Lecture 9-a
Time Series:
Identification of AR, MA & ARMA Models

Brooks (4th edition): Chapter 6

Review: Times Series

- A time series y_t is a process observed in sequence over time, $t = 1, ..., T$ $\Rightarrow Y_t = \{y_1, y_2, y_3, ..., y_T\}$.

- Popular models for $E[y_t \mid I_{t-1}]$:

– AR process: $E_t[y_t \mid I_{t-1}] = f(y_{t-1}, y_{t-2}, y_{t-3}, ...\}$

Example: AR(1) process, $y_t = \mu + \phi_1\, y_{t-1} + \varepsilon_t$.

– MA process: $E_t[y_t \mid I_{t-1}] = f(\varepsilon_{t-1}, \varepsilon_{t-2}, \varepsilon_{t-3}, ...)$

Example: MA(1) process, $y_t = \mu + \theta_1\, \varepsilon_{t-1} + \varepsilon_t$

– ARMA process: $E_t[y_t \mid I_{t-1}] = f(y_{t-1}, y_{t-2}, ..., \varepsilon_{t-1}, \varepsilon_{t-2}, ...)$

Example: ARMA(1,1) process, $y_t = \mu + \phi_1\, y_{t-1} + \theta_1\, \varepsilon_{t-1} + \varepsilon_t$

Review: Times Series – Forecasting

- We want to select an appropriate time series model to forecast y_t. The linear models we consider: AR(p), MA(q) or ARMA(p, q).

- Steps for forecasting:

(1) Identify the appropriate model. That is, determine AR, MA or ARMA and the order of the model -i.e., p, q.

Tools: ACF, PACF, Information Criteria

(2) Estimate the model.

OLS, Method of Moments (complicated).

(3) Test the model.

Make sure errors are WN.

(4) Forecast.

Review: MA Process

- A linear MA(q) model:

$$y_t = \mu + \theta_1\, \varepsilon_{t-1} + \theta_2\, \varepsilon_{t-2} + \ldots + \theta_q\, \varepsilon_{t-q} + \varepsilon_t$$

- **Stationarity?** Yes ⇒ **MA(q) is always stationary**.

- **ACF:**

$$\rho(k) = \frac{\gamma(k)}{\gamma(0)} = \frac{\sum_{j=k}^{q} \theta_j \theta_{j-k}}{(1 + \theta_1^2 + \theta_2^2 + \ldots + \theta_q^2)} \qquad \text{for } |\, k \,| \leq q \quad (\theta_0 = 1)$$

$$\rho(k) = 0 \qquad \text{for } |\, k \,| > q$$

ACF as identification tool: After lag q, the autocorrelations are 0.

- **Estimation**: Complicated to estimate, we do not observe the errors, ε_t's. Direct estimation is impossible. Indirect methods:

(1) Using method of moments (MM)
(2) Using AR(∞) representation

Review: AR Process

- A linear AR(p) model:

$$y_t = \mu + \phi_1 \, y_{t-1} + \phi_2 \, y_{t-2} + \ldots + \phi_p \, y_{t-p} + \varepsilon_t, \qquad \varepsilon_t \sim WN.$$

Using the lag operator we write the AR(p) process: $\phi(L)\, y_t = \varepsilon_t$

with $\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \ldots - \phi_p L^p$

- **Stationarity?** Depends on the ϕ_j's.

We need the roots of $\phi(z) = 0$ to be **outside the unit circle**.

Example: For the AR(1) process

$$\phi(z) = 1 - \phi_1 z = 0 \;\Rightarrow |z| = \frac{1}{|\phi_1|} > 1.$$

Its corresponding ACF: $\rho(k) = \frac{\gamma(k)}{\gamma(0)} = \phi_1^k$

Review: AR(1) Process – ACF

- **ACF**: For AR(1) $\rho(k) = \frac{\gamma(k)}{\gamma(0)} = \phi_1^k$

If stationary, ACF decays with k. Patterns:

– when $0 < \phi_1 < 1$ ⇒ All autocorrelations are positive.

– when $-1 < \phi_1 < 0$ ⇒ The sign of $\rho(k)$ shows an alternating sign pattern beginning with a negative value.

ACF as identification tool: Exponential decay.

- **Estimation**: OLS. We define

$$\boldsymbol{x}_t = (1 \; y_{t-1} \; y_{t-2} \ldots . \, y_{t-p})$$
$$\boldsymbol{\beta} = (\, \phi_1 \; \phi_2 \; \ldots . \; \phi_p)$$

Then, the model can be written as

$$y_t = \boldsymbol{x}_t{}'\boldsymbol{\beta} + \varepsilon_t$$

- The **OLS** estimator is $\mathbf{b} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$

Review: ARMA Process – Stationarity & ACF

- ARMA(p, q) process: A combination of AR(p) & MA(q) processes:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q}$$

$$\Rightarrow \phi(L)\, y_t = \mu + \theta(L)\, \varepsilon_t$$

- Usually, we insist that $\phi(L) \neq 0$, $\theta(L) \neq 0$ & that the polynomials $\phi(L)$, $\theta(L)$ have no **common factors**.

- **Stationarity?** Since MA(q) processes are always stationary, the stationarity conditions come from the AR(p) part. Thus, we require the roots $\phi(L) = 0$ to be **outside the unit circle**.

- **ACF**: A **recursive formula**. After lag q, we see the exponential decay, given by the AR part. The ACF for an ARMA(1,1):

$$\rho(k) = \phi_1{}^{k-1} \frac{(1 + \phi_1\,\theta_1) * (\phi_1 + \theta_1)}{1 + \theta_1{}^2 + 2\phi_1\theta_1}$$

Review: ARMA Process – Stationarity & ACF

- **ACF**: A **recursive formula**. After lag q, we see the exponential decay, given by the AR part. The ACF for an ARMA(1,1):

$$\rho(k) = \phi_1{}^{k-1} \frac{(1 + \phi_1\,\theta_1) * (\phi_1 + \theta_1)}{1 + \theta_1{}^2 + 2\phi_1\theta_1}$$

ACF as identification tool: If stationary, exponential decay.

- **Estimation**: Complicated by MA part. In practice, use iterative OLS. Steps:

1. Estimate AR(p) part.
2. Use Step (1) to estimate (unobserved) noise ε_t
3. Regress y_t against $y_{t-1}, y_{t-2}, \ldots, y_{t-p}, \hat{\varepsilon}_{t-1}, \ldots, \hat{\varepsilon}_{t-q}$
4. Get new estimates of ε_t. Repeat Step (3).

Review: ACF – Estimation & Distribution

- **ACF Estimation**

Easy: Use sample moments to estimate $\gamma(k)$ and plug in formula:

$$r_k = \hat{\rho}_k = \frac{\sum(Y_t - \bar{Y})(Y_{t+k} - \bar{Y})}{\sum(Y_t - \bar{Y})^2}$$

- The sample **correlogram** is the plot of the ACF against k.

- **ACF Distribution**

The asymptotic distribution of the sample $r_k = \hat{\rho}_k$ is normal with:

$$\mathbf{r} \xrightarrow{d} \mathrm{N}(\boldsymbol{\rho}, \mathbf{V}/T), \qquad \mathbf{V} \text{ is the covariance matrix.}$$

Under H_0: $\rho_k = 0$ for all $k > 1$.

$$\mathbf{r} \xrightarrow{d} \mathrm{N}(\mathbf{0}, \mathbf{I}/T) \qquad \Rightarrow \mathrm{Var}[r_k] = 1/T\ .$$

- Under H_0, the $\mathrm{SE}[r_k] = 1/\sqrt{T}$ $\quad\Rightarrow$ **95% C.I.: $0 \pm 1.96 * 1/\sqrt{T}$**

Review: ACF – Identification

- The ACF can be used as a tool to select an ARMA(*p, q)* model. In general, it is used to select the lag q in an MA model.

	AR(p)	MA(q)	ARMA(p, q)
ACF	Tails off	**0 after lag q**	Tails off

Note: Ideally, “Tails off” is exponential decay. In practice, we may see decay with a lot of “noise” and a lot of non-zero values.

- In the next slides, we simulate ARMA models. This is an “ideal” situation, we know the model that generated the data, we know what the ACF should look like in theory. Then, we look at the empirical ACF to see if it is easy to guess the model and order of the model.

Review: ACF – AR(1)

Simulated Example: Sample ACF for an AR(1) process:
Under stationarity:

$$\rho(k) = \frac{\gamma(k)}{\gamma(0)} = \phi_1^k \qquad k = 0, 1, 2, \ldots$$

If $|\phi_1| < 1$, the ACF will show exponential decay.

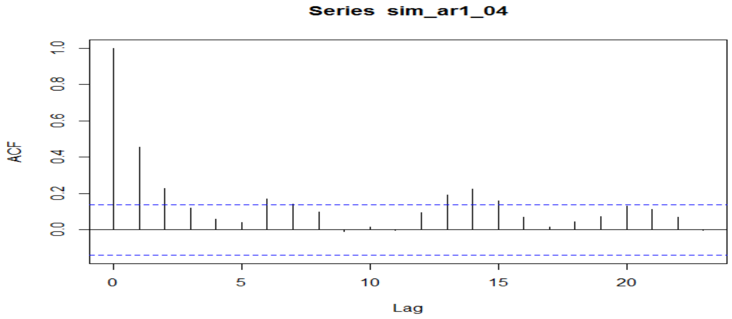
- Suppose $\phi_1 = 0.4$. Then, the theoretical ACF:

$\rho(0) = \mathbf{1}$
$\rho(1) = \mathbf{0.4}$
$\rho(2) = 0.4^2 = \mathbf{0.16}$
$\rho(3) = 0.4^3 = \mathbf{0.064}$
$\rho(4) = 0.4^4 = \mathbf{0.0256}$
$\vdots$
$\rho(k) = \mathbf{0.4}^k$

Review: ACF – AR(1)

Simulated Example (continuation): Plot of simulated series and ACF, with 95% C.I. = [**-0.1386, 0.1386**].

```
> sim_ar1_04 <- arima.sim(list(order=c(1,0,0), ar=0.4), n=200)          # sim AR(1)
```



Review: ACF – MA(*q* = 3)

