**White Noise** = ARIMA(0,0,0)

* Fixed, constant mean
* Fixed, constant variance
* No correlation over time
* simulating WN : arima.sim(model=c(0,0,0), n, mean, sd)
* estimating WN model : arima(x, order=c(0,0,0))
* No autocorrelation for any lags

**Random Walk** (RW) = ARIMA(0,1,0)

* + example of non-stationary process
  + no specified mean or variance
  + strong dependence over time
  + its changes or increments are white noise
    - * **Yt = Yt-1 + e** ; e = mean zero white noise
      * has on only one parameter, variance of white noise
  + Yt - Yt-1 = e , white noise with mean 0
    - * + i.e diff(Y) = WN
  + **RW with Drift**
    - Yt = C + Yt-1 + e , two parameters C and variance
    - i.e Yt - Yt-1 = C + e , WN with mean C and variance
  + the RW ACF plot is likely to show large **autocorrelation** for many lags without quick decay to zero

**The Autoregressive Model**

, ,

**(Yt - ) = \* (Yt-1 - ) + e**  ; **e is WN(0, 2)**

Three parameters : Mean - , Slope - , WN variance 2

* If phi = 0, Yt is WN with (, 2)
* If phi != 0, Yt is auto correlated
* If phi = 1 and =0, Its a Random Walk, which is not stationary
* Large values of , leads to greater autocorrelation
* Negative values of result in oscillatory time series

> **Persistence** is defined by a high correlation between an observation

and its lag, while anti-persistence is defined by a large amount of

variation between an observation and its lag

> Example and ACFs : pg 5-6

> Simulating: arima.sim(model = list(ar), n) ; -1 <= ar <= 1

> **AR Model Estimation and Forecasting**

- (Yt - ) = \* (Yt-1 - ) + e ; e is WN(0, )

arima(x, model=c(1,0,0))

ar1 =

Intercept =

2 = of WN

- Forecasting :  **= + \* ( - )**

- predict(mode, h)

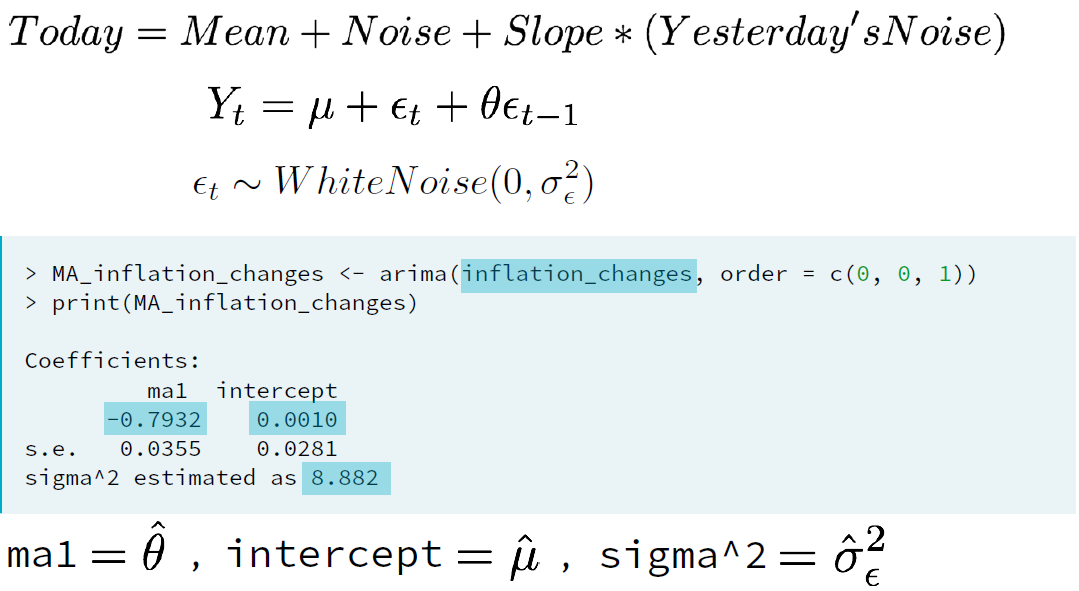
- = Yt -

- Forecast Std.Error calculated ???

