Time Series I – Time Domain Methods

John Fricks

Astrostatistics Summer School
Penn State University
University Park, PA 16802

May 21, 2007

Introduction

- Time series is the study of data consisting of a sequence of DEPENDENT random variables (or vectors).
- This contrasts a sequence of independent observations and regression which studies the dependence of one variable on another.
- In this tutorial, we will discuss models which give rules to generate future observations based on current observations.

Regression Model

A basic regression model is typically written as follows:

$$y_t = \beta_0 + \beta_1 x_t + \epsilon_t$$

for $t = 1, ..., n$ where β_0 and β_1 are fixed coefficients and x_t is a covariate. The sequence ϵ_t are independent and identically distributed normal random variables with variance σ^2.

Autoregressive (AR) Model

A basic time series model related to regression is the Autoregressive Model (AR)

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

for $t = 1, \ldots, n$ where ϕ is a constant and w_t is a sequence of independent and identically distributed normal random variables (often called a white noise sequence in this context).
Note that the result of this model will be a single *dependent* series–a time series, $x_1, \ldots, x_t$. This contrasts the regression model which relates two variables.

Vector Autoregressive Model

An obvious generalization of this model is a vector version

$$\mathbf{x}_t = \Phi \mathbf{x}_{t-1} + w_t$$

for $t = 1, .., n$ where $x_t = (x_{t1}, ..., x_{tp})'$ and w_t is a sequence of independent $p \times 1$ normal random vectors with covariance matrix Q. The matrix Φ is a $p \times p$ matrix.

The Linear State Space Model

Now, imagine that we cannot actually observe our system of interest $\mathbf{x_t}$ which is a Vector Autoregressive Model. Instead we may observe a linear transformation of x_t with additional observational error. In other words, the complete model is as follows:

$$\mathbf{y}_t = A\mathbf{x_t} + v_t \quad , \ \mathbf{x}_t = \Phi \mathbf{x}_{t-1} + w_t$$

where $\mathbf{y}_t$ is a $q \times 1$ vector, A is a $q \times p$ matrix, and v_t is a sequence of independent normal random variables with mean zero and covariance matrix R. Also, the sequence v_t is independent of the sequence w_t. The equation on the left is generally called the *observation equation*, and the equation on the right is called the *system equaion*.

What can we do with the state space model?

- Maximum Likelihood estimation of the parameters (including standard errors of our estimates).
- Bayesian estimation of parameters.
- Filtering–conditional distribution of the systems given our observations. We will, therefore, have a "guess" of our unseen system, x_t given our observations y_t.
- Prediction–predict the next observation given the observations up to the current time.

Filering

Suppose you may observe $y_1, ..., y_n$, but you are really interested in $x_1, .., x_n$ and estimating parameters such as ϕ. While you cannot "know" x_t, you can have an optimal estimate of x_t. The goal will be to calculate

$$p(x_t | y_t, ..., y_1)$$

For a Gaussian model this means that you'll know $E(x_t | y_t, ..., y_1)$ (your guess) and also the conditional variance of x_t.

Steps for Filtering

Here's an outline of how that works–assume that you know all the parameters. Assume that you have the guess at the last time step, i.e. $p(x_{t-1}|y_{t-1},...,y_1)$.

1. **Predict** the next system observation based on what you have.

$$p(x_t|y_1,...,y_{t-1} = \int p(x_t|x_{t-1})p(x_{t-1}|y_{t-1},...,y_1)dx_{t-1}$$

2. Calculate the guess for the observation, y_t, based on this prediction.

$$p(y_t|y_{t-1},...,y_1) = \int p(y_t|x_t,y_{t-1},...,y_1)p(x_t|y_{t-1},...,y_1)dx_t$$

3. Use Bayes rule to **update** the prediction for x_t with the current observation y_t

$$p(x_t|y_t,...,y_1) = \frac{p(x_t|y_{t-1},...,y_1)p(y_t|x_t,y_{t-1}...,y_1)}{p(y_t|y_{t-1},...,y_1)}$$

The Likelihood Function

- Remember that the likelihood function is simply the density of the data evaluated at the observations.

$$p(y_T, ..., y_1) = \Pi_{t=1}^{T} p(y_t | y_{t-1}, ..., y_1)$$

  Now, we have a likelihood to maximize to obtain parameters such as ϕ.
- When the errors are Gaussian, finding $p(x_t | y_t, ..., y_1)$ for each t is known as calculating the Kalman filter. In general, this density is called the filter density. These calculations are reduced to matrix operations in the linear Gaussian case.

