25 Spring 439/639 TSA: Lecture 13

Dr Sergey Kushnarev

Table of contents

Extended Autocorrelation Function (EACF)

Extended Autocorrelation Function (EACF) can help us find the order (p, q) of ARMA(p, q) model. (*Remark:* in comparison, we can use (sample) ACF for MA(q) and PACF for AR(p).)

The sample EACF can be computed by the TSA package in R. (No Python analogue.)

Idea. We first note a simple fact: suppose a process (Y_t) is ARMA, then applying the (true) AR part filter to (Y_t) gives an MA process. This fact suggests a very natural idea: to test whether a process is ARMA given its observations, we can (i) first try to fit an AR regression using the observed data, (ii) then test if the residuals follow an MA process. The only issue of this idea is we do not know the order p and q for AR and MA part, so we need to try different pairs of (p, q) and repeat this two-step idea.

Let's take a look at the algorithm to compute sample EACF in R. By its default, it tries p up to 7 and q up to 13.

(The outer loop) For $p = 0, 1, ..., 7$: fit an AR(p) for (Y_t) and find the residuals $W_t = Y_t - \widehat{Y_t}$.

(The inner loop) For $q = 0, 1, ..., 13$: fit an MA(q) for (W_t) and find the residuals $e_t = W_t - \widehat{W_t}$. Now do hypothesis testing. If (e_t) is white noise, output 0; If (e_t) is not white noise, output $\times$.

The output of the sample EACF is listed in a table of 0's and $\times$'s. And all the zeroes in the table are in a "triangle" area. This is because

- For an ARMA(p, q) process, we can always overfit it with an ARMA(p, q') model for any $q' > q$.
- For an ARMA(p, q) process, we can always overfit it with an ARMA$(p + 1, q + 1)$ model.

So the upper-left corner of these zeroes reflects the correct (p, q) order.

Unit Root Test: Augmented Dickey–Fuller Test

The Unit Root Test (of AR polynomial) is also named as Augmented Dickey–Fuller Test (ADF test). It tests the stationarity of a time series given its observed samples.

Suppose we want to do the following hypothesis testing

$$H_0 : \text{AR polynomial has a unit root} \quad \text{vs.} \quad H_a : \text{AR polynomial does not have a unit root}$$
$$\text{or} \quad H_0 : (Y_t) \text{ is nonstationary} \quad \text{vs.} \quad H_a : (Y_t) \text{ is stationary}$$

This can be done by the command $adf.test()$ in R. A small p-value of this test implies stationarity, and a large p-value implies non-stationarity.