Fitting an Autoregressive Model

Elias Putzig

07/13/2018

1 Mathematical Approach

Given time series data, we would like to fit an autoregressive (AR) model of order p ($AR(p)$), which is of the following form.

$$x_t = \sum_{\tau=1}^{p} K_\tau x_{t-\tau} + \epsilon_t \tag{1}$$

The goal is to fit the coefficients K_τ and the variance (σ_ϵ^2) of the white noise. White noise has the following form: $\langle \epsilon_t \epsilon_{t+k} \rangle = \delta_{k,0}\sigma_\epsilon^2$, where $\delta_{t,t'}$ is the Kronecker delta, and$\langle\rangle$ is used to denote an expected value.

The Yule-Walker Equations

In order fit the coefficients, we shall use the Yule-Walker equations, which describe the relationship between coefficients K_τ, and autocorrelations at lag k:

$$c_k = \langle x_t x_{t+k} \rangle$$

Let us first write x_{t+k} in the form of Eq. 1, multiply by x_t, then take the expected value.

$$x_t x_{t+k} = \sum_{\tau=1}^{p} K_\tau x_t x_{t+k-\tau} + x_t \epsilon_{t+k}$$

$$\langle x_t x_{t+k} \rangle = \sum_{\tau=1}^{p} K_\tau \langle x_t x_{t+k-\tau} \rangle + \langle x_t \epsilon_{t+k} \rangle$$

The last term in this expression can be expanded as

$$\langle x_t \epsilon_{t+k} \rangle = \sum_{\tau=1}^{p} K_\tau \langle \epsilon_{t+k} x_{t-\tau} \rangle + \langle \epsilon_{t+k} \epsilon_t \rangle = 0 + \delta_{k,0}\sigma_\epsilon^2$$

where we have used the fact that x_t is not correlated with noise from a later time to drop all terms under the sum. This means that we are working with

$k \geq 0$ from now on. This is slightly counterintuitive as $c_k = c_{-k}$, but it is necessary to set the difference $k - \tau$.

Coming back to the autocorrelations,

$$\langle x_t x_{t+k} \rangle = \sum_{\tau=1}^{p} K_\tau \langle x_t x_{t+k-\tau} \rangle + \delta_{k,0} \sigma_\epsilon^2$$

we can now begin to write a few terms, in order to see the general form.

$$\begin{aligned} c_0 = \langle x_t^2 \rangle &= \sum_{\tau=1}^{p} K_\tau \langle x_t x_{t-\tau} \rangle + \sigma_\epsilon^2 \\ &= \sum_{\tau=1}^{p} K_\tau c_\tau + \sigma_\epsilon^2 \end{aligned}$$

$$\begin{aligned} c_1 = \langle x_t x_{t+1} \rangle &= \sum_{\tau=1}^{p} K_\tau \langle x_t x_{t+1-\tau} \rangle \\ &= K_1 c_0 + K_2 c_1 + \ldots + K_p c_{p-1} \end{aligned}$$

$$\begin{gathered} c_2 = \langle x_t x_{t+2} \rangle = \sum_{\tau=1}^{p} K_\tau \langle x_t x_{t+2-\tau} \rangle \\ = K_1 c_1 + K_2 c_0 + K_3 c_1 + K_4 c_2 \ldots + K_p c_{p-2} \end{gathered}$$

The general expression form, the Yule-Walker equations, can be expressed as follows:

$$c_k = \sum_{\tau=1}^{p} K_\tau c_{|k-\tau|} + \delta_{k,0} \sigma_\epsilon^2 \tag{2}$$

Inverting the Yule Walker Equations

One may invert the Yule-Walker equations by separating c_0, and writing the expression for $k = 1$ to p in terms of $\gamma_k = c_k / c_0$. If we define the vectors $\gamma = [\ \gamma_1 \quad \gamma_2 \quad \gamma_3 \quad \ldots \quad \gamma_p\ ]^T$ and $K = [\ K_1 \quad K_2 \quad K_3 \quad \ldots \quad K_p\ ]^T$ we can write $\gamma = \boldsymbol{A}_\gamma K$ where $[\boldsymbol{A}_\gamma]_{i,j} = \gamma_{|i-j|}$, or less compactly