The ARMA Model

- The ARMA model is the most basic time series model. It can be fit with every major statistical package.
- The advantage of the ARMA model is that it can be used to efficiently fit many, if not most, stationary time series.
- The disadvantage is that it is often not very descriptive of the underlying processes; the parameters are difficult to interpret.

Autoregressive (AR) Model of order p

We have already seen a basic AR(1) model. We can extend the AR model to include more previous observations. The AR(p) model is as follows:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + w_t$$

Autoregressive (AR) Model of order p

Note that the AR(p) model can be expressed as a single dimension of a multivariate AR(1) process.

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + w_t$$

$$\begin{pmatrix} x_t \\ x_{t-1} \\ x_{t-2} \\ \ldots \\ x_{t-p+1} \end{pmatrix} = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \ldots & \phi_p \\ 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 1 & \ldots & 1 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ x_{t-2} \\ x_{t-3} \\ \ldots \\ x_{t-p} \end{pmatrix} + w_t$$

where w_t has covariance $p \times p$ matrix with σ^2 as the first entry and zeros otherwise.

Moving Average Model

Another basic time series model is the Moving Average Model. The MA(q) model is as follows:

$$x_t = w_t + \theta_1 w_{t-1} + \ldots + \theta_q w_{t-q}$$

This model is the moving average across a window of size $q+1$. Note that this is quite different than the Autoregressive model expand on this. The autoregressive model is recursive which leads to "memory" that falls off over lags longer than the order of the model. The moving average model has no dependency for observations that do not have overlapping windows.

ARMA model

These two models, Autoregressive (AR) and Moving Average (MA) can be combined into an ARMA(p,q)model:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + w_t + \theta_1 w_{t-1} + \ldots + \theta_q w_{t-q}$$

How can we fit such a model? How do we select a good model? What are the steps if we are handed data that follows such a model?

ARMA model using the Backshift Operator

There is an alternative way to write an ARMA model. This is done with the backshift operator. We use the symbol B to denote moving back one time step.

$$Bx_t = x_{t-1}$$

This allows us to write the ARMA model as

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

where

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$$

and

$$\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \ldots + \theta_q B^q$$

This is going to help us write the ARMA model in other forms, and to write extensions the model such as SARIMA.

ARMA as a special case of the linear state space model.

We have already seen that an AR(p) can be written as the system equation of a linear state space model. How can we incorporate the MA(q) part of the model. This can be seen using the backshift operator formulation. Let

$$y_t = \theta(B)x_t.$$

Since x_t is AR(p), it can be represented as

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

Putting these together, we obtain

$$\phi(B)y_t = \theta(B)w_t.$$

ARMA as a special case of the linear state space model.

Putting these things together we obtain

$$y_t = (1 \ \theta_1 \ \ldots \ \theta_r) x_t$$

and

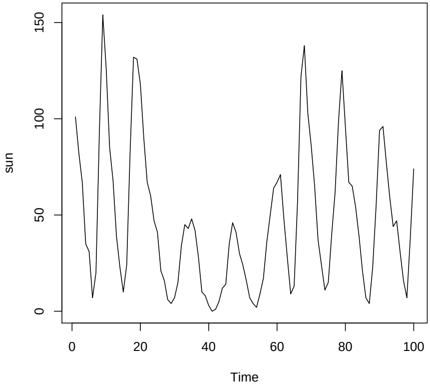
$$\begin{pmatrix} x_t \\ x_{t-1} \\ x_{t-2} \\ \ldots \\ x_{t-r+1} \end{pmatrix} = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \ldots & \phi_r \\ 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 1 & \ldots & 1 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ x_{t-2} \\ x_{t-3} \\ \ldots \\ x_{t-p} \end{pmatrix} + w_t$$

where r is the maximum of p and q and any undefined parameters are set to zero.