Simulated Example:

$$y_t = \mu + \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \theta_3\varepsilon_{t-3}$$

Suppose θ_1 = **0.5**; θ_2 = **0.4**; θ_3 = **0.2**. Then, the theoretical ACF:

$$\rho(0) = 1$$

$$\rho(1) = \frac{\theta_1 + \theta_2\theta_1 + \theta_3\theta_2}{(1+{\theta_1}^2 + {\theta_2}^2 + {\theta_3}^2)} = \frac{0.5+0.4*0.5+0.1*0.4}{1 + 0.5^2 + 0.4^2 + 0.1^2} = \mathbf{0.5211}$$

$$\rho(2) = \frac{\theta_2 + \theta_3\theta_1}{(1+{\theta_1}^2 + {\theta_2}^2 + {\theta_3}^2)} = \frac{0.4 + 0.1*0.5}{1 + 0.5^2 + 0.4^2 + 0.1^2} = \mathbf{0.3169}$$

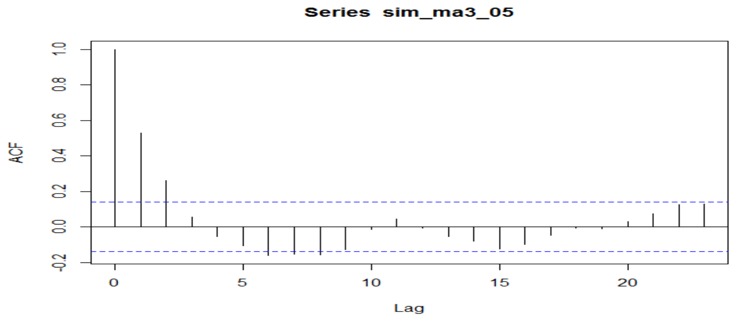
$$\rho(3) = \frac{\theta_3}{(1+{\theta_1}^2 + {\theta_2}^2 + {\theta_3}^2)} = \frac{0.1}{1 + 0.5^2 + 0.4^2 + 0.1^2} = \mathbf{0.0704}$$

$$\rho(k) = \mathbf{0} \qquad \text{for } |k| > 3.$$

Review: ACF – MA(*q* = 3)

Simulated Example (continuation): Plot of simulated series and ACF with 95% C.I. = [**-0.1386, 0.1386**].

```
> sim_ma3_05 <- arima.sim(list(order=c(0,0,3), ma=c(0.5, 0.4, 0.2)), n=200)      # sim MA(3)
```



Review: ACF – ARMA(1, 1)

Simulated Example: Sample ACF for an ARMA(1,1) process:

$$y_t = \phi_1 y_{t-1} + \varepsilon_t + \theta_1 \varepsilon_{t-1}$$

The ACF for an ARMA(1,1):

$$\rho(k) = \phi_1^{\,k-1} \frac{(1+\phi_1\theta_1)*(\phi_1+\theta_1)}{1+\theta_1^2+2\phi_1\theta_1}$$

Suppose $\phi_1 = \mathbf{0.4}$, $\theta_1 = \mathbf{0.5}$. Then, the theoretical ACF:

$$\rho(0) = \mathbf{1}$$

$$\rho(1) = \frac{(1+0.4*0.5)*(0.4+0.5)}{1+0.5^2+2*0.4*0.5} = \mathbf{0.6545}$$

$$\rho(2) = 0.4 * \frac{(1+0.4*0.5)*(0.4+0.5)}{1+0.5^2+2*0.4*0.5} = \mathbf{0.2618}$$

$$\rho(3) = 0.4^2 * \frac{(1+0.4*0.5)*(0.4+0.5)}{1+0.5^2+2*0.4*0.5} = \mathbf{0.0233}$$

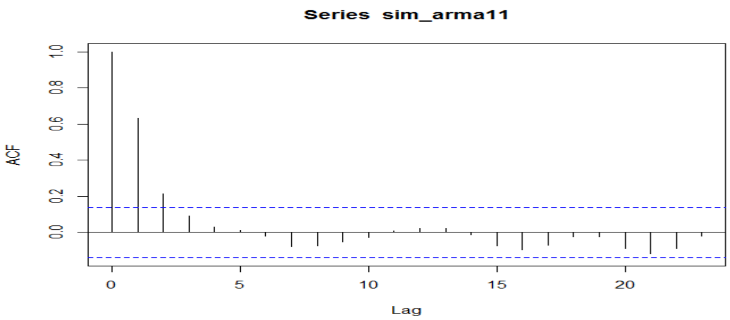
⋮

$$\rho(k) = 0.4^{k-1} * \frac{(1+0.4*0.5)*(0.4+0.5)}{1+0.5^2+2*0.4*0.5}$$

Review: ACF – ARMA(1,1)

Simulated Example (continuation): Plot of simulated series and ACF with 95% C.I. = [**-0.1386, 0.1386**].

```
> sim_arma11 <- arima.sim(list(order=c(1,0,1), ar=0.4, ma=0.5), n=200)    #sim ARMA(1,1)
```



Review: ACF – Example: U.S. Stock Returns

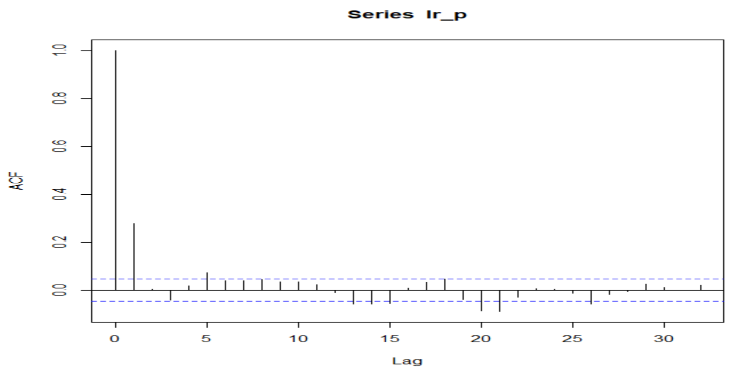
Example: **US Monthly Returns** (1871 – 2020, T = 1,795)

```
Sh_da <- read.csv("https://www.bauer.uh.edu/rsusmel/4397/Shiller_2020data.csv",
head=TRUE, sep=",")
x_P <- Sh_da$P
x_D <- Sh_da$D
T <- length(x_P)
lr_p <- log(x_P[-1]/x_P[-T])
lr_d <- log(x_D[-1]/x_D[-T])
acf_p <- acf(lr_p)                                   # acf: R function that estimates the ACF
> acf_p
Autocorrelations of series ‘lr_p’, by lag

     0      1      2      3      4      5      6      7      8      9     10     11
 1.000  0.279  0.004 -0.043  0.017  0.074  0.039  0.039  0.044  0.035  0.034  0.022
    12     13     14     15     16     17     18     19     20     21     22     23
-0.010 -0.059 -0.058 -0.056  0.009  0.033  0.047 -0.040 -0.087 -0.090 -0.029  0.005
    24     25     26     27     28     29     30     31     32
 0.003 -0.013 -0.058 -0.018 -0.005  0.026  0.011  0.000  0.020
```

SE[r_k] = $1/\sqrt{T}$ = $1/\sqrt{1,795}$ = .0236. ⇒ 95% C.I.: ± 2* 0.0236

Review: ACF – Example: U.S. Stock Returns

Example (continuation): Correlogram for **US Monthly Returns** (1871 – 2020), with 95% CI = [**-0.0472, 0.0472**].



Note: With the exception of first correlation, correlations are small. However, many are significant, not strange result when T is large.

Review: ACF – Example: U.S. Stock Dividends

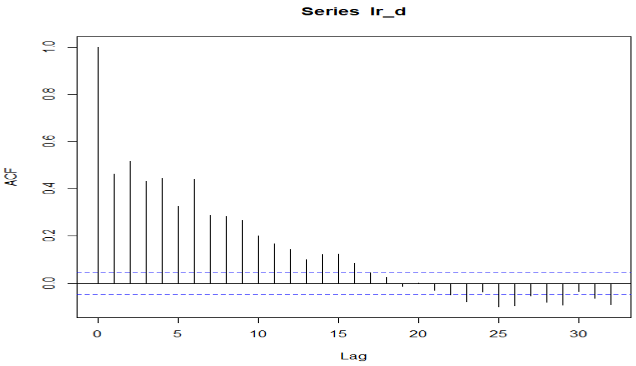
Example: **US Monthly Changes in Dividends** (1871 – 2020, *T* = **1,795**)

```
acf_d <- acf(lr_d)
> acf_d
Autocorrelations of series ‘lr_d’, by lag

    0      1      2      3      4      5      6      7      8      9     10     11
 1.000  0.462  0.516  0.432  0.444  0.326  0.442  0.288  0.283  0.265  0.202  0.168
   12     13     14     15     16     17     18     19     20     21     22     23
 0.142  0.100  0.122  0.123  0.085  0.045  0.026 -0.013  0.001 -0.029 -0.049 -0.077
   24     25     26     27     28     29     30     31     32
-0.038 -0.100 -0.095 -0.055 -0.081 -0.092 -0.034 -0.063 -0.089
```

High correlations and significant even after 32 months!

Review: ACF – Example: U.S. Stock Dividends

Example (continuation): Correlogram for **US Monthly Changes in Dividends** (1871 – 2020, *T*=**1,795**), with 95% CI = [**-0.0472, 0.0472**].



Note: Correlations are positive for almost 1.5 years, then become negative.

Review: ACF – Joint Significance Tests

- Recall we compute the Ljung-Box (LB) statistic as:

$$LB = T * (T+2)\ \sum_{k=1}^{m}\left(\frac{\hat{\rho}_k^2}{(T-k)}\right)$$

The LB test can be used to determine if the first m sample ACFs are jointly equal to zero.

Under H$_0$: $\rho_1 = \rho_2 = \ldots = \rho_m = 0$, $LB \xrightarrow{d} \chi^2_m$

- Usually, we use LB tests to make sure the chosen ARMA model does not have any correlation structure on the residuals –i.e., they look WN.