- Interval = for 95% confidence

**-** Dissipating **autocorrelation** across several lags

**The Simple Moving Average**

* **Yt = + et + Ѳ et-1** 
  + Mean:
  + Slope: *Ѳ*
  + WN variance:2
* If slope = 0, then Yt is WN(2)
* If slope !=0, then Yt is auto-correlated
* Larger value of *Ѳ* leads to greater autocorrelation
* Negative value of *Ѳ* leads to oscillatory time series
* Simulation
  + arima.sim(model=list(ma= *Ѳ*), n)
* Estimation
  + 

- Forecasting :  **= + \***

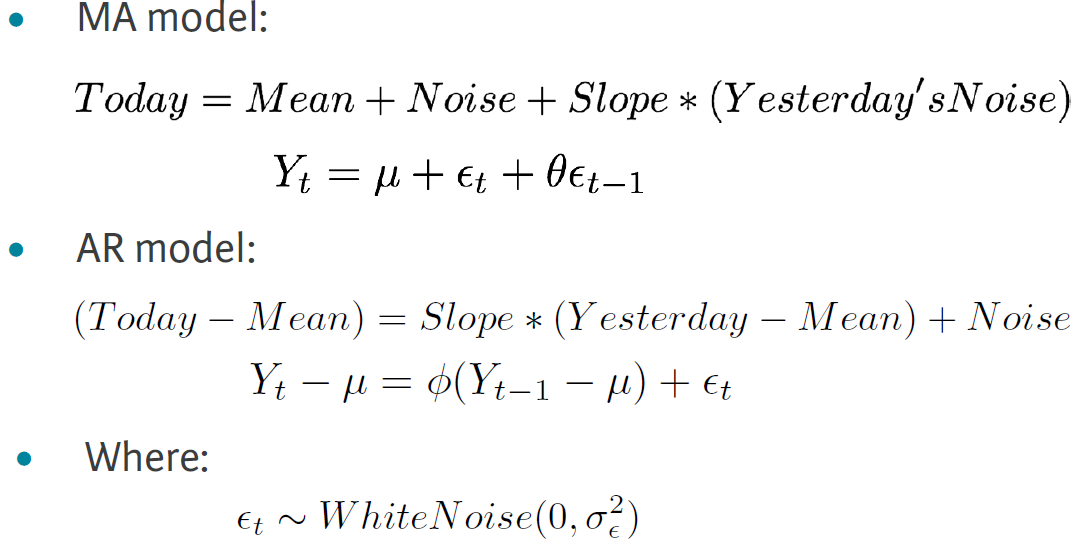
- predict(mode, h)

- = Yt -

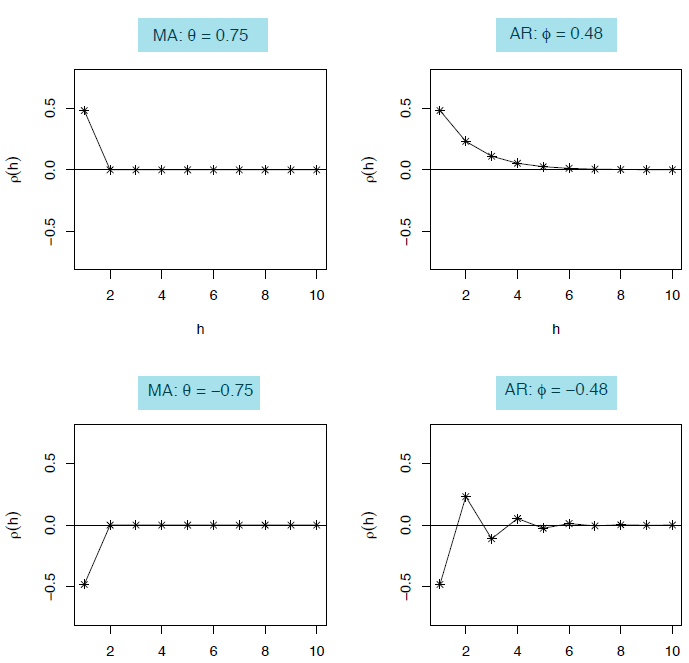
- Interval = for 95% confidence

- **Autocorrelation** for the first lag only

**MA and AR Models**



MA models have autocorrelation only at lag-1 where AR models can have it at other lags



Model fitness measured using AIC and BIC, lower the better