Seminar 9: ACF, PACF and ARMA models

Gaetan Bakalli

November 24^{th}, 2020

Correction of HW8

From the property, of the distribution of μ and σ we have:

$$\Sigma = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^4 \end{pmatrix}.$$

Then, using the hint we obtain:

$$\frac{\partial g}{\partial \theta} = \begin{pmatrix} \frac{1}{\sqrt{\sigma^2}} & -\frac{1}{2}\mu\sigma^{-3} \end{pmatrix}.$$

where $g = \frac{\mu}{\sqrt{\sigma^2}}$

Correction of HW8

Hence:

$$\frac{\partial g}{\partial \theta}\Sigma\frac{\partial g'}{\partial \theta} = \begin{pmatrix} \frac{1}{\sqrt{\sigma^2}} \\ -\frac{1}{2}\mu\sigma^{-3} \end{pmatrix}\begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^4 \end{pmatrix}\begin{pmatrix} \frac{1}{\sqrt{\sigma^2}} & -\frac{1}{2}\mu\sigma^{-3} \end{pmatrix}$$
$$= 1 + \frac{1}{4}\left(\frac{\mu}{\sigma}\right)^2.$$

Denoting by $z_{\alpha/2}$ and $z_{1-\alpha/2}$ the $\alpha/2$ and $1-\alpha/2$ quantiles of $\mathcal{N}(0, 1 + \frac{1}{4}(\frac{\hat{\mu}}{\hat{\sigma}})^2)$, and using the delta method, we can write:

$$\mathrm{P}\left(z_{\alpha/2} \leq \sqrt{T}(g(\hat{\theta}) - g(\theta)) \leq z_{1-\alpha/2}\right) = 1 - \alpha.$$

Rearranging:

$$\mathrm{P}\left(\widehat{SR} - \frac{z_{1-\alpha/2}}{\sqrt{T}} \leq SR \leq \widehat{SR} - \frac{z_{\alpha/2}}{\sqrt{T}}\right) = 1 - \alpha,$$

which gives the asymptotic confidence interval for SR.

Delta methods 95% confidence intervals (Solutions)

Solution

- Define the variance of the asymptotic distribution of *SR* as `VAR = 1 + 0.25*(mu_hat/sigma_hat)**2`
- Compute **z = norm.ppf(0.975, 0, VAR**0.5)** from a normal PDF.
- `delta_method = [SR_hat - z/np.sqrt(T), SR_hat + z/np.sqrt(T)]`

HW8

Common mistakes

- Square of T **deltamethod = [SRobs - z/(math.pow (T, 2)), SRobs + z/(math.pow (T, 2))]**.
- Not taking the square-root of 16 for **np.random.normal(loc=1, scale=16, size=T)**.
- Doing the derivatives of SR with symbolic toolbox in Python. Better to do it by hand.
- Use MLE estimator instead of MM. Easier the second one.
- Using .95 as quantile instead of .975 like this:**norm.ppf(0.95, mu, math.sqrt(**`sigma_2`**))**. CI are two tailed.
- For delta method CI, not using the sandwich formula $\frac{\partial g}{\partial \theta} \Sigma \frac{\partial g'}{\partial \theta}$ as the asymptotic variance.

Results must make sense

```
In [59]: #use conf_interval command
         conf_interval = np.percentile(array2,[2.5,97.5])
         print(conf_interval)

         [-5.90513041  8.80310158]
```

Notion of Dependence

Covariance (or correlation) $\quad$ = measure of *linear dependence* between two random variables. Will be used to measure dependence between current observations and past observations. Current return: Y_t Return with lag τ: $Y_{t-\tau}$ The covariance between current and lagged return

$$\gamma(\tau) = Cov(Y_t, Y_{t-\tau})$$

is called the *autocovariance of order* τ.

Autocorrelation

The *autocorrelation function (ACF)* $= \rho(\tau), \quad \tau = 1, 2, ...$ of successive orders will allow to detect linear temporal dependence.

Autocorrelation of order τ

$$\rho(\tau) = \frac{Cov(Y_t, Y_{t-\tau})}{V(Y_t)} = \frac{\gamma(\tau)}{\gamma(0)}$$

ARMA process

Pure AR

A pure Auto-Regressive (AR) process of order 1 is defined as followed:

$$Y_t = \mu + \omega_1 Y_{t-1} + \varepsilon_t$$

where ε_t is a noise term. Hence we can define an AR of order p as:

$$Y_t = \mu + \omega_1 Y_{t-1} + ... + \omega_p Y_{t-p} + \varepsilon_t.$$

Pure MA

A pure Moving-Average (MA) process of order 1 is defined as followed:

$$Y_t = \mu + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \varepsilon_{t-q}$$

where ε_t is a noise term. Hence we can define an MA of order q as:

$$Y_t = \mu + \varepsilon_t - \theta_1 \varepsilon_{t-1} - ... - \theta_q \varepsilon_{t-q}$$

ARMA process

ARMA (p, q)

Hence is it now easy to see that Y_t can be represented a combination of pure AR(p) and and MA(q) process in the following manner:

$$Y_t = \mu + \omega_1 Y_{t-1} + ... + \omega_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - ... - \theta_q \varepsilon_{t-q}.$$

Order p and q detection

Reminder: Autocorrelation

$$\rho(\tau) = \frac{Cov(Y_t, Y_{t-\tau})}{V(Y_t)} = \frac{\gamma(\tau)}{\gamma(0)}$$

Partial autocorrelation

The partial autocorrelation $r(\tau)$ is defined as the last coefficient in the autoregression of order τ,

$$Y_t = \mu + \omega_1 Y_{t-1} + ... + \omega_\tau Y_{t-\tau} + \varepsilon_t,$$

i.e., $r(\tau) = \omega_\tau$.

Order p and q detection

Order detection

if pure MA(q), autocorrelations $\rho(\tau)$ are zero after order q,

$$\rho(\tau) = 0, \quad \tau > q$$

if pure AR(p), partial autocorrelations $\rho(\tau)$ are zero after order p,

$$r(\tau) = 0, \quad \tau > p$$

Detection?

- Plotting $r(\tau)$ and $\rho(\tau)$ function for orders $\tau = 1, 2, ...$
- Assess if $r(\tau)$ and $\rho(\tau)$ are significantly $\neq 0$ trough confidence intervals.

HW9

Usefull functions

- Import **import statsmodels.api as sm** and use `sm.graphics.tsa.plot_acf` to plot the *acf*.
- Repeat same procedure for *pacf*.
- Import **import from statsmodels.stats.diagnostic import** `acorr_ljungbox` and use `acorr_ljungbox` to compute the Ljung-Box test
- **ols** function to fit the models and **resid** to extract the residuals and plot their *acf* and *pacf*.