Review: ACF – Joint Significance Tests

Example: LB test with **20 lags** for **US Monthly Returns** and **Changes in Dividends** (1871 – 2020, T = 1,795)

```
> Box.test(lr_p, lag=20, type= "Ljung-Box")

	Box-Ljung test

data:  lr_p
X-squared = 208.02, df = 20, p-value < 2.2e-16
```

⇒ Reject H$_0$ at 5% level. Joint significant first 20 correlations.

```
> Box.test(lr_d, lag=20, type= "Ljung-Box")

	Box-Ljung test

data:  lr_d
X-squared = 2762.7, df = 20, p-value < 2.2e-16
```

⇒ Reject H$_0$ at 5% level. Joint significant first 20 correlations.

Review: Partial ACF (PACF)

- The ACF gives us a lot of information about the order of the dependence when the series we analyze follows a MA process: The ACF is zero after q lags for an MA(q) process.

- We introduce a new function that behaves like the ACF of MA models, but for AR models: The **partial autocorrelation function** (**PACF**).

- The PACF is similar to the ACF. It measures correlation between observations that are k time periods apart, after controlling for correlations at intermediate lags.

Review: Partial ACF

Intuition: Suppose we have an AR(1):

$$y_t = \phi_1 y_{t-1} + \varepsilon_t.$$

Then,

$$\gamma(2) = \phi_1^2 \gamma(0)$$

The correlation between y_t and y_{t-2} is not zero, as it would be for an MA(1), because y_t is dependent on y_{t-2} through y_{t-1}.

Suppose we break this chain of dependence by removing ("partialing out") the effect y_{t-1}. Then, we consider the correlation between $[y_t - \phi_1 y_{t-1}]$ & $[y_{t-2} - \phi_1 y_{t-1}]$ –i.e, the correlation between y_t & y_{t-2} with the linear dependence of each on y_{t-1} removed:

$$\gamma(2) = \text{Cov}(y_t - \phi_1 y_{t-1}, y_{t-2} - \phi_1 y_{t-1}) = \text{Cov}(\varepsilon_t, y_{t-2} - \phi_1 y_{t-1}) = 0$$

Similarly,

$$\gamma(k) = \text{Cov}(\varepsilon_t, y_{t-k} - \phi_1 y_{t-1}) = 0 \quad \text{for all } k > 1.$$

Review: Partial ACF

- The PACF ϕ_{hh} is also the last coefficient in the **best linear prediction** of y_t given $y_{t-1}, y_{t-2}, ..., y_{t-h}$. (⇒ OLS!)

OLS estimation steps:

Regress y_t against y_{t-1}

$\Rightarrow \phi_{11}$: estimated coefficient of y_{t-1}.

Regress y_t against y_{t-1} & y_{t-2}

$\Rightarrow \phi_{22}$: estimated coefficient of y_{t-2}.

⋮

Regress y_t against $y_{t-1}, y_{t-2}, \dots y_{t-h}$

$\Rightarrow \phi_{hh}$: estimated coefficient of y_{t-h}.

- OLS estimation is simple, easy to use. Estimation by Yule-Walker equation (Method of Moments) is possible.

- The plot of the PACF is called the **partial correlogram**.

Partial ACF – AR(p = 2)

Simulated Example: We simulate an AR(2) process:

$$y_t = \mu + \phi_1 \, y_{t-1} + \phi_2 \, y_{t-2} + \varepsilon_t$$

```
sim_ar22 <- arima.sim(list(order=c(1,0,0), ar=c(0.5, 0.3)), n=200)        #simulate AR(2) series
plot(sim_ar22, ylab="Simulated Series", main=(expression(AR(2):~~~phi==c(0.5,0.3))))
pacf_ar22 <- pacf(sim_ar22)
```

Print PACF

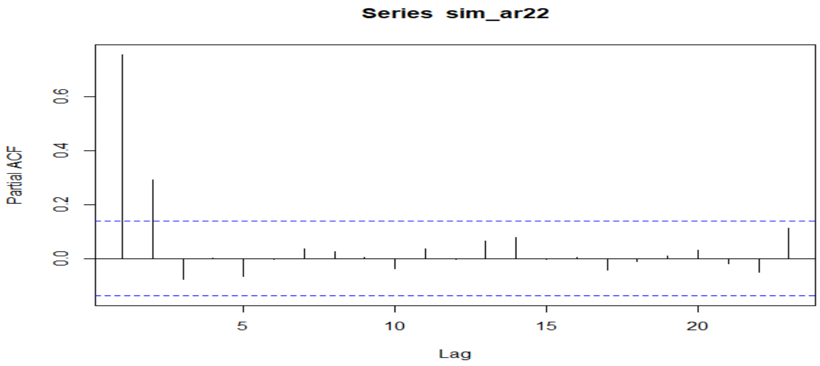
```
> pacf_ar2

     1      2      3      4      5      6      7      8      9     10     11
 0.558  0.286  0.038  0.103 -0.010  0.009  0.111  0.060 -0.021 -0.076  0.016
    12     13     14     15     16     17     18     19     20     21     22     23
-0.086 -0.139  0.100  0.061 -0.156  0.078 -0.103  0.043 -0.075  0.104  0.024  0.061
```

SE$[PACF_k] \approx 1/\sqrt{200}$ = .0707.   ⇒ 95% C.I.: 0 ± 2 * **0.0707**

Partial ACF – AR(p = 2)

Simulated Example (continuation): Plot of simulated series & PACF with 95% C.I. = [**-0.1386, 0.1386**].

```
> plot(sim_ar22, ylab="Simulated Series", main=(expression(AR(2):~~~phi==c(0.5,0.3))))
> pacf_ar2 <- pacf(sim_ar22)
```



Partial ACF – AR(p = 2)

Simulated Example (continuation):
Note: The PACF can be calculated by h regressions, each one with h lags. The hh coefficient is the h^{th} order PACF. Using *ar* function:

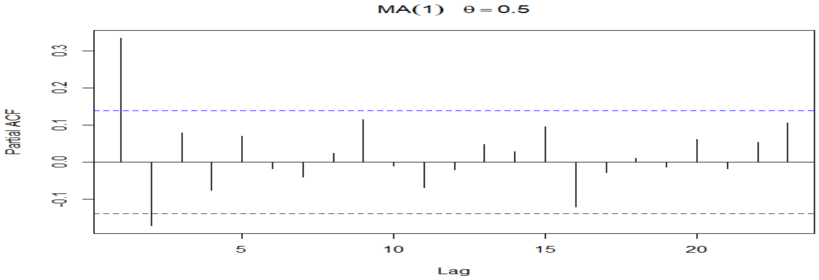
```
> ar(sim_ar2, order.max=1, method = “ols")

Coefficients:
     1
0.5586

Intercept: -0.008403 (0.0761)

Order selected 1  sigma^2 estimated as  1.152


> ar(sim_ar2, order.max=2, method = "ols")

Coefficients:
     1       2
0.3974  0.2869

Intercept: -0.009847 (0.07326)

Order selected 2  sigma^2 estimated as  1.063
```

Partial ACF – MA(q)

- Following the analogy that PACF for AR processes behaves like an ACF for MA processes, we expect exponential decay ("*tails off*") in the partial correlogram for MA process.

Simulated Example: We simulate an MA(1) process with θ_1 = 0.5.

```
sim_ma1 <- arima.sim(list(order=c(0,0,1), ma = 0.5), n=200)
> pacf(sim_ma1)
```

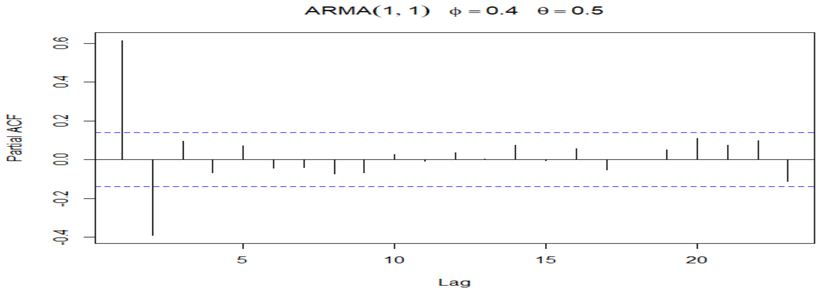


Partial ACF – ARMA(p, q)

- For an ARMA processes, after lag p, the MA part will dominate the behavior of the PACF, thus we expect exponential decay ("*tails off*") in the partial correlogram.

Simulated Example: We simulate an ARMA(1) process with ϕ_1= 0.4 & θ_1= 0.5.

```
sim_arma11 <- arima.sim(list(order=c(1,0,1), ar=0.4, ma=0.5), n=200)
> pacf(sim_arma11)
```



PACF – Example: U.S. Stock Returns

Example: **US Monthly Returns** (1871 – 2020, T = 1,795)

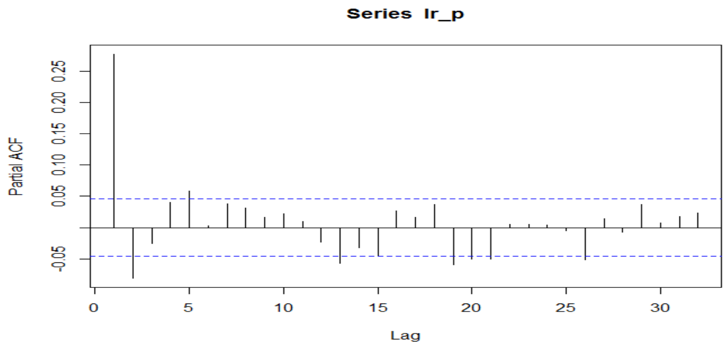
```
pacf_p <- acf(lr_p)                              # pacf: R function that estimates the PACF
> pacf_p

Partial autocorrelations of series ‘lr_p’, by lag

    1      2      3      4      5      6      7      8      9      10     11
 0.278 -0.081 -0.026  0.041  0.058  0.002  0.038  0.032  0.016  0.022  0.009
    12     13     14     15     16     17     18     19     20     21     22     23
-0.023 -0.057 -0.032 -0.045  0.027  0.017  0.037 -0.059 -0.051 -0.050  0.005   24
    23     24     25     26     27     28     29     30     31     32
 0.006  0.004 -0.005 -0.051  0.014 -0.007  0.037  0.008  0.018  0.023
```

SE$[PACF_k] \approx 1/\sqrt{1,795}$ = .0236. $\Rightarrow$ 95% C.I.: 0 $\pm$ 2 * **0.0236**

PACF – Example: U.S. Stock Returns

Example (continuation): Correlogram for **US Monthly Returns** (1871 – 2020, T = 1,795) with 95% C.I. = [**-0.0472, 0.0472**].

```
> pacf(lr_p)
```



Note: With the exception of the first partial correlation, partial correlations are small, though, again, some are significant.

