EFRP Econometrics with R project
Description and documentation

Szentistványi János

The problem: The code focuses on the problem of reliable model fitting on an ARMA(p,q) type time series model with noise. Algorithms that try to find the best-fitting ARMA(p,q) model for a given time series based on AIC or BIC information criteria can often return the wrong model, mostly due to the noise that is present in the series, small sample size, or not well specified parameters for the fitting function. In this code, I simulate 4 different ARMA models with different sample size, AR and MA coefficients, order (p and q of the ARMA), and normally distributed noise for the time series. All simulations generate the time series 1000 times.

Data generating processes: The first DGP is an ARMA(2,3) process of length 1000 with noise 0.1, coefficients of AR=(0.4, -0.3) and MA=(-0.2, 0.4, 0.3). The second process in the same specification, except for the noise, which is 0.5 in this case: this way we can compare the accuracy difference between the noise parameters. The third model is a rather simple ARMA(2,2) process of length 500, noise = 0.25, coefficients AR=(0.6, -0.3) and MA=(0.5, 0.25). The last model is a complex one, an ARMA(4,4) with noise = 0.2, length = 750. In the fitting part, the auto.arima function uses maximum possible values of the AR and MA orders, which are in my examples given as AR=(3,3,4,5) and MA=(3,3,4,4) for each model respectively. In lines 7 to 22 of the code, the number and parameters of the simulated models can be extended by adding elements to the lists: the length of each list must match the nModels parameter.

Results: The results show that even in less complex models, the accuracy of the AIC and BIC best fit is low (58% max in my sample). For more complicated models, or less sample size or for higher noise, the accuracy seems to decrease even more. In the 4th, most complex ARMA(4,4) model, the AIC shows a 5% accuracy rate.

Further analysis: Further analysis is possible with the inclusion of more models, which are mostly the same except for one parameter: this way the effect of one parameter change could be estimated. The accuracy of finding the best fitted model could be improved with assumptions on the model parameters before running the function. For example, if max p and max q are 5 and 5 in auto.arima, while the model is an ARMA(2,2) the results will be less accurate than with lower max p and max q values.