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \vdots \\ \gamma_p \end{bmatrix} = \begin{bmatrix} \gamma_0 & \gamma_1 & \gamma_2 & \cdots & \gamma_{p-1} \\ \gamma_1 & \gamma_0 & \gamma_1 & & \gamma_{p-2} \\ \gamma_2 & \gamma_1 & \ddots & & \vdots \\ \vdots & & & & \gamma_1 \\ \gamma_{p-1} & \gamma_{p-2} & \cdots & \gamma_1 & \gamma_0 \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ \vdots \\ K_p \end{bmatrix}$$

The coefficients K_τ can then be expressed in terms of autocorrelations at lag k by inverting $\boldsymbol{A}_\gamma$.

$$K = \boldsymbol{A}_\gamma^{-1} \gamma \tag{3}$$

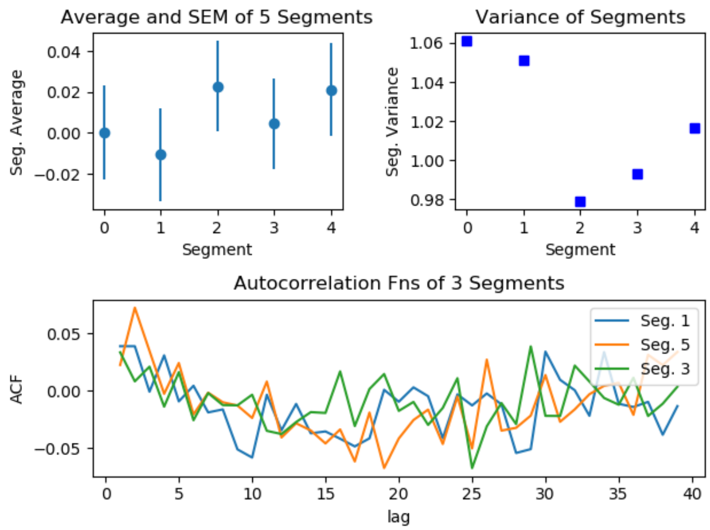

Figure 1: This figure shows the mean with standard error (top left), variance (top right), and autocorrelation function (bottom) of different segments of the data provided.

The variance of the noise can be found once we have the coefficients.

$$\sigma_\epsilon^2 = c_0\big(1 - \sum_{\tau=1}^{p} K_\tau \gamma_\tau\big) \tag{4}$$

2 Data Analysis

The first step in evaluating a time series is typically to check whether it is stationary. This can be done by breaking the series up into pieces, and seeing if statistical such as the mean, variance, and autocorrelation function remain the same throughout time.

We shall proceed from here using some sample data; a time series of 10000 points. I broke this data into 5 segments of 2000 points each, and the aforementioned statistics are shown in Fig. 1. This time series seems to be stationary, as the error-bars on the averages overlap, and the variance only changes by a few percent. The autocorrelation functions seem to agree within their error, but they are rather noisy.

2.1 Box-Jenkins Method and the AC

Since the time series seems to be stationary, we can proceed to fit our autoregressive model. In the following, we will use the Box-Jenkins Method; first checking that the model is stationary, then making sure that there are sufficient

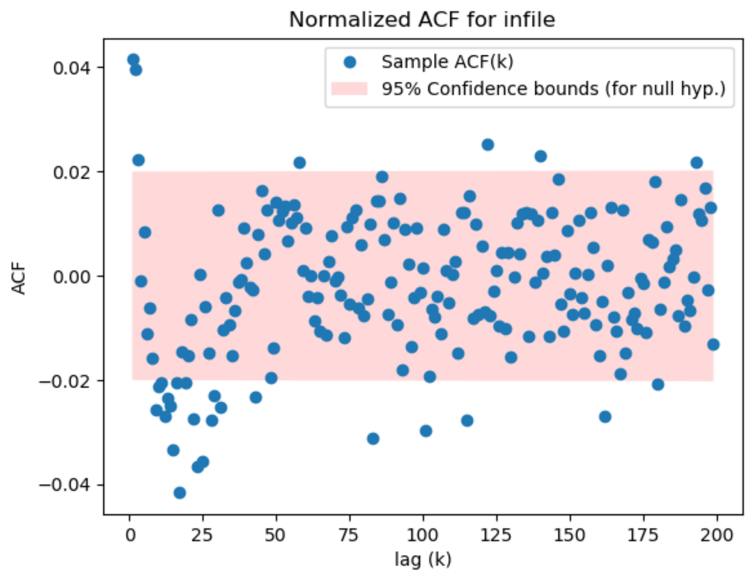

Figure 2: This plot shows the autocorrelation function for the sample data from lag $k = 1$ (c_1) to 200, normalized by the variance in the data (c_0). The red region shows the approximate error bounds for a 95% confidence interval for autocorrelations of white noise. The bounds are at $\pm 2/\sqrt{N-k}$ where $N = 10000$ is the number of points in the sample data, and $N - k$ is the number of points used for calculation of c_k.

correlations in values at different lag times (c_k for lag k) to justify fitting an AR model. Finally, we check the order (p) of the AR model using the sample partial autocorrelation function (PACF).