PACF – Example: U.S. Stock Dividends

Example: **US Monthly Stock Dividends** (1871 – 2020, *T*=1,795)

```
pacf_d <- pacf(lr_d)
> pacf_d

Partial autocorrelations of series ‘lr_d’, by lag

    1      2      3      4      5      6      7      8      9     10     11
 0.462  0.385  0.160  0.150 -0.033  0.189 -0.054 -0.056  0.027 -0.082 -0.019
   12     13     14     15     16     17     18     19     20     21     22     23
-0.063 -0.035  0.067  0.043  0.010 -0.057 -0.046 -0.043 -0.008 -0.031 -0.039
    24     25     26     27     28     29     30     31     32
-0.041  0.050 -0.036 -0.030  0.091  0.006 -0.017  0.044 -0.002 -0.042
```

Higher partial correlations than for stock returns.

ARIMA Models: Identification – Correlations

- Correlation approach.

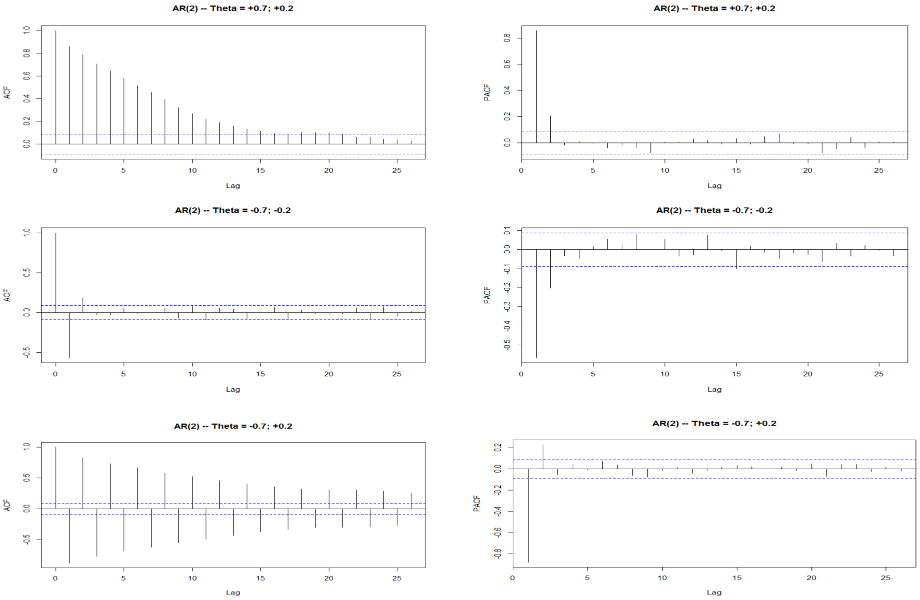
Basic tools: sample ACF and sample PACF.
  - ACF identifies order of MA: Non-zero at lag q; zero for lags > q.
  - PACF identifies order of AR: Non-zero at lag p; zero for lags > p.
  - All other cases, try ARMA(p, q) with $p > 0$ and $q > 0$.

<u>Summary</u>: For $p > 0$ & $q > 0$.

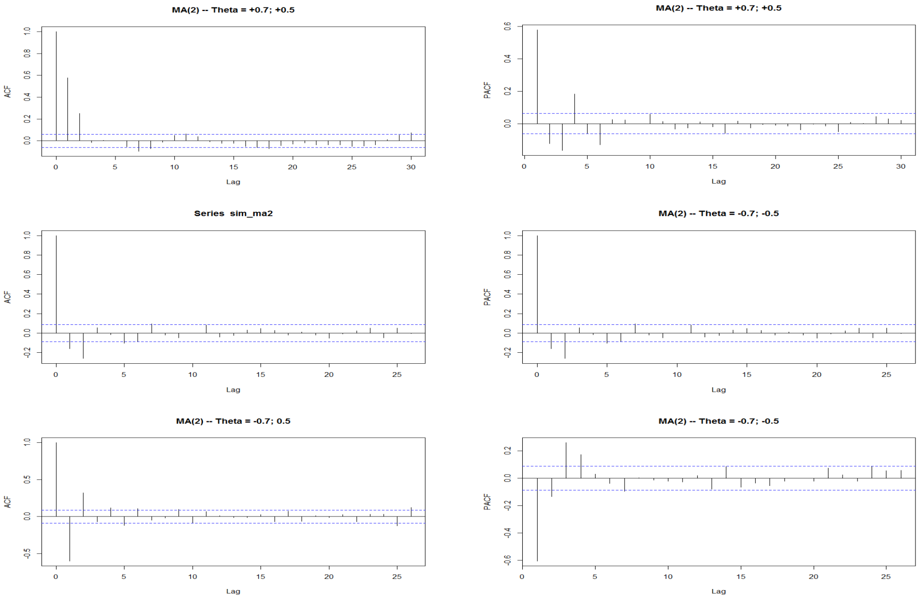
	AR(p)	MA(q)	ARMA(p, q)
ACF	Tails off	**0 after lag q**	Tails off
PACF	**0 after lag p**	Tails off	Tails off

<u>Note</u>: Ideally, "Tails off" is exponential decay. In practice, in these cases, we may see a lot of non-zero values for the ACF and PACF.

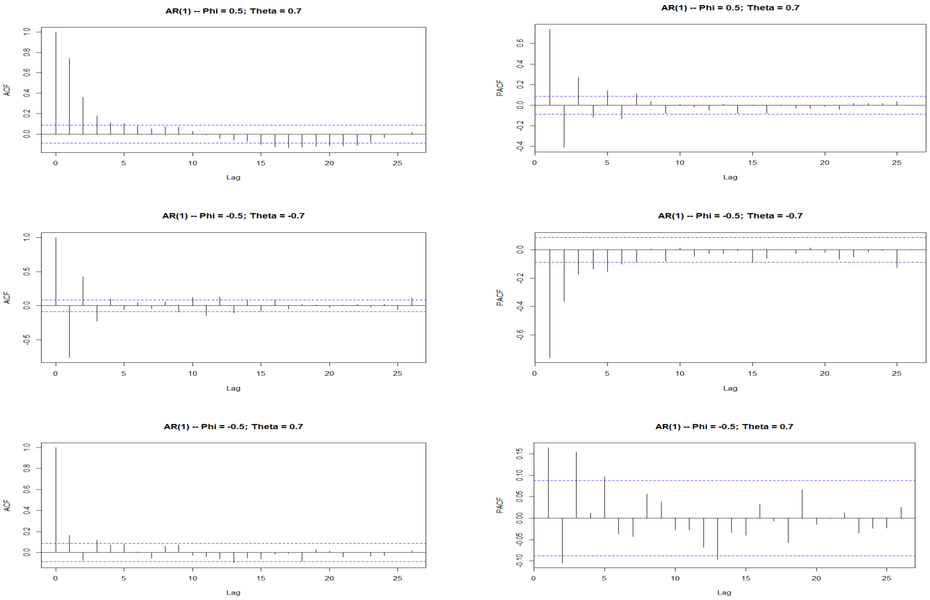
ARMA Models: Identification – AR(2)



ARMA Models: Identification – MA(2)

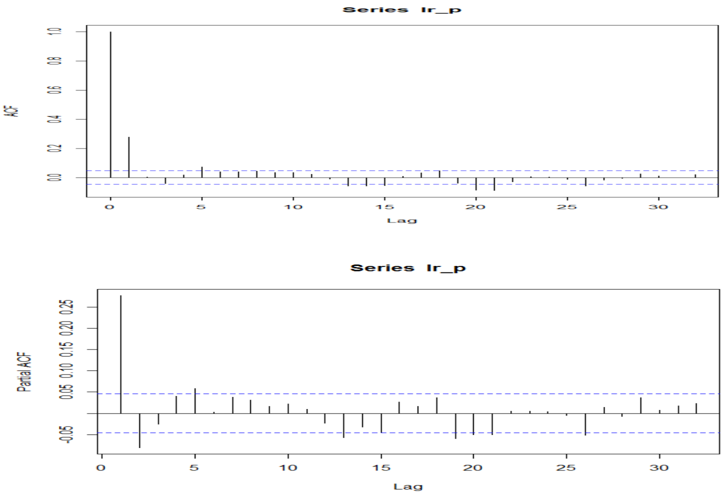


ARMA Models: Identification – ARMA(1, 1)



ARMA Models: Identification – ARMA(p, q)?

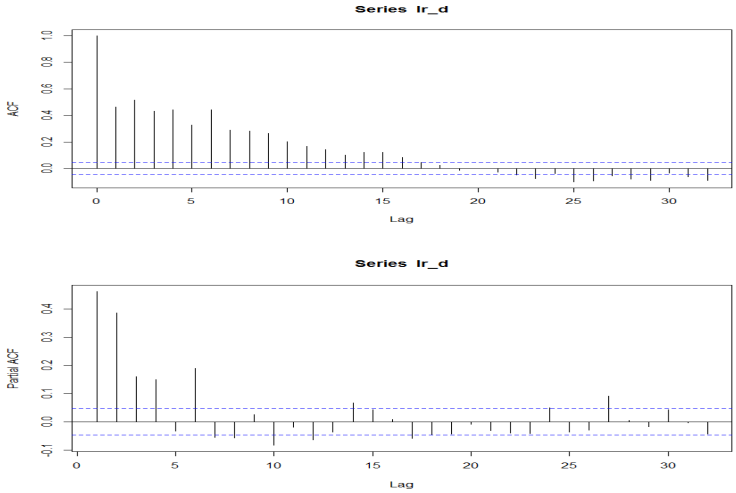
Example: **Monthly US Returns** (1871 - 2020).



- Note: ARMA(1,1), MA(1), AR(2)?

ARMA Models: Identification – ARMA(p, q)?

