol{\alpha}$ and σ^2 can be estimated using linear regression
- This viewpoint also enables the fitting of *nonlinear* AR models

Model Order Selection, References

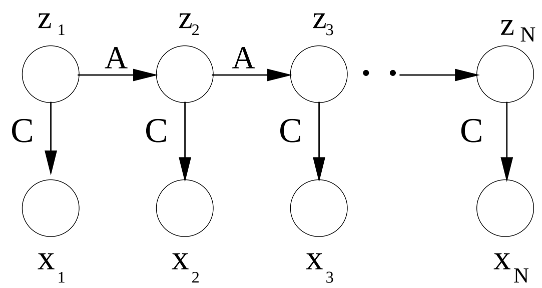
- For a MA(q) process there should be a cut-off in the autocorrelation function for lags greater than q
- For general ARMA models this is model order selection problem, discussed in an upcoming lecture

Some useful books:

- The Analysis of Time Series: An Introduction. C. Chatfield, Chapman and Hall, 4th edition, 1989
- Time Series: A Biostatistical Introduction. P. J. Diggle, Clarendon Press, 1990

Linear-Gaussian HMMs

- HMM with continuous state-space and observations
- Filtering problem known as Kalman filtering



- Dynamical model

$$\mathbf{z}_{n+1} = A\mathbf{z}_n + \mathbf{w}_{n+1}$$

where $\mathbf{w}_{n+1} \sim N(\mathbf{0}, \Gamma)$ is Gaussian noise, i.e.

$$p(\mathbf{z}_{n+1}|\mathbf{z}_n) \sim N(A\mathbf{z}_n, \Gamma)$$

- Observation model

$$\mathbf{x}_n = C\mathbf{z}_n + \mathbf{v}_n$$

where $\mathbf{v}_n \sim N(\mathbf{0}, \Sigma)$ is Gaussian noise, i.e.

$$p(\mathbf{x}_n|\mathbf{z}_n) \sim N(C\mathbf{z}_n, \Sigma)$$

- Initialization

$$p(\mathbf{z}_1) \sim N(\mu_0, V_0)$$

Inference Problem – filtering

- As whole model is Gaussian, only need to compute means and variances

$$p(\mathbf{z}_n|\mathbf{x}_1, \ldots, \mathbf{x}_n) \sim N(\mu_n, V_n)$$

$$\mu_n = E[\mathbf{z}_n|\mathbf{x}_1, \ldots, \mathbf{x}_n]$$
$$V_n = E[(\mathbf{z}_n - \mu_n)(\mathbf{z}_n - \mu_n)^T|\mathbf{x}_1, \ldots, \mathbf{x}_n]$$

- Recursive update split into two parts
- **Time update**

$$p(\mathbf{z}_n|\mathbf{x}_1, \ldots, \mathbf{x}_n) \to p(\mathbf{z}_{n+1}|\mathbf{x}_1, \ldots, \mathbf{x}_n)$$

- **Measurement update**

$$p(\mathbf{z}_{n+1}|\mathbf{x}_1, \ldots, \mathbf{x}_n) \to p(\mathbf{z}_{n+1}|\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1})$$

- Time update
$$\mathbf{z}_{n+1} = A\mathbf{z}_n + \mathbf{w}_{n+1}$$
thus
$$\mathbb{E}[\mathbf{z}_{n+1}|\mathbf{x}_1, \dots \mathbf{x}_n] = A\boldsymbol{\mu}_n$$
$$\mathrm{cov}(\mathbf{z}_{n+1}|\mathbf{x}_1, \dots \mathbf{x}_n) \stackrel{def}{=} P_n = AV_nA^T + \Gamma$$
- Measurement update (like posterior in Factor Analysis)
$$\boldsymbol{\mu}_{n+1} = A\boldsymbol{\mu}_n + K_{n+1}(\mathbf{x}_{n+1} - CA\boldsymbol{\mu}_n)$$
$$V_{n+1} = (I - K_{n+1}C)P_n$$
where
$$K_{n+1} = P_nC^T(CP_nC^T + \Sigma)^{-1}$$
- K_{n+1} is known as the Kalman gain matrix

Simple example

$$z_{n+1} = z_n + w_{n+1}$$
$w_n \sim N(0, 1)$
$$x_n = z_n + v_n$$
$v_n \sim N(0, 1)$
$$p(z_1) \sim N(0, \sigma^2)$$
In the limit $\sigma^2 \to \infty$ we find
$$\mu_3 = \frac{5x_3 + 2x_2 + x_1}{8}$$

- Notice how later data has more weight
- Compare $z_{n+1} = z_n$ (so that w_n has zero variance); then
$$\mu_3 = \frac{x_3 + x_2 + x_1}{3}$$

Applications

Much as a coffee filter serves to keep undesirable grounds out of your morning mug, the Kalman filter is designed to strip unwanted noise out of a stream of data. Barry Cipra, SIAM News 26(5) 1993

- Navigational and guidance systems
- Radar tracking
- Sonar ranging
- Satellite orbit determination

Extensions

Dealing with non-linearity
- The Extended Kalman Filter (EKF)
  If $\mathbf{x}_n = f(\mathbf{z}_n) + \mathbf{v}_n$ where f is a non-linear function, can linearize f, e.g. around $\mathbb{E}[\mathbf{z}_n|\mathbf{x}_1, \dots \mathbf{x}_{n-1}]$. Works for weak non-linearities
- For very non-linear problems use sampling methods (known as particle filters). Example, work of Blake and Isard on tracking, see
  `http://www.robots.ox.ac.uk/~vdg/dynamics.html`

It is possible to train KFs using a forward-backward algorithm