Time Series Analysis

Sherzod Akhundjanov

Recommended Readings

1. Tsay, R.S. (2010) *Analysis of Financial Time Series*, Third Edition, Wiley.
   - Chapters: 2, 3, 8
2. Hamilton, J.D. (1994) *Time Series Analysis*, First Edition, Princeton University Press.
   - Chapters: 2, 3, 4, 11, 15, 19, 21
3. Greene, H.W. (2011) *Econometric Analysis*, Seventh Edition, Pearson.
   - Chapters: 20, 21
4. Shumway, R.H., Stoffer, D.S. (2011) *Time Series Analysis and Its Applications: With R Examples*, Third Edition, Springer.
5. Cowpertwait, P.S.P., Metcalfe, A.V. (2009) *Introductory Time Series with R*, Springer.

Introduction

- A **time series** is a collection of observations made at different times on a given system or process
- Example:
  - Quarterly earnings per share of Johnson & Johnson stock
  - Annual global mean land-ocean temperature deviations
  - Daily investment returns on the New York Stock Exchange
- Data is collected at **regular time intervals**, such as hourly, daily, weekly, monthly, quarterly or annually
- A **cross-sectional** data is a collection of observations for many systems or processes at the same point in time

Implementation in R

`ts()`: (Built-in command in R)

- Used to create time-series objects (i.e., to input the data)
- If not used, R will not recognize the data as time series

astsa package:

- Contains time series data sets and various time series methods

Introduction

```
install.packages("astsa")
library(astsa)

# EXAMPLE 1 - JOHNSON & JOHNSON QUATERLY EARNINGS PER SHARE (1960-1980)
data<-jj
?jj
class(data)
print(data)
sum(is.na(data))
length(data)
summary(data)

# TIME SERIES PLOT
output<-ts(data, start=c(1960,1), frequency=4)
plot(output,
     xlab="Time",
     ylab="Earnings Per Share ($)",
     main="J&J Quarterly Earnings Per Share (1960-1980)")

# HISTOGRAM AND EMPIRICAL DISTRIBUTION
hist(data,
     main="Density Plot",
     xlab="J&J Earnings Per Share",
     ylab="density",
     col=NULL,
     prob=TRUE,
     ylim=c(0,0.30))
lines(density(data, kernel="epanechnikov", bw="ucv"),
      col=2)
```

Introduction

```
# EXAMPLE 2 - GLOBAL MEAN LAND-OCEAN TEMPERATURE DEVIATIONS (1880-2009)
data<-gtemp
?gtemp
class(data)
print(data)
sum(is.na(data))
length(data)
summary(data)

# TIME SERIES PLOT
output<-ts(data, start=1880, frequency=1)
plot(output,
     xlab="Time",
     ylab="Mean temperature deviations",
     main="Global mean land-ocean temperature deviations (1880-2009)")

# HISTOGRAM AND EMPIRICAL DISTRIBUTION
hist(data,
     main="Density Plot",
     xlab="Mean temperature deviations",
     ylab="density",
     col=NULL,
     prob=TRUE)
lines(density(data, kernel="epanechnikov", bw="ucv"),
      col=2)
```

Introduction

```r
# EXAMPLE 3 - RETURNS TO THE NEW YORK STOCK EXCHANGE (1984-1991)
data<-nyse
?nyse
class(data)
print(data)
sum(is.na(data))
length(data)
summary(data)

# TIME SERIES PLOT
output<-ts(data)
plot(output,
     xlab="Time",
     ylab="Returns of the NYSE",
     main="Returns of the New York Stock Exchange (1984-1991)")

# HISTOGRAM AND EMPIRICAL DISTRIBUTION
hist(data,
     main="Density Plot",
     xlab="Returns of the NYSE",
     ylab="density",
     col=NULL,
     prob=TRUE,
     ylim=c(0,70))
lines(density(data, kernel="epanechnikov", bw="ucv"),
      col=2)
```

Introduction

What do you see in these plots?

- Trend?
- Seasonal pattern?
- Cyclical pattern?
- Outliers?
- Nonconstant variance?

Introduction

- Why the need for time series modelling
  - An appropriate economic theory to the relationship between series may be available and hence one considers only the statistical relationship of the given series with its past values
  - Sometimes even when the set of explanatory variables are available, it may not be possible to obtain the entire set of such variables to estimate the regression model and one would then have to use only a single series of the dependent variable to forecast the future values
  - The primary objective of time series analysis is to develop models that provide plausible descriptions for the data
- Important applications
  - Governments forecast inflation, unemployment, and GPD growth rates to taylor their policies accordingly
  - Firms may be interested in demand for their product or their market share
  - Housing finance institutions may want to forecast the mortgage interest rate and the demand for housing loans