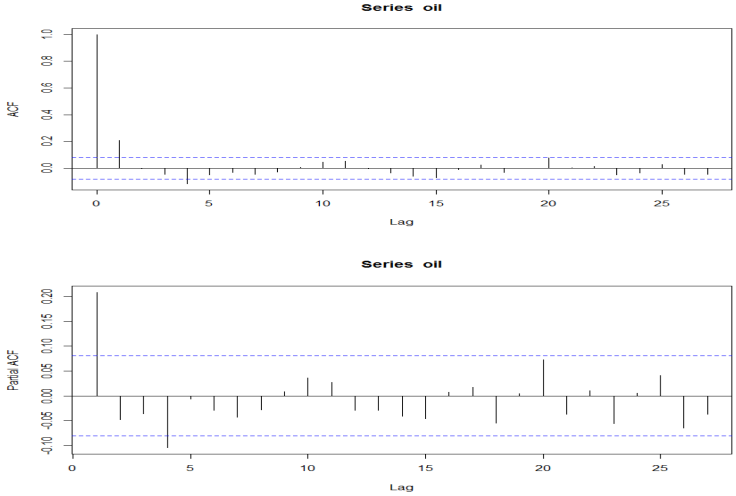
Example: **Monthly Changes in US Dividends** (1871 - 2020).



- Note: Not clear: Maybe long a ARMA(p, q) or needs differencing?

ARMA Models: Identification – ARMA(p, q)?

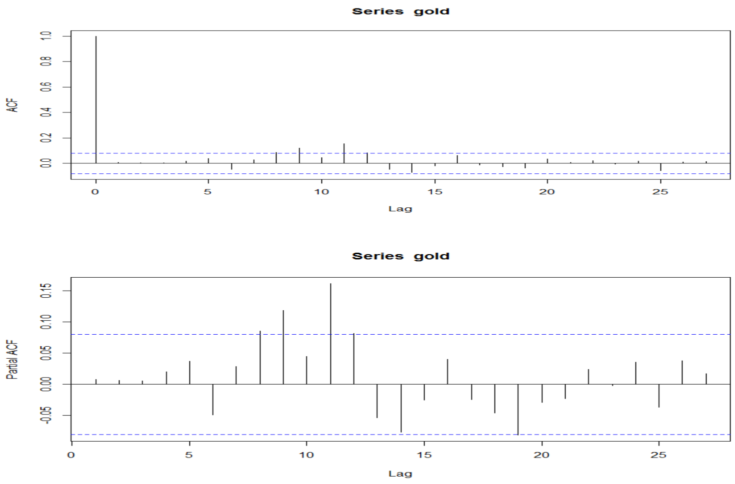
Example: **Monthly Log Changes in Oil Prices** (1973 - 2020).



- Note: MA(1), AR(4)?

ARMA Models: Identification – ARMA(p, q)?

Example: **Monthly Log Changes in Gold** (1973 - 2020).



- Note: No clear ARMA structure.

ARMA Model: Identification - IC

- It is difficult to identify an ARMA model using the ACF and PACF. It is common to rely on information criteria (IC).

- IC's are equal to the estimated variance or the log-likelihood function plus a penalty factor, that depends on k. Many IC's:

- **Akaike Information Criterion (AIC)**

$\text{AIC} = -2 * (\ln L - k) = -2 \ln L + 2 * k$

$\Rightarrow$ if normality $\text{AIC} = T * \ln(\boldsymbol{e}'\boldsymbol{e}/T) + 2 * k$ (+constants)

- **Bayes-Schwarz Information Criterion (BIC or SBIC)**

$\text{BIC} = -2 * \ln L - \ln(T) * k$

$\Rightarrow$ if normality $\text{AIC} = T * \ln(\boldsymbol{e}'\boldsymbol{e}/T) + \ln(T) * k$ (+constants)

- **Hannan-Quinn (HQIC)**

$\text{HQIC} = -2*(\ln L - k\ [\ln(\ln(T))]$

$\Rightarrow$ if normality $\text{AIC} = T * \ln(\boldsymbol{e}'\boldsymbol{e}/T) + 2\, k\ [\ln(\ln(T))]$ (+constants)

ARIMA Model: Identification - IC

- There are modifications of IC to get better finite sample behavior, a popular one is AIC corrected, AICc, statistic:

$$AICc = T\, ln\hat{\sigma}^2 + \frac{2k(k+1)}{T-k-1}$$

- AICc converges to AIC as *T* gets large. Using AICc is not a bad idea.

- For AR(p) models, other AR-specific criteria are possible: Akaike's final prediction error (FPE), Akaike's BIC, Parzen's CAT.

- Hannan and Rissannen's (1982) **minic** *(=Min*imum *IC)*: Calculate the BIC for different p*'s* (estimated first) and different q*'s*. Select the best model –i.e., lowest BIC.

Note: Box, Jenkins, and Reinsel (1994) proposed using the AIC above.

ARMA Model: Identification - IC

Example: Monthly US Returns (1871 - 2020) Hannan and Rissannen (1982)'s minic, based on AIC.

Minimum Information Criterion

Lags	**MA 0**	**MA 1**	**MA 2**	**MA 3**	**MA 4**	**MA 5**
AR 0	-6403.59	-6552.94	-6552.69	-6554.27	-6552.88	-6557.37
AR 1	-6545.22	-6552.23	-6551.86	-6552.42	-6552.64	**-6561.48**
AR 2	-6554.76	-6553.28	-6554.85	-6554.35	**-6564.32**	-6559.48
AR 3	-6553.94	-6552.53	-6554.44	-6552.33	-6550.36	-6558.52
AR 4	-6554.98	-6559.83	**-6559.92**	-6558.94	-6554.1	-6558.16
AR 5	**-6558.81**	-6558.65	-6557.45	-6555.78	-6558.66	-6556.06

- Note: Best Model is ARMA(2,4); other potential candidates: ARMA(1,5); ARMA(5,0), ARMA(4,2).

Non-Stationary Time Series Models

- A trend is usually easy to spot. A more sophisticated visual tool is the ACF: a slow decay in ACF is indicative of highly correlated data, which suggests a trend.

- A series with a trend is not stationary. To build a forecasting model, we need to remove the trend from the series. The models we consider:

(1) Deterministic trend: y_t is a function of t. For example,

$$y_t = \alpha + \beta\, t + \varepsilon_t$$

(2) Stochastic trend: y_t is a function of aggregated errors, ε_t, over time. For example,

$$y_t = \mu + y_{t-1} + \varepsilon_t = y_0 + t\,\mu + \sum_{j=0}^{t} \varepsilon_{t-j}$$

- The process to remove the trend depends on the structure of the DGP of y_t.

Non-Stationary Time Series Models

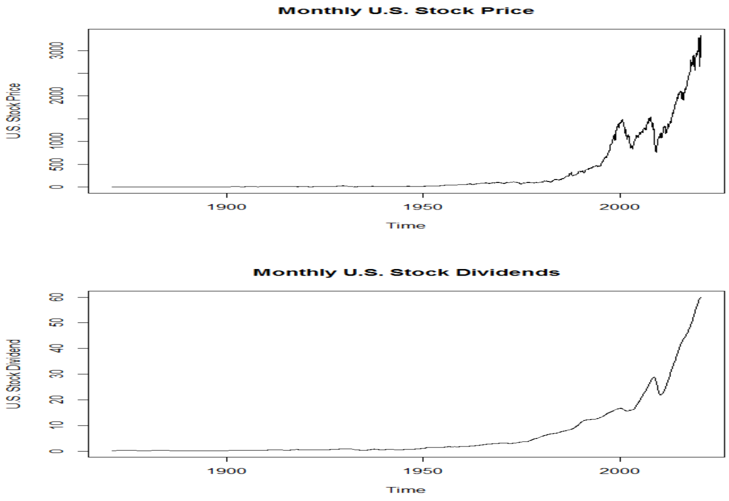
- The process to remove the trend depends on the nature of the DGP of the trending y_t:

(1) Deterministic trend – Simple model: $y_t = \alpha + \beta\, t + \varepsilon_t$
– Solution: **Detrending** –i.e., regress y_t on a constant and a time trend, t. Then, keep residuals for further modeling.

(2) Stochastic trend – Simple model: $y_t = \mu + y_{t-1} + \varepsilon_t$.
– Solution: **Differencing** –i.e., apply $\Delta = (1 - L)$ operator to y_t. Then, use Δy_t for further modeling.

Non-Stationary Time Series Models

Example: Plot of US Monthly Prices and Dividends (1871 – 2020)



Non-Stationary Models: Deterministic Trend

- Suppose we have the following model, with a determinist trend:

$$y_t = \alpha + \beta\, t + \varepsilon_t.$$

- $\{y_t\}$ will show only temporary departures from trend line $\alpha + \beta\, t$. This type of model is called a **trend stationary** (TS) model.

- Note that trivially, by definition, ε_t is WN. Then, removing $\alpha + \beta\, t$ from y_t creates a WN series –i.e., the influence of t from $y_t$is gone:

$$\varepsilon_t = y_t - \alpha - \beta\, t$$

- When we replace α & β by their OLS estimates, we **detrend** y_t. The residual from the OLS is called **detrended** y_t.

$$e_t = y_t - \hat{\alpha} - \hat{\beta}\, t$$ (the residuals are the *detrended* y_t series)

Non-Stationary Models: Deterministic Trend

- We can detrend in more complicated models. For example, suppose e have a stationary AR(p) model with linear and quadratic trends:

$$y_t = \alpha + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \beta_1 t + \beta_2 t^2 + \varepsilon_t.$$

- Note that removing from y_t a constant, a linear and a quadratic trend creates a series, w_t, which is composed of a WN error, ε_t, and the AR(p) part:

$$w_t = \varepsilon_t + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} = y_t - \alpha - \beta_1 t - \beta_2 t^2$$

- This is a stationary series: the dependence on t is gone. We will work with the residual from a regression of y_t agains a constant, t and t^2:

$$\hat{w}_t = y_t - \hat{\alpha} - \hat{\beta}_1 t - \hat{\beta}_2 t^2 \qquad (\hat{w}_t = \textbf{detrended } y_t).$$

Remark: We do not necessarily get stationary series by detrending.

