Autoregression and Moving Average

- Box-Jenkins (1976)
- No economic theory. For fitting and prediction only.

Time Series Probability Model

- $X(t, \omega)$
- For a fixed $t = 1, \ldots, T$, $X(t, \cdot)$ is a random variable
- For a fixed $\omega \in \Omega$, $X(\cdot, \omega)$ is a sequence

Some Simple Models

- White noise: $(e_t)_{t=-\infty}^{\infty}$:
    - $E[e_t] = 0$, $E[e_t^2] = \sigma_e^2$, and $E[e_t, e_s] = 0$ for all $t \neq s$.

ARMA

- AR(p)

$$y_t = \mu + \gamma_1 y_{t-1} + \cdots \gamma_p y_{t-p} + e_t$$

- MA(q)

$$y_t = \mu + e_t - \theta_1 e_{t-1} - \theta_q e_{t-q} + e_t$$

- ARMA(p,q)

$$(1 - \Gamma(L)) y_t = \mu + \Theta(L) e_t$$

Stationarity: in AR form whether all roots lies out of the unit cycle.

Autocorrelation Pattern

- MA(q): finite dependence
- AR(1): geometric decline
  - $E[y_t] = \mu/(1-\gamma_1)$
  - $\mathrm{var}[y_t] = \sigma_e^2/(1-\gamma_1^2)$
  - $E[y_t|y_{t-1}] = \mu + \gamma_1 y_{t-1}$
  - $\mathrm{var}[y_t|y_{t-1}] = \sigma_e^2$

Modeling

- Transform into stationary time series by taking logarithm and/or difference.
- Fit ARMA(p,q)

Estimation

- MLE for MA(q)
- MLE for ARMA(p,q)
- OLS for AR(p)

AR(1) without drift

$$y_t = \gamma_1 y_{t-1} + e_t$$

$$\begin{aligned}
\sqrt{T}(\hat{\gamma}_1 - \gamma_1) &= \sqrt{T}\frac{\sum_{t=2}^T y_{t-1}e_t}{\sum_{t=2}^T y_{t-1}^2} \\
&\overset{d}{\to} N\left(0, \frac{\mathrm{var}[y_t]\mathrm{var}[e_t]}{(\mathrm{var}[y_t])^2}\right) \\
&\sim N\left(0, \frac{\sigma_e^2}{\sigma_e^2/(1-\gamma_1^2)}\right) \\
&\sim N\big(0, 1-\gamma_1^2\big)
\end{aligned}$$

What happens when $|\gamma_1| \to 1$?

Unit Root

- ARIMA(1,1,0)

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

- Nonstationary
- Brownian motion: normal innovation
- Random walk

Implication

- mean
- variance
- h-period ahead forecast

The OLS estimator

$$T(\hat{\gamma}_1 - 1) \stackrel{d}{\to} \text{ a stable distribution.}$$

but the asymptotic distribution is not normal.

```
[ ]  library(quantmod, quietly = TRUE)
     getSymbols("^GSPC") # S&P 500

     y = GSPC$GSPC.Close
     plot(y, type = "l")

     tail(y)
```

- Null hypothesis: unit root.

$$\Delta y_t = \mu + (\gamma_1 - 1) y_{t-1} + e_t = \mu + \theta y_{t-1} + e_t$$

  where $\theta = \gamma_1 - 1$. Under the null, $\theta = 0$.

- The t-statistic is the test statistic for the Dicky-Fuller test.
- Under the null, the t-statistic asymptotically follows a pivotal distribution.

- In this numerical example, the test does not reject the null.

Notice: the test is one-sided.

```
library(urca) # package for unit root and cointegration
print( summary(ur.df(y, type = "drift", lags = 0) ) )
```

```
y = arima.sim( model = list(ar =.95), n = 100)
plot(y)
```

```
library(urca); summary( ur.df( y, type = "none", selectlags =
"AIC" ) ) # ,
```

Specification of DF test

- Drift and trend
- Augmented Dicky-Fuller test: add more differenced lag terms

Other tests

- Phillips-Perron test
- KPSS test

Cointegration

In a regression

$$y_t = \beta x_t + e_t$$

- If y_t and x_t are I(1) series
- But a linear combination $e_t = y_t - \beta x_t$ is I(0)

then we say y_t and x_t are cointegrated.

```
data(denmark)
?denmark
```

period: Time index from 1974:Q1 until 1987:Q3.