Characteristics of Time Series Data

- Time series data have certain features that cross-sectional data do not
  - Gauss-Markov and classical linear assumptions
  - No longer can assume random sampling
  - Special attention when applying OLS
- Time series data are almost always correlated with each other (**autocorrelated**)
- We may want to exploit that correlation, or merely to cope with it

Characteristics of Time Series Data

1. **Univariate time series**
   - A time series that consists of single observations recorded over regular time intervals
2. **Multivariate time series**
   - Data on multiple time series

Here, we start with the univariate case, and then generalize the methods to the multivariate case

Set up

- A time series is modelled as the realization of a collection of random variables $Y_1, Y_2, \ldots$ or more generally as $\{Y_t : t \in T\}$, where T is one of the following sets:
  1. $\mathbb{N} = \{1, 2, \ldots\}$
  2. $\mathbb{Z} = \{0, \pm 1, \pm 2, \ldots\}$
  3. $\mathbb{Z}_+ = \{0, 1, 2, \ldots\}$
- We denote realizations of a random variable Y_t as y_t
- Simplifying assumptions:
  - In some applications, it is possible that T is an interval of time, but we shall not consider this in this course
  - The sampling times are **equi-distant**

Moments

- Let $\{Y_t : t \in T\}$ be a time series
- The stochastic behavior of Y_t is determined by the probability density/mass function

$$f_Y(y_{t_1}, y_{t_2}, \ldots, y_{t_k})$$

- ***r*th moment about origin (raw moment)** of random variable Y_t is

$$\mu_r' = \mathrm{E}(Y^r) = \begin{cases} \int_{-\infty}^{\infty} y^r f_Y(y)dy & \text{if } Y \text{ is continuous} \\ \sum_{y \in R(Y)} y^r f_Y(y) & \text{if } Y \text{ is discrete} \end{cases}$$

- The **mean** (first raw moment) of Y_t is

$$\mu_Y \equiv \mu_{Y,t} \equiv \mu_Y(t) = \mathrm{E}(Y_t) = \begin{cases} \int_{-\infty}^{\infty} y f_Y(y)dy & \text{if } Y \text{ is continuous} \\ \sum_{y \in R(Y)} y f_Y(y) & \text{if } Y \text{ is discrete} \end{cases}$$

where the sample mean is

$$\hat{\mu}_Y(t) = \frac{1}{T}\sum_{t=1}^{T} y_t$$

Moments

- We could also use a *weighted mean*

$$\hat{\mu}^*_Y(t) = \frac{1}{T}\sum_{t=1}^{T} w_t y_t \quad \text{where} \quad \sum_{t=1}^{T} w_t = 1$$

  - Usually some weighted means have *smaller variance* than unweighted means
  - But the optimal weights depend on other unknown parameters of the time series
  - Hence, we typically use unweighted means as an estimator of μ_Y

Moments

- ***r*th moment about the mean (central moment)** of random variable Y_t is

$$\mu_r = \mathrm{E}[(Y - \mathrm{E}(Y))^r] = \begin{cases} \int_{-\infty}^{\infty} (y - \mu_Y)^r f_Y(y) dy & \text{if } Y \text{ is continuous} \\ \sum_{y \in R(Y)} (y - \mu_Y)^r f_Y(y) & \text{if } Y \text{ is discrete} \end{cases}$$

- The **variance** (second central moment) of Y_t is

$$\begin{aligned} \gamma_Y(t,t) \equiv \gamma_Y(0) \equiv \mathrm{Var}(Y_t) &= \mathrm{E}[(Y_t - \mu_{Y,t})^2] \\ &= \begin{cases} \int_{-\infty}^{\infty} (y - \mu_{Y,t})^2 f_Y(y) dy & \text{if } Y \text{ is continuous} \\ \sum_{y \in R(Y)} (y - \mu_{Y,t})^2 f_Y(y) & \text{if } Y \text{ is discrete} \end{cases} \end{aligned}$$

where the sample variance is

$$\hat{\gamma}_Y(t,t) \equiv \hat{\gamma}_Y(0) = \frac{1}{T-1} \sum_{t=1}^{T} (y_t - \hat{\mu}_Y(t))^2$$

Quantifying Dependence

- In time series, the nature of the association between Y_t and its past lags, Y_{t-h} for $h > 0$, is important
- Note that Y_t and Y_{t-h} series are of different length
  - Need to have the same number of observations in both series for the computation of autocovariance and autocorrelation functions
- First series $\{Y_t : Y_{h+1}, Y_{h+2}, \ldots, Y_T\}$
  - If $h = 1$, $\{Y_t : Y_2, Y_3, \ldots, Y_T\}$
- Second series $\{Y_{t-h} : Y_1, Y_2, \ldots, Y_{T-h}\}$
  - If $h = 1$, $\{Y_{t-1} : Y_1, Y_2, \ldots, Y_{T-1}\}$
- The association between Y_t and Y_{t-h} is studied for different values of h $(h = 0, 1, 2, 3, \ldots)$