Non-Stationary Models: Deterministic Trend

- Many economic series exhibit "exponential trend/growth". They grow over time like an exponential function over time instead of a linear function. In this cases, it is common to work with logs

$$\ln(y_t) = \alpha + \beta\, t + \varepsilon_t. \qquad (\Rightarrow y_t = e^{\alpha + \beta\, t + \varepsilon_t})$$

⇒ The average growth rate is: $E[\Delta \ln(y_t)] = \beta$

- We can have a more general model:

$$y_t = \alpha + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \beta_1 t + \beta_2 t^2 + \ldots + \beta_k t^k + \varepsilon_t.$$

- Estimation of AR(p) with a trend component:

- OLS.
- Frish-Waugh method (a 2-step method):
  (1) Detrend y_t: regress y_t against a constant & a time trend, t.
  Then, get the residuals (=y_t without the influence of t).
  (2) Use residuals to estimate the AR(p) model.

Non-Stationary Models: Deterministic Trend

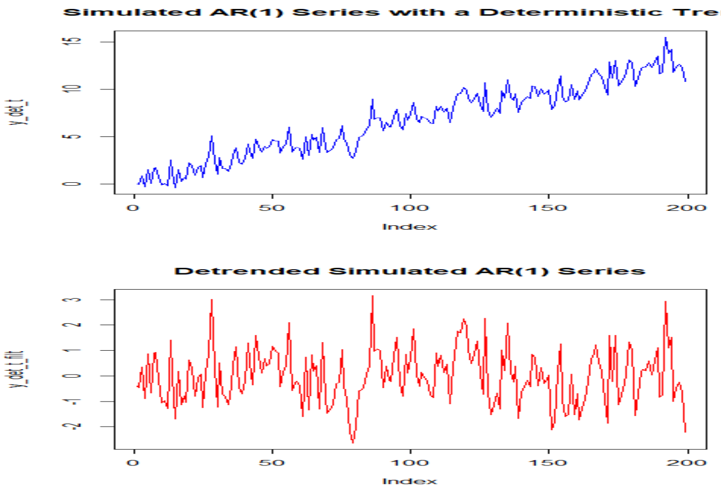
Simulated Example: We simulate an AR(1) series with a trend:

$$y_t = 0.3 + 0.2\, y_{t-1} + 0.05\, t + \varepsilon_t.$$

```
T_sim <- 200                                      # Length of simulation
y_sim <- matrix(0,T_sim,1)                        # Vector to accumulate simulated data
u <- rnorm(T_sim, sd = 1)                         # Draw T_sim normally distributed errors
mu <- 0.3                                         # Constant
phi1 <- 0.2                                       # Change to create different AR(1) patterns
mu_t <- .05                                       # Trend coefficient
t <- 2                                            # Time index for observations
while (t <= T_sim) {
 y_sim[t] = mu + phi1 * y_sim[t-1] +  mu_t * t + u[t]  # y_sim simulated values
  t <- t + 1
}
y_det_t <- y_sim[2: T_sim]
plot(y_det_t, type="l", col = "blue", main = "Simulated Series with a Deterministic Trend")

# Detrend series
trend <- c(1:(T_sim1-1))
fit_det_t <- lm(y_det_t ~ trend)
y_det_t_filt <- fit_det_t$residuals                          # Filtered series
plot(y_det_t_filt, type="l", main = "Detrended Simulated Series")
```

Non-Stationary Models: Deterministic Trend

Simulated Example (continuation): We plot the simulated AR(1) series (blue) and the detrended simulated series (red).



Non-Stationary Models: Deterministic Trend

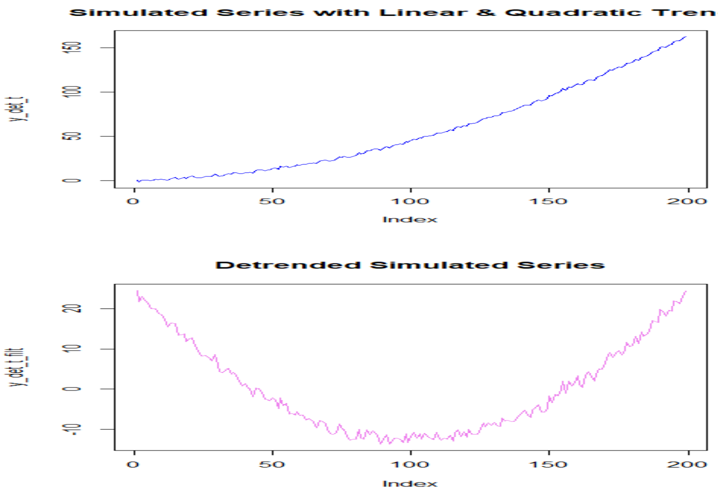
Simulated Example (continuation): Now, we add a quadratic trend:

$$y_t = 0.3 + 0.2\, y_{t-1} + 0.05\, t + 0.003\, t^2 + \varepsilon_t.$$

```
mu_t2 <- .003                                          # Trend square coefficient
t <- 2                                                 # Time index for observations
while (t <= T_sim) {
y_sim[t] = mu + phi1 * y_sim[t-1] +  mu_t * t + u[t]   # y_sim simulated autocorrelated values
t <- t + 1
}
y_det_t <- y_sim[2: T_sim]
plot(y_det_t, type="l", col = "blue", main = "Simulated Series with a Deterministic Trend")

# Detrend series with only a linear trend
trend <- c(1:(T_sim1-1))
fit_det_t <- lm(y_det_t ~ trend)
y_det_t_filt <- fit_det_t$residuals                    # Filtered series
plot(y_det_t_filt, type="l", main = "Detrended Simulated Series")

## Detrend series with a linear & Quadratic trends
trend2 <- trend^2fit_det_t <- lm(y_det_t ~ trend + trend2)
y_det_t_filt <- fit_det_t$residuals                    # Filtered series
plot(y_det_t_filt, type="l", col = "violet", main="Detrended Simulated Series")
```

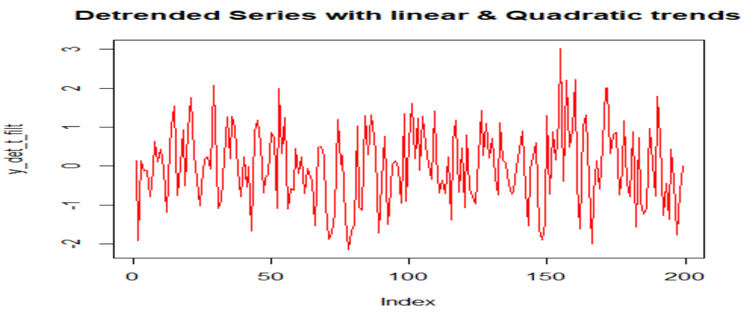
Non-Stationary Models: Deterministic Trend

Simulated Example (continuation): We plot the simulated AR(1) series (blue) and the detrended series with a linear trend (violet).



Non-Stationary Models: Deterministic Trend

Simulated Example (continuation): We plot the detrended simulated series with a linear and quadratic trends (red).

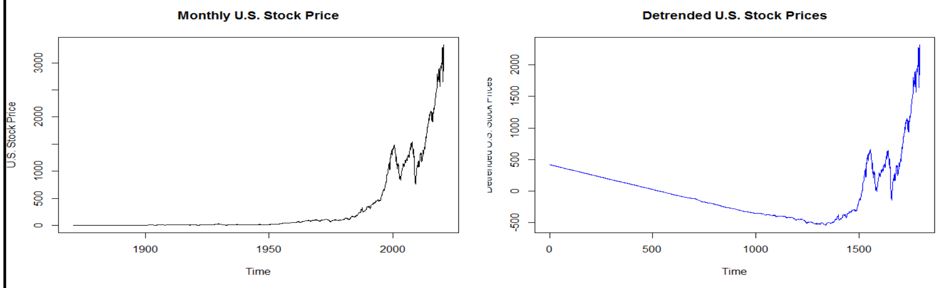


Remark: A series with a quadratic trend, needs to be detrended with a quadratic trend, otherwise extra patterns (U-shape, in this case) appear. Once we use an appropriate detrending model, we use the detrended series –i.e., the residuals– for furthering (ARMA) modeling.

Non-Stationary Models: Deterministic Trend

Example: We detrend U.S. Stock Prices

```
T <- length(x_P)                                    # length of series
trend <- c(1:T)                                     # create trend
det_P <- lm(x_P ~ trend)                            # regression to get detrended e
detrend_P <- det_P$residuals
plot(detrend_P, type="l", col="blue", ylab ="Detrended U.S. Prices", xlab ="Time")
title("Detrended U.S. Stock Prices")
```

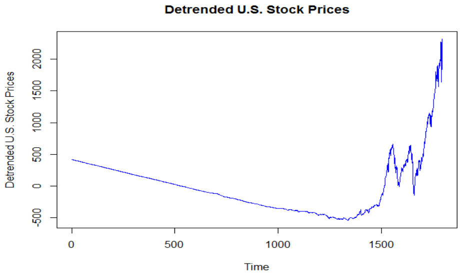


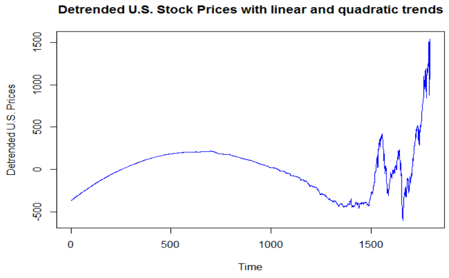
Note: Extra pattern in detrended series ⇒ Using the wrong model.

