

# Lab Session

## Auto- & cross-regressive models & least-squares solutions

In this session we will be looking into developing *(auto)regressive models*. Recall that the generic form of the model is

$$\hat{y}[t] = \sum_{i=1}^p a_i y[t-i]$$

Recall that we can solve for the unknown set of coefficients,  $\{a_i\}$ , in several ways

1. By forming the *embedding* matrix,  $\mathbf{M}$ , from lagged versions of a data series. As  $\mathbf{M}$ , in general is non-square, we will have to use the pseudo-inverse, i.e. evaluate  $(\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T$ .
2. Estimate the autocorrelation matrix,  $\mathbf{R}$ , and use  $\mathbf{R}^{-1}$  to infer the coefficients.

**Preamble** Load the data, provided at

<http://www.robots.ox.ac.uk/~sjrob/AIMS/LabSession>

This consists of three data samples of financial time series (foreign exchange rates) sampled once per second. These are shown below.

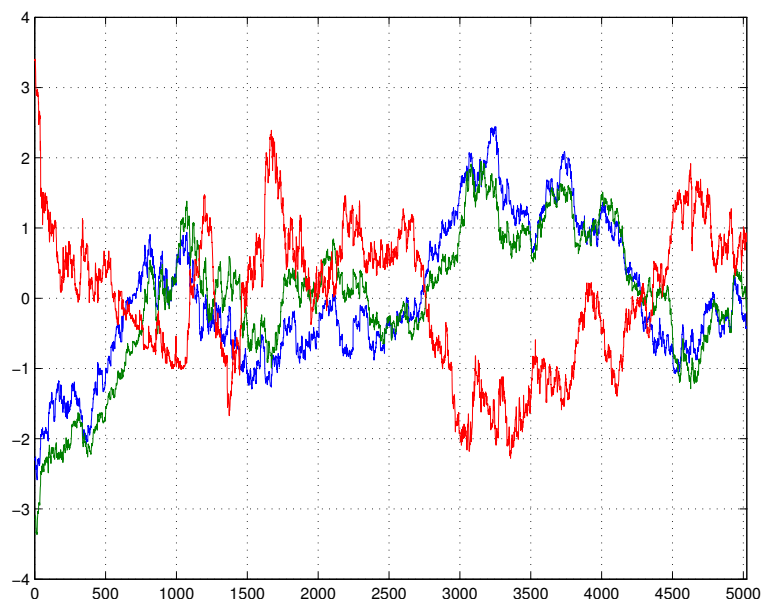


Figure 1: Some financial data.

**Direct 1-d solutions** We start out by looking at one of these time series - up to you which one.

- By evaluating the least-squares solution for  $\mathbf{a}$  using the embedding method, write code to infer the coefficients.
- Re-evaluate the solution for  $\mathbf{a}$  using a direct inversion of the autocorrelation matrix.
- [optional] Use a Toeplitz solver - the Yule-Walker recursions, for example, to evaluate the coefficients rather than a direct inversion of the autocorrelation matrix. For large model orders,  $p \gg 1$ , can you notice a difference in speed?
- As the AR models represent *one-step prediction* models, you can see how well they do at forecasting - would you make money from this?

**Spectral estimates** It is pretty straight-forward now to write code to evaluate the spectrum from the AR coefficients. You may want to experiment with the AR order and see what effect this has on the spectral estimation.

### Cross-regression

- Modify the embedding matrix method, so that

$$\hat{y}[t] = \sum_{i=1}^p a_i z[t-i]$$

in which  $y$  is being modelled by observing another timeseries,  $z$ . The coefficients of this model now describe the cross-regression i.e. how the past of  $z$  effects the present value of  $y$ . We can look at the magnitude of the coefficients as well as the predictions of  $y$  to give an idea about the information that ‘flows’ from  $z$  to  $y$ . Do any of the timeseries have strong interactions? If so, is there any indication of which one is driving which?

- These coefficients give us some information regarding the spectral modes (resonances) in  $y$  which can be ‘explained’ by observing  $z$ .

**Regularisation** [optional] If you have the time, you can look at regularising the solutions of the least-squares linear models. From a *Bayesian* perspective this is equivalent to putting *priors* on the coefficients which penalise large values. A simple method to achieve this is *shrinkage*, which adds ‘jitter’, of magnitude  $\alpha$  say along the leading diagonal of anything we invert - in other words  $\mathbf{A}^{-1}$  becomes  $(\mathbf{A} + \alpha \mathbf{I})^{-1}$ . This has the effect of helping reduce rank-deficiency in  $\mathbf{A}$  and has the knock on, in our system, of *shrinking* the values of  $\mathbf{a}$  towards zero. If you get a chance try it, and see what effect it has. You should be able to obtain coefficients for model orders close to the number of data points without getting singular solutions.