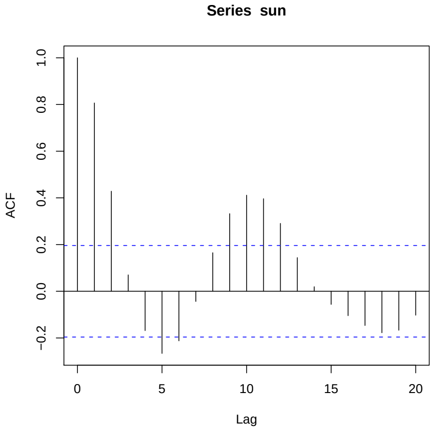
ARMA as a special case of the linear state space model.

- The important thing to note is that the ARMA model is simply a special case of the linear state space model and, therefore, requires no additional computational methods.
- One implication of this is that assuming that we observe an ARMA process with additional observational error can be easily incorporated.
- Missing data can be easily incorporated into this framework.

Wolfer sunspots 1770-1869



ACF of Sunspot Data



What is the ACF?

The ACF is a plot of the sample autocorrelation function, $\hat{\rho}(h)$, which is defined as

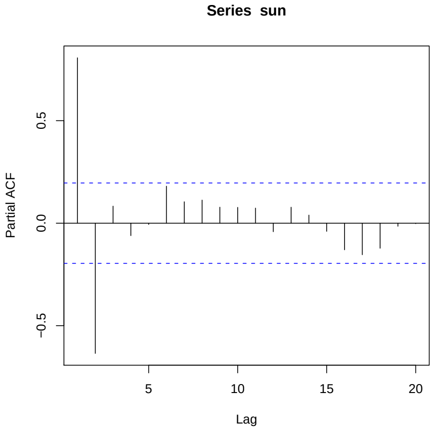
$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}$$

where

$$\hat{\gamma}(h) = \frac{\sum_{t=1}^{n-h}(x_{t+h} - \bar{x})(x_t - \bar{x})}{n}.$$

which is the sample autocovariance function. This is an estimator for the correlation between x_t and x_{t-h}. For an MA(q) process, this ACF should cut off after q lags.

PACF of Sunspot Data

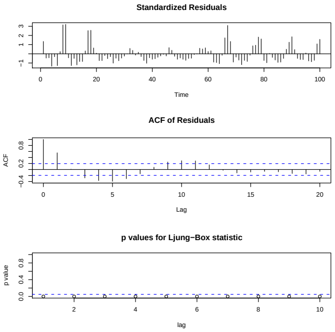


What is the PACF?

- The PACF is a plot of the sample partial autocorrelation function.
- This plot estimates the correlation between the x_t and the x_{t-h} with the linear effect of the intermediate observations, $x_{t-1}, \ldots, x_{t-h+1}$, removed.
- The algorithm to do this (Durbin-Levinson) is a little complicated, but can be done quickly.
- For an AR(p) model, the PACF should cut off after the pth lag.

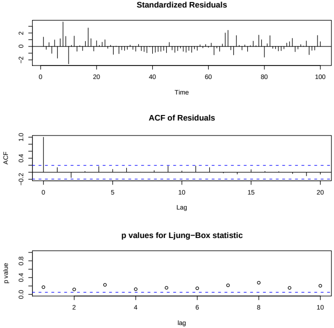
Fit 1

```
arima(x = sun, order = c(1, 0, 0))
         ar1  intercept
      0.8199    50.4814
s.e.  0.0568    11.4640
sigma^2 estimated as 459.9:  log likelihood = -449,  aic = 904
```



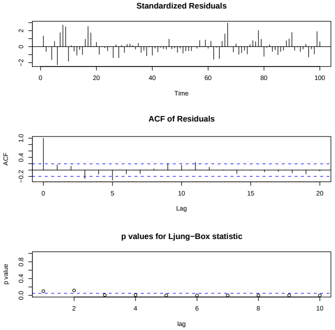
Fit 2

```
arima(x = sun, order = c(2, 0, 0))
          ar1      ar2  intercept
       1.4067  -0.7117    48.3502
s.e.   0.0705   0.0701     4.9705
sigma^2 estimated as 228.7:  log likelihood = -414.79,  aic = 837.58
```