Non-Stationary Models: Deterministic Trend

Example: We detrend U.S. Stock Prices adding a square trend

```
trend2 <- trend^2
det_P <- lm(x_P ~ trend + trend2)            # regression to get detrended e
detrend_P <- det_P$residuals
plot(detrend_P, type="l", col="blue", ylab ="Detrended U.S. Prices", xlab ="Time")
title("Detrended U.S. Stock Prices with linear and quadratic trends")
```



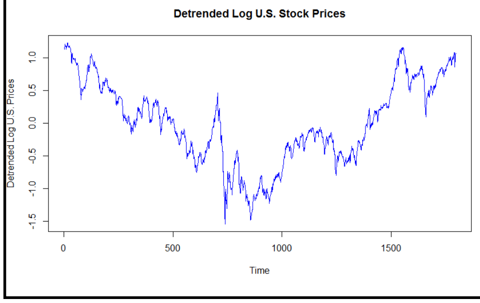


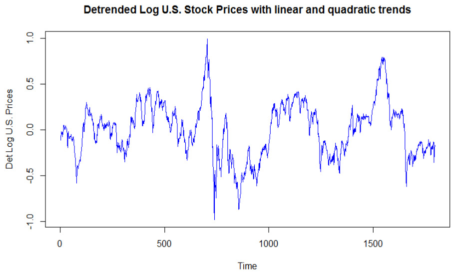
⇒ Still using the wrong model to detrend: Try exponential trend.

Non-Stationary Models: Deterministic Trend

Example: We detrend Log U.S. Stock Prices adding a squared trend

```
l_P <- log(x_P)
det_lP <- lm(l_P ~ trend)                    # regression to get detrended e
detrend_lP <- det_lP$residuals
plot(detrend_lP, type="l", col="blue", ylab ="Detrended Log U.S. Prices", xlab ="Time")
title("Detrended Log U.S. Stock Prices")

det_lP2 <- lm(l_P ~ trend + trend2)          # regression to get detrended e
det_lP2 <- det_lP2$residuals
plot(det_lP2, type="l", col="blue", ylab ="Det Log U.S. Prices", xlab ="Time")
title("Detrended Log U.S. Stock Prices with linear and quadratic trends")
```





Non-Stationary Models: Stochastic Trend

- The more modern approach is to consider trends in time series as a variable trend.

- A variable trend exists when a trend changes in an unpredictable way. Therefore, it is considered *stochastic*.

- Recall the AR(1) model: $y_t = \mu + \phi_1\, y_{t-1} + \varepsilon_t$

- As long as $|\phi_1| < 1$, everything is fine, we have a stationary AR(1) process: OLS is consistent, t-stats are asymptotically normal, etc.

- Now consider the special case where $\phi_1 = 1$:

$$y_t = \mu + y_{t-1} + \varepsilon_t$$

Q: Where is the (stochastic) trend? No t term.

Non-Stationary Models: Stochastic Trend

- Let us replace recursively the lag of y_t on the right-hand side:

$$\begin{aligned} y_t &= \mu + y_{t-1} + \varepsilon_t \\ &= \mu + (\mu + y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \\ &\dots \\ &= y_0 + \underbrace{t\,\mu}_{\boxed{\text{Deterministic trend}}} + \underbrace{\textstyle\sum_{j=0}^{t} \varepsilon_{t-j}}_{\boxed{\text{Accumulation of errors (shocks) – stochastic part}}} \end{aligned}$$

- This process is called a "*random walk with drift*": y_t grows with t.

- Each ε_t shock represents a shift in the intercept. All values of $\{\varepsilon_t\}$ have a 1 as coefficient $\Rightarrow$ each shock never vanishes (permanent).

- We remove the trend by **differencing** y_t

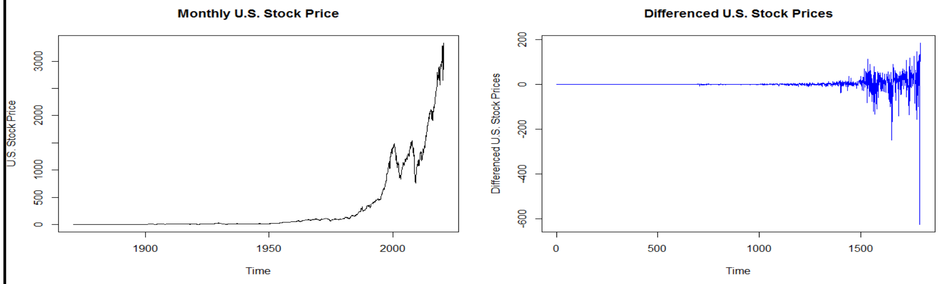
$$\Rightarrow \Delta y_t = (1 - L)\, y_t = \mu + \varepsilon_t$$

Note: Applying the $(1 - L)$ operator to a time series is called *differencing*

Non-Stationary Models: Stochastic Trend

Example: We difference U.S. Stock Prices, using the *diff* R function:

```
diff_P <- diff(x_P)
> plot(diff_P,type="l", col="blue", ylab ="Differenced U.S. Stock Prices", xlab ="Time")
> title("Differenced U.S. Stock Prices")
```



Remark: Trend is gone ⇒ Use first differences for AR modeling.

Non-Stationary Models: Stochastic Trend

- y_t is said to have a *stochastic trend* (ST), since each ε_t shock gives a permanent and random change in the conditional mean of the series.

- For these situations, we use *Autoregressive Integrated Moving Average* (**ARIMA**) models.

- Q: Deterministic or Stochastic Trend?

They appear similar: Both lead to growth over time. The difference is how we think of ε_t. Should a shock today affect y_{t+1}?

– TS: $y_{t+1} = \mu + \beta\,(t+1) + \varepsilon_{t+1}$ $\Rightarrow$ ε_t does not affect y_{t+1}.

– ST: $y_{t+1} = \mu + y_t + \varepsilon_{t+1} = \mu + [\mu + y_{t-1} + \varepsilon_t] + \varepsilon_{t+1}$

$= 2 * \mu + y_{t-1} + \varepsilon_t + \varepsilon_{t+1}$ $\Rightarrow$ ε_t affects y_{t+1}.

(In fact, the shock ε_t has a *permanent* impact.)

ARIMA(p, d, q) Models

- For $p, d, q \geq 0$, we say that a time series $\{y_t\}$ is an *ARIMA* (p, d, q) *process* if $w_t = \Delta^d\, y_t = (1-L)^d\, y_t$ is ARMA(p, q). That is,

$$\phi(L)(1-L)^d\; y_t = \theta(L)\,\varepsilon_t$$

- Applying the $(1-L)$ operator to a time series is called *differencing*.

Notation: If y_t is non-stationary, but $\Delta^d\, y_t$ is stationary, then y_t is *integrated* of order d, or I(d). A time series with *unit root* is I(1). A stationary time series is I(0).

Examples:

Example 1: RW: $y_t = y_{t-1} + \varepsilon_t$.

y_t is non-stationary, but

$$w_t = (1-L)\;y_t = \varepsilon_t \qquad \Rightarrow w_t \sim \text{WN!}$$

Now, y_t ~ ARIMA(0, 1, 0).

ARIMA(p, d, q) Models

Example 2: AR(1) with time trend: $y_t = \mu + \delta\, t + \phi_1\, y_{t-1} + \varepsilon_t$.

y_t is non-stationary, but

$$\begin{aligned} w_t &= (1-L)\, y_t \\ &= \mu + \delta\, t + \phi_1\, y_{t-1} + \varepsilon_t - [\mu + \delta\,(t-1) + \phi_1\; y_{t-2} + \varepsilon_{t-1}]. \\ &= \delta + \phi_1\, w_{t-1} + \varepsilon_t - \varepsilon_{t-1} \qquad \Rightarrow w_t \sim \text{ARIMA(1, 1)}. \end{aligned}$$

Now, y_t ~ ARIMA(1, 1, 1).

- We call both process *first difference stationary*.

Note:

– Example 1: Differencing a series with a unit root in the AR part of the model reduces the AR order.

– Example 2: Differencing can introduce an extra MA structure. We introduced non-invertibility ($\theta_1 = 1$). This happens when we difference a TS series. Detrending should be used in these cases.

ARIMA(p, d, q) Models

- In practice:

A root near 1 of the AR polynomial $\Rightarrow$ differencing
A root near 1 of the MA polynomial $\Rightarrow$ over-differencing

- In general, we have the following results:

- Too little differencing: not stationary.
- Too much differencing: extra dependence introduced.

- Finding the right d is crucial. For identifying preliminary values of d:

- Use a time plot.
- Check for slowly decaying (persistent) ACF/PACF.

Note: There are many formal tests for unit roots. Most popular tests: ADF (Augmented Dickey-Fuller) and PP (Phillips-Perron).

ARIMA Models: Unit Roots 1?

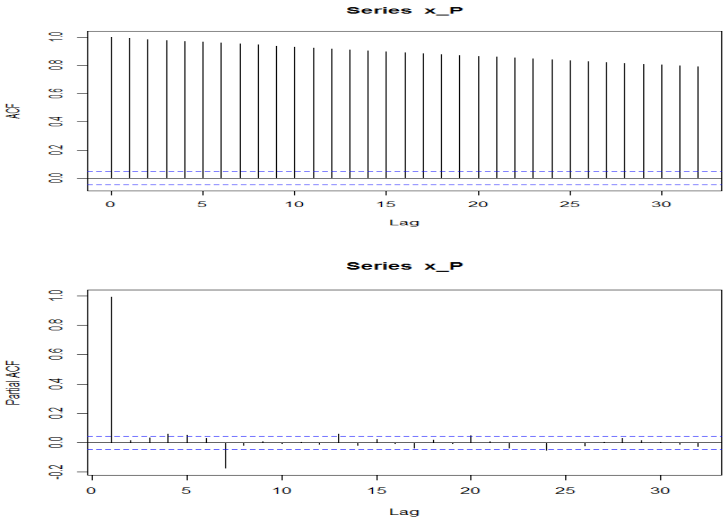
Example 1: Monthly Stock Price levels (1871-2020)

```
acf_P <- acf(x_P)
> acf_P
Autocorrelations of series ‘x_p’, by lag

    0     1     2     3     4     5     6     7     8     9    10    11
1.000 0.992 0.984 0.977 0.971 0.966 0.961 0.954 0.946 0.938 0.931 0.924
   12    13    14    15    16    17    18    19    20    21    22    23
0.917 0.911 0.904 0.897 0.891 0.884 0.877 0.871 0.865 0.860 0.854 0.848
   24    25    26    27    28    29    30    31    32
0.841 0.834 0.827 0.821 0.815 0.809 0.803 0.797 0.790
```

Very high autocorrelations. Looks like $\phi_1 \approx 1$.

ARIMA Models – Unit Roots 1: ACF & PACF

Example 1: Monthly Stock Price levels (1871-2020)



ARIMA Models: Unit Roots 2?