Quantifying Dependence

- The **autocovariance** function of a time series $\{Y_t : t \in T\}$ is

$$\gamma_Y(t, t-h) \equiv \mathrm{Cov}(Y_t, Y_{t-h}) = \mathrm{E}[(Y_t - \mu_{Y,t})(Y_{t-h} - \mu_{Y,t-h})]$$
$$\forall t, t-h \in T, \quad h > 0$$

where the sample autocovariance is

$$\hat{\gamma}_Y(t, t-h) = \frac{1}{T-1} \sum_{t=h+1}^{T} (y_t - \hat{\mu}_Y(t))(y_{t-h} - \hat{\mu}_Y(t-h))$$

- Symmetry
  - $\mathrm{Cov}(Y_t, Y_{t-h}) = \mathrm{Cov}(Y_{t-h}, Y_t), \quad \forall t, t-h \in T$
- Smoothness
  - If a series is smooth, nearby values will be very similar, hence the autocovariance will be large
  - If a series is choppy, even nearby values may be nearly uncorrelated

Quantifying Dependence

- The **autocorrelation** function (**ACF**) of a time series $\{Y_t : t \in T\}$ is

$$\rho(Y_t, Y_{t-h}) \equiv \rho_Y(t, t-h) = \frac{\mathrm{Cov}(Y_t, Y_{t-h})}{\sqrt{\mathrm{Cov}(Y_t, Y_t)\mathrm{Cov}(Y_{t-h}, Y_{t-h})}} = \frac{\gamma_Y(t, t-h)}{\sqrt{\gamma_Y(t,t)\gamma_Y(t-h, t-h)}}$$

  where the sample autocorrelation is

$$\hat{\rho}(Y_t, Y_{t-h}) = \frac{\sum_{t=h+1}^{T}(y_t - \hat{\mu}_Y(t))(y_{t-h} - \hat{\mu}_Y(t-h))}{\sqrt{\sum_{t=h+1}^{T}(y_t - \hat{\mu}_Y(t))^2 \sum_{t=h+1}^{T}(y_{t-h} - \hat{\mu}_Y(t-h))^2}}$$

Return

  - $-1 \leq \rho_Y(t, t-h) \leq 1,\ h > 0$
  - $\rho_Y(t, t) = \rho_Y(0) = 1$ (correlation with itself)
  - In practice, full sample mean ($\hat{\mu}_Y(t)$) and variance ($\hat{\gamma}_Y(t,t)$) are used to normalize both Y_t and Y_{t-h}
- The ACF gives a better picture of the *linear association (dependence)* between the two variables Y_t and Y_{t-h}

Quantifying Dependence

- The autocorrelation function, when measuring the linear association between $\{Y_t : Y_{h+1}, Y_{h+2}, \ldots, Y_T\}$ and $\{Y_{t-h} : Y_1, Y_2, \ldots, Y_{T-h}\}$ does not control for other lags
  - The ordinary ACF comprises all (*direct* and *indirect*) correlation between Y_t and Y_{t-h}
  - The dependence structure among the intermediate process variables (lags 1 through $h-1$) plays an important role
    - Y_t and Y_{t-2} may not be directly correlated
    - It may be the case that Y_t is correlated with Y_{t-1} and Y_{t-1} is correlated with Y_{t-2}
- The **partial autocorrelation** function (**PACF**) is used to measure the degree of association between Y_t and Y_{t-h} when the effects at other time lags are removed
  - That is, the partial autocorrelation at lag h is the autocorrelation between Y_t and Y_{t-h} that is not accounted for by lags 1 through $h-1$
  - That is, only the *direct* correlation between Y_t and Y_{t-h}

Quantifying Dependence

- Other measures of dependence include
  - **Kendall's** τ (Kendall, 1938, *Biometrika*)

$$\begin{aligned} \rho_\tau &= \text{Prob}[(X - \tilde{X})(Y - \tilde{Y}) > 0] - \text{Prob}[(X - \tilde{X})(Y - \tilde{Y}) < 0] \\ &= \text{E}[\text{sign}(X - \tilde{X})(Y - \tilde{Y})] \end{aligned}$$

  - **Spearman's** ρ (Spearman, 1904, *American Journal of Psychology*)
    - The Spearman correlation coefficient is defined as the Pearson correlation coefficient between the **ranked** (ordered) variables
    - Hence, rank correlation
- Why is there different measures of dependence?
  - Correlation coefficient encounters problems when the distributions are not normal (spherical, in general)

Quantifying Dependence

- Let $\{Y_t : t \in T\}$ and $\{X_s : s \in T\}$ be two different time series
- The **cross covariance** function of Y_t and X_s is

$$\gamma_{Y,X}(t,s) = \mathrm{Cov}(Y_t, X_s) = \mathrm{E}[(Y_t - \mu_{Y,t})(X_s - \mu_{X,s})], \quad s,t \in T$$