- `LRM` Logarithm of real money, M2.
- `LRY` Logarithm of real income.
- `LPY` Logarithm of price deflator.
- `IBO` Bond rate.
- `IDE` Bank deposit rate.

```
sjd <- denmark[, c("LRM", "LRY", "IBO", "IDE")]
matplot(sjd[,c(3,4)], type = "l")
```

```
y = sjd[,3]; x = sjd[,4]
reg = lm(y~x)
plot(residuals(reg), type = "l")
```

Source of Cointegration

Common shock is the source of cointegration

For example, if $y_{1t} = \mu_1 + \beta_1 t + e_{1t}$ and $y_{2t} = \mu_2 + \beta_2 t + e_{2t}$, where e_{1t} and e_{2t} are two white noises, then the cointegration vector must be $(1, \theta)$ where

$$\theta = -\beta_1/\beta_2$$

.

Cointegration

More generally, for an m-vector y_t is cointegrated if there exists a parameter vector γ (normalize the first element to be 1) such that $y_t'\gamma$ is I(0).

- The number of linear independent cointegrated vectors is called the **cointegration rank**.
- The cointegration rank arranges from 1 to $m-1$.

Vector Autoregression (VAR)

Christopher Sims (Nobel Prize 2011)

An m-equation system

$$y_t = \mu + \Gamma_1 y_{t-1} + \cdots + \Gamma_p y_{t-p} + v_t$$

where $E[v_t v_t'] = \Omega$.

For prediction purpose, as a reduced-form of structural simultaneous equations.

Estimation

- For consistency and asymptotic normality, use OLS equation by equation
- For asymptotic efficient, use multiple-equation GLS

Invertibility

Write the VAR(p) as

$$(I_m - \Gamma(L))y_t = \mu + v_t$$

where $\Gamma(z) = \Gamma_1 z + \cdots + \Gamma_p z^p$.

Stable means that all roots of the pth order polynomial equation

$$I_m - \Gamma(z) = 0_m$$

lies out of the unit circle.

Impulse Response Function

IRF characterizes the diffusion of an exogenous shock with the dynamic system.

$$\begin{aligned} y_t &= (I_m - \Gamma(L))^{-1}(\mu + v_t) \\ &= \bar{y} + \left(v_t + \sum_{i=1}^{\infty} A_i v_{t-i} \right) \end{aligned}$$

```
[ ]  library(tsDyn)
     data(barry)
     plot(barry)
```

```
[ ]  ## For VAR
     mod_var <- lineVar(barry, lag = 2)
     print(mod_var)
```

```
[ ]
     irf_var = irf(mod_var, impulse = "dolcan", response = c("dolcan",
     "cpiUSA", "cpiCAN"), boot = FALSE)
     print(irf_var)
```

```
[ ]  plot(irf_var)
```

```
[ ]  ## For VECM
     mod_VECM <- VECM(barry, lag = 2, estim="ML", r=2)
     print(mod_VECM)
     irf_vecm = irf(mod_VECM, impulse = "dolcan", response =
     c("dolcan", "cpiUSA", "cpiCAN"), boot = FALSE)
     print(irf_vecm)
```

```
[ ]  plot(irf_vecm)
```

Structural VAR

- Unrestricted VAR: too many parameters? $m + p \cdot m^2 + m(m+1)/2$
- Use economic theory to reduce the number of unknown parameters

VECM Representation

For the m-equation VAR system

$$y_t = \Gamma y_{t-1} + e_t,$$

we can rewrite it as

$$\Delta y_t = (\Gamma - I_m) y_{t-1} + e_t = \Pi y_{t-1} + e_t.$$

- Since LHS is stationary, the $m \times m$ matrix Π on the RHS must only have rank r.
- Otherwise, the RHS will be I(1) and the two sides of the equation are unbalanced.
- VECM is the base for the cointegration rank test (Johansen, 1992).

Numerical Example: Johansen Test

The result shows that there is only 1 cointegration relationship among the 4 time series.

```
[ ]  sjd.vecm = ca.jo(sjd, ecdet = "const", type="eigen", K=2,
     spec="longrun")
     summary(sjd.vecm)
```