The autocorrelation function (ACF, see Fig. 2) shows that there are significant correlations in the data. This can be seen as many points exceed the 95% confidence bounds for the amplitude of the correlations in white noise. The correlations are not very strong, however, as they only exceed these bounds by about a factor of 2. Still, this justifies the attempt to fit an AR model, so we shall move on to the discussion of the partial autocorrelation function.

2.2 The PACF

The PACF at lag k ($PACF(k)$) gives the correlation between points that are k steps apart in a time series, with the effects of lags 1 to $k-1$ are accounted for. For an AR model, $PACF(k) = K_{k,k}$, where $K_{k,k}$ is the k'th coefficient (K_k) of an autoregressive model of order k ($AR(k)$, sorry about all the ks). We shall therefore calculate this by fitting an the coefficients of $AR(k)$ for each value k using Eqs. 3 and 4, then using the last coefficient, K_k, for our point $PACF(k)$.

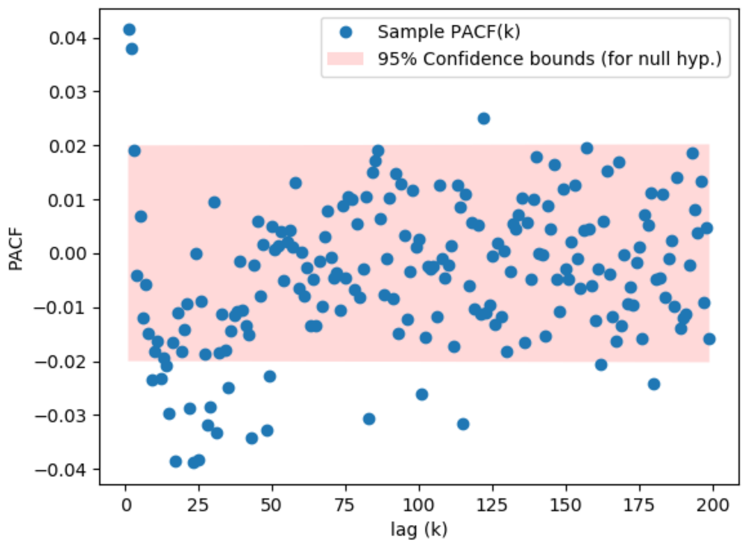

Figure 3: This plot shows the partial autocorrelation function (PACF) of the sample data for lag $k = 1$ to 200. The error bounds are the same as described in Fig. 2. The PACF does not stay within the error bounds, even at large k, which indicates that there are significant correlations at large lags, even with the effect of smaller lags removed.

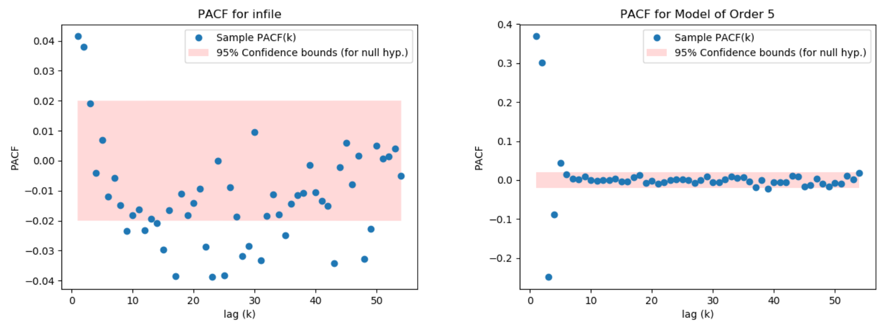

Figure 4: The plots above show the PACE for the sample data ("insample-data.csv," left) and for an autoregressive model of order 5 (right).

There is a faster way to do this (Levinson–Durbin recursion), but inverting the Yule-Walker equations repeatedly for higher order models is fast enough for the following analysis.