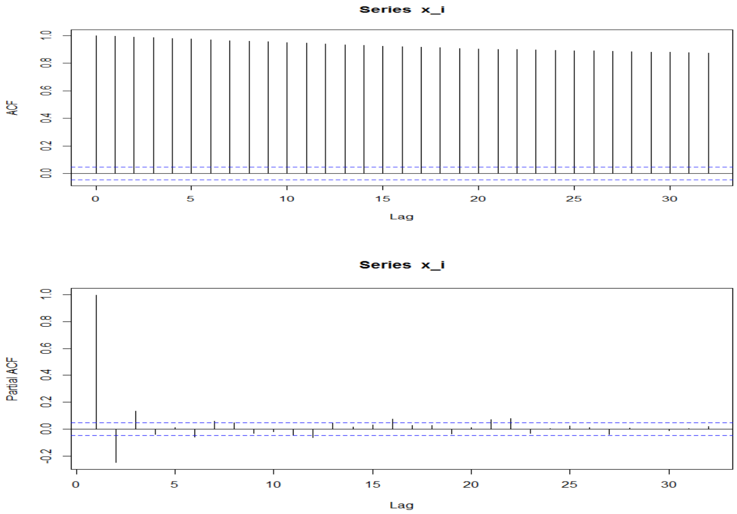
Example 2: Monthly Interest Rates (1871-2020)

```
acf_i <- acf(x_i)
> acf_i
Autocorrelations of  series 'x_i', by lag

    0     1     2     3     4     5     6     7     8     9    10    11
1.000 0.996 0.990 0.985 0.980 0.975 0.970 0.965 0.960 0.956 0.951 0.946
   12    13    14    15    16    17    18    19    20    21    22    23
0.940 0.934 0.929 0.924 0.919 0.915 0.912 0.908 0.904 0.901 0.899 0.896
   24    25    26    27    28    29    30    31    32
0.894 0.891 0.889 0.887 0.884 0.882 0.879 0.877 0.874
```

Very high autocorrelations. Looks like $\phi_1 \approx 1$.

ARIMA Models – Unit Roots 2: ACF & PACF

Example 2: Monthly Interest Rates (1871-2020)



ARIMA Models – Random Walk

- A **random walk** (**RW**) is a process where the current value of a variable is composed of the past value plus an error term defined as a white noise (a normal variable with zero mean and variance one).

- RW is an ARIMA(0,1,0) process

$$y_t = y_{t-1} + \varepsilon_t \quad \Rightarrow \Delta y_t = (1 - L) y_t = \varepsilon_t, \qquad \varepsilon_t \sim WN(0, \sigma^2).$$

- Popular model. Used to explain the behavior of financial assets, unpredictable movements (Brownian motions, drunk persons).

- A special case (limiting) of an AR(1) process: a **unit-root** process.

- Implication: $\text{E}[y_{t+1} | I_t] = y_t \qquad \Rightarrow \Delta y_t$ is absolutely random.

- Thus, a RW is nonstationary, and its variance increases with t.

ARIMA Models – RW with Drift

- Change in y_t is partially deterministic (μ) and partially stochastic.

$$y_t - y_{t-1} = \Delta y_t = \mu + \varepsilon_t$$

- Recall that y_t can also be written as

$$y_t = y_0 + t\,\mu \;+ \textstyle\sum_{j=0}^{t} \varepsilon_{t-j}$$

$\Rightarrow \varepsilon_t$ has a permanent effect on the mean of y_t.

- Recall the difference between conditional and unconditional forecasts:

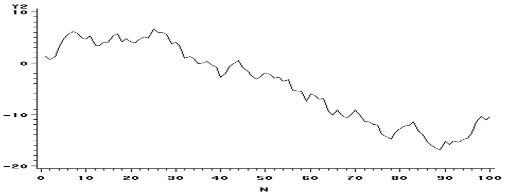
$\text{E}[y_t] = y_0 + t\,\mu$ (Unconditional forecast)

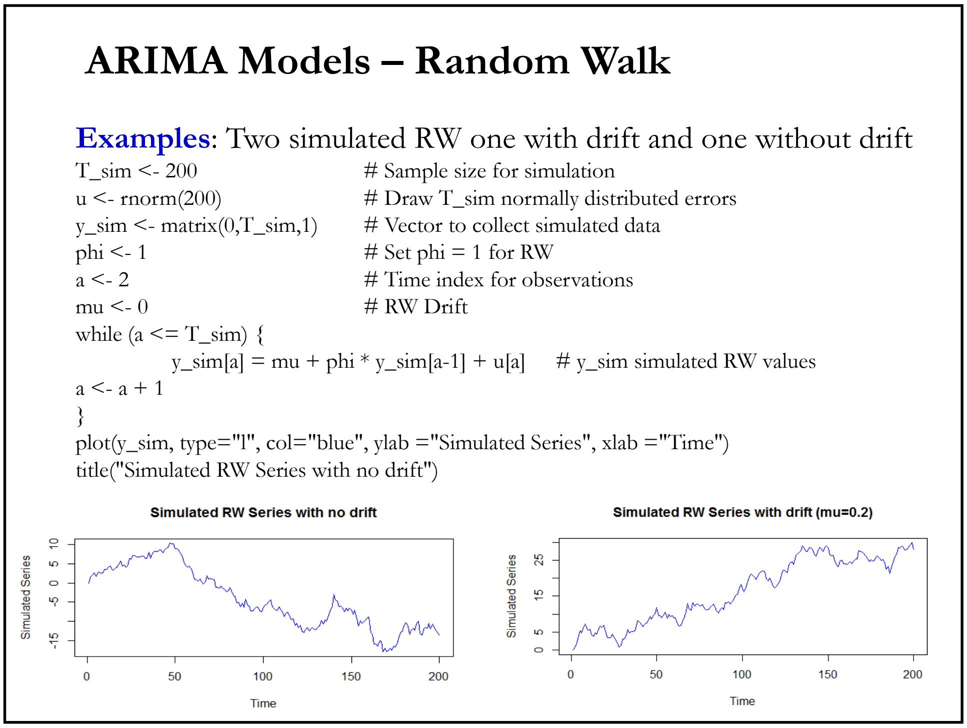
$\text{E}[y_{t+s}\,|\,y_t] = y_t + s\,\mu$ (Conditional forecast)

ARIMA Models – Random Walk

Examples: A simulated RW in R

```
T_sim <- 200
u <- rnorm(200)                # Draw T_sim normally distributed errors
y_sim <- matrix(0,T_sim,1)
rho <- 1                       # Change to create different correlation patterns
a <- 2
mu <- 0                        # Time index for observations
while (a <= T_sim) {
        y_sim[a] = mu + rho * y_sim[a-1] + u[a] # y_sim simulated autocorrelated values
a <- a + 1
}
plot(y_sim, type="l", col="blue", ylab ="Simulated Series", xlab ="Time")
title("Simulated RW Series with no drift")
```





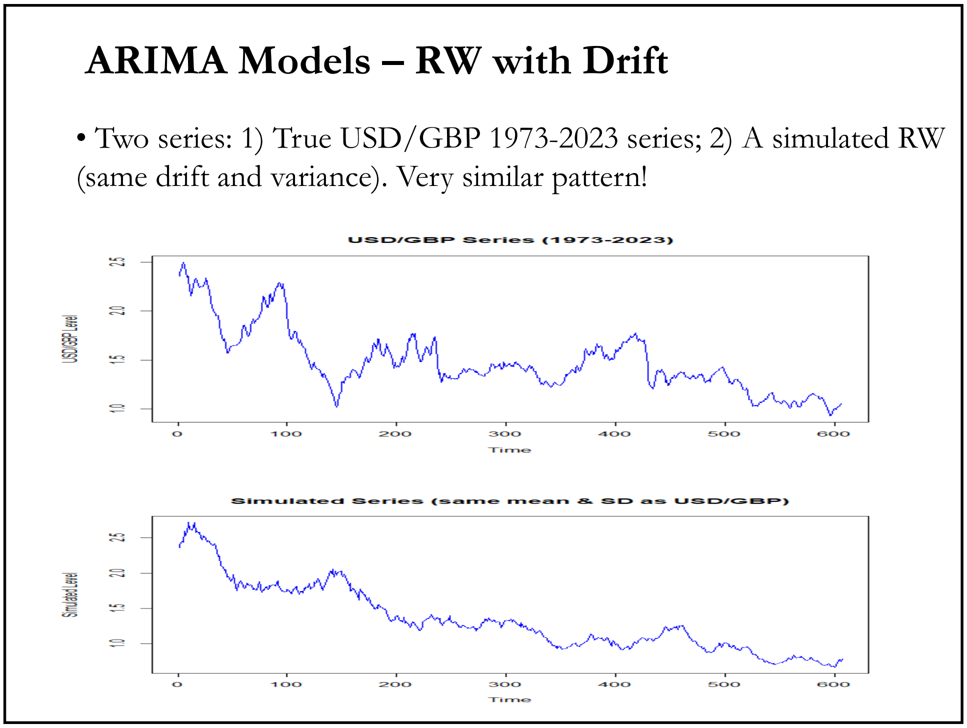
ARIMA Models – Random Walk

Examples: Two simulated RW one with drift and one without drift

```
T_sim <- 200                    # Sample size for simulation
u <- rnorm(200)                 # Draw T_sim normally distributed errors
y_sim <- matrix(0,T_sim,1)      # Vector to collect simulated data
phi <- 1                        # Set phi = 1 for RW
a <- 2                          # Time index for observations
mu <- 0                         # RW Drift
while (a <= T_sim) {
        y_sim[a] = mu + phi * y_sim[a-1] + u[a]      # y_sim simulated RW values
a <- a + 1
}
plot(y_sim, type="l", col="blue", ylab ="Simulated Series", xlab ="Time")
title("Simulated RW Series with no drift")
```

Simulated RW Series with no drift

Simulated RW Series with drift (mu=0.2)



ARIMA Models – RW with Drift

- Two series: 1) True USD/GBP 1973-2023 series; 2) A simulated RW (same drift and variance). Very similar pattern!

USD/GBP Series (1973-2023)

Simulated Series (same mean & SD as USD/GBP)