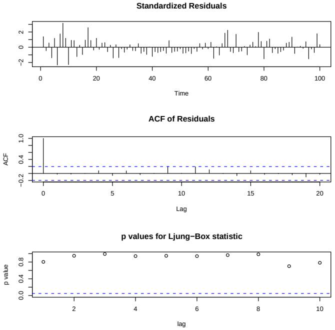
Fit 3

```
arima(x = sun, order = c(1, 0, 1))
         ar1     ma1  intercept
      0.7239  0.7523    50.2528
s.e.  0.0726  0.0681     9.9355
sigma^2 estimated as 257.8:  log likelihood = -420.73,  aic = 849.46
```



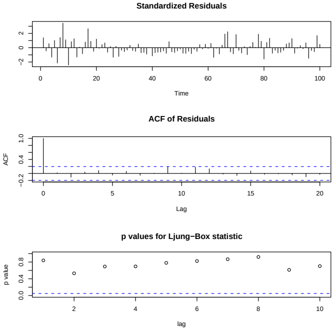
Fit 4

```
Call:
arima(x = sun, order = c(2, 0, 1))
         ar1      ar2     ma1  intercept
      1.2241  -0.5591  0.3844    48.6226
s.e.  0.1125   0.1078  0.1321     6.0308
sigma^2 estimated as 214.5:  log likelihood = -411.67,  aic = 833.35
```



Fit 5

```
Call:
arima(x = sun, order = c(3, 0, 0))
         ar1      ar2     ar3  intercept
      1.5528  -1.0018  0.2073    48.6030
s.e.  0.0981   0.1543  0.0989     6.0927
sigma^2 estimated as 218.9:  log likelihood = -412.65,  aic = 835.29
```



The ARIMA Model

- The first important extension of the model is the ARIMA model (I is for integrated.). The assumption here is that the data will follow an ARMA model after differencing.
- There is a tacit assumption that an ARMA model is stationary, i.e. that the dependency between x_t and x_{t-h} depends only on lag, h. (Restrictions must be put on the parameters of ARMA models to guarantee stationarity.)
- Differencing also eliminates unwanted trends.

An ARIMA(p,d,q)

$$\phi(B)\nabla^d x_t = \theta(B)w_t$$

where

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p,$$

$$\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \ldots + \theta_q B^q,$$

and

$$\nabla = 1 - B$$

The SARIMA model

- The SARIMA model is the seasonal ARIMA model.
- The SARIMA model allows us to model dependency between nearby observations and also across "seasons". For example, the temperature in January could depend as much on last January's temperature as it does on December's temperature.
- Note that these models are useful when a KNOWN and FIXED season is to be modeled.

An $ARIMA(p,d,q) \times (P,D,Q)_s$ Model

$$\Phi(B^s)\phi(B)\nabla_s \nabla x_t = \Theta(B)\theta(B)w_t$$

where

$$\phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^p \ , \ \theta(B) = 1 + \theta_1 B + \ldots + \theta_q B^q$$

and

$$\Phi(B) = 1 - \Phi_1 B - .. - \Phi_P B^P \ , \ \Theta(B) = 1 + \Theta_1 B + \ldots + \Theta_q B^q.$$

Also,

$$\nabla_s = 1 - B^s \ , \ \nabla = 1 - B.$$

Other Models

- Extended Kalman Filter. What if our model is not linear? Using Taylor expansions, we can approximate the non-linear model with a linear model.
- Particle Filters–We can use increased computational capabilities to use simulation to compute filters and likelihood functions via simulation. This is especially useful for Bayesian analysis but can be used for likelihood.
- There are a number of other time series that are more tractable such as ARCH which allows for heteroskedastic error terms.

Research

- Robert Shumway and David Stoffer. *Time Series Analysis and Its Applications.* Springer NY, 2006.
- Peter Brockwell and Richard Davis. *Time Series: Theory and Methods, Second Ed.* Springer NY, 1991.
- G.E.P. Box and G.M. Jenkins. *Time Series Analysis, Forecasting, and Control.* Oakland, CA: Holden Day, 1970.