Since the PACF at point k shows whether points that are k spaces apart (lag k) in a time series are correlated even with lower lags accounted for, it can be used to indicate whether an AR model of order k is justified. If this correlation ($PACF(k)$) is greater than that which would be expected from white noise, then it is justified. For a model of order p ($AR(p)$) the PACF should stay mainly within the error bounds for $k > p$. This behavior can be seen in the data generated by an $AR(5)$ model in Fig. 4 (on the right), but the data provided does not seem to behave as well. In Fig. 3 it looks like a reasonable cuttoff might be $k = 50$, however the PACF does not stay well within the error bounds for any range of k that can reasonably be measured with this data.

2.3 Checking the Fit

Since our PACF plots did not clearly indicate an order for the AR model, we shall check the accuracy of our predictions. A simple way to do this is to break the sample data (not provided) into a training set and a test set. The training set can then be used to fit the coefficients of the AR model, and the test set can be used to measure the accuracy of the predictions.

I broke the sample data into a 9000 point block to fit the AR model, and used the remaining 1000 points to test the fit. The mean-squared error in the predictions from the test set are shown for model orders from order $p = 0$ to 100 in Fig. 5. The AR model does not do a much better job of fitting the data than white noise ($p = 0$). It is minimum at $p = 35$, so the best course of action seems to be to use an $AR(35)$ fit, where it beats out white noise by about 2%.

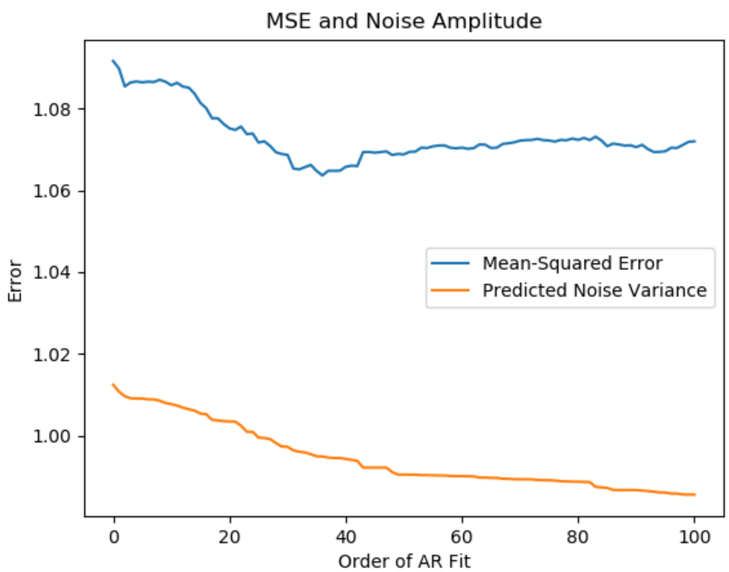

Figure 5: The mean-squared error (MSE) and predicted noise variance (σ_ϵ^2) for AR model fits ($AR(p)$) of the sample data from order $p = 0$ to 100. The fit was generated with a test set of the first 9000 points in the sample data, and the error was measured with the last 1000 points.

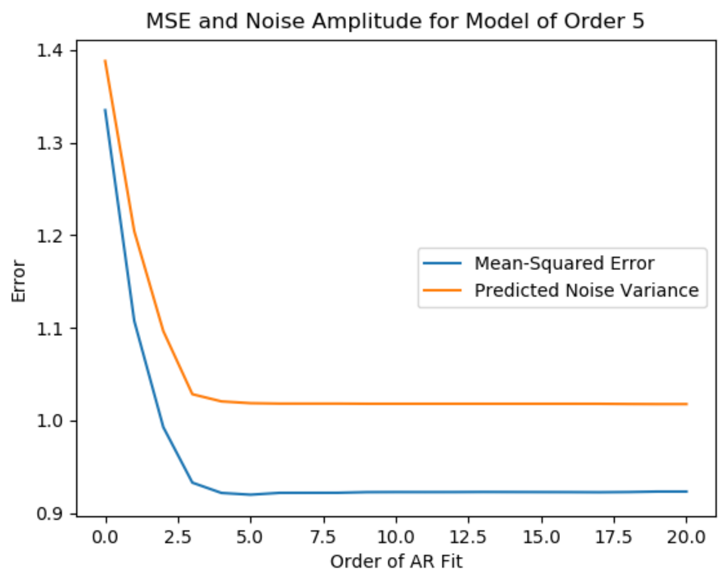

Figure 6: This plot shows the error in the fit of an $AR(5)$ model, for the sake of comparison. This was measured in the same way it was for the sample data shown in Fig. 5, and was used to make sure that the measurement of error vs model order was working properly.