- The **cross correlation** function (CCF) of Y_t and X_s is

$$\begin{aligned}\rho(Y_t, X_s) = \rho_{Y,X}(t,s) &= \frac{\mathrm{Cov}(Y_t, X_s)}{\sqrt{\mathrm{Cov}(Y_t, Y_t)\mathrm{Cov}(X_s, X_s)}} \\ &= \frac{\gamma_{Y,X}(t,s)}{\sqrt{\gamma_Y(t,t)\gamma_X(s,s)}}\end{aligned}$$

- The CCF measures the linear association between the Y_t and X_s series at time points t and s, respectively
- In general, $\rho_{Y,X}(t,s) \neq \rho_{X,Y}(t,s)$

Implementation in R

`acf()` (Built-in command in R)

- Computes (and by default plots) estimates of the autocovariance or autocorrelation function
- `type = c("correlation", "covariance", "partial")`
- `plot = c("TRUE", "FALSE")`: The second option produces *numeric* values

`pacf()` (Built-in command in R)

- Used for the partial autocorrelations
- `plot = c("TRUE", "FALSE")`: The second option produces *numeric* values

`ccf()` (Built-in command in R)

- Computes the cross-correlation or cross-covariance of two univariate series
- `type = c("correlation", "covariance")`
- `plot = c("TRUE", "FALSE")`: The second option produces *numeric* values

Implementation in R

`acf2()` (from **astsa** package)

- Computes both ACF and PACF for the given sample
- The zero lag value of the ACF is removed (unlike `acf()` command)

`acf()` and `pacf()` help us identify the process underlying the time series

- Whether the correlation persists over time
- Whether there is a seasonal component in the time series
- Whether the process is stationary or nonstationary

Quantifying Dependence - Example 1

```
# SIMULATED DATA (WHITE NOISE PROCESS)
set.seed(11)
yt<-rnorm(20)

# PLOT ACF AND PACF
par(mfrow=c(2,1))
acf(ts(yt), type="correlation", lag.max=6)
acf(ts(yt), type="correlation", lag.max=6, plot=FALSE)
pacf(ts(yt), lag.max=6)
pacf(ts(yt), lag.max=6, plot=FALSE)

library(astsa)
acf2.data<-acf2(ts(yt), max.lag=6)
print(acf2.data)

# ANALYSIS OF THE LAGS
install.packages("quantmod")
library(quantmod)
yt_1<-Lag(yt, k=1)
yt_2<-Lag(yt, k=2)
yt_3<-Lag(yt, k=3)
yt_4<-Lag(yt, k=4)
yt_5<-Lag(yt, k=5)
yt_6<-Lag(yt, k=6)
data<-data.frame(cbind(Time=1:20, yt, yt_1, yt_2, yt_3, yt_4, yt_5, yt_6))
dim(data)
print(data)
```

Quantifying Dependence - Example 1

```
# SAMPLE CORRELATIONS
cor(data$yt,data$yt)
cor(data$yt[2:nrow(data)],data$Lag.1[2:nrow(data)])   #There is a difference between output
cor(data$yt[3:nrow(data)],data$Lag.2[3:nrow(data)])   #produced by cor() and acf(). acf()
cor(data$yt[4:nrow(data)],data$Lag.3[4:nrow(data)])   #uses sample moments obtained from the
cor(data$yt[5:nrow(data)],data$Lag.4[5:nrow(data)])   #full sample, whereas cor() uses moments
cor(data$yt[6:nrow(data)],data$Lag.5[6:nrow(data)])   #obtained from each sample under
cor(data$yt[7:nrow(data)],data$Lag.6[7:nrow(data)])   #consideration.

# SAMPLE CORRELATIONS (AS IN ACF)
ma.yt<-yt-mean(yt)
sum(ma.yt*ma.yt)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1)],NA), na.rm=TRUE)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1,-2)],rep(NA,2)), na.rm=TRUE)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1,-2,-3)],rep(NA,3)), na.rm=TRUE)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1,-2,-3,-4)],rep(NA,4)), na.rm=TRUE)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1,-2,-3,-4,-5)],rep(NA,5)), na.rm=TRUE)/sum(ma.yt^2)
sum(ma.yt*c(ma.yt[c(-1,-2,-3,-4,-5,-6)],rep(NA,6)), na.rm=TRUE)/sum(ma.yt^2)
```

Quantifying Dependence - Example 2

```
# THE CLASSIC BOX JENKINS AIRLINE DATA (1949-1960)
# MONTHLY TOTALS OF INTERNATIONAL AIRLINE PASSENGERS
data(AirPassengers)
AP<-AirPassengers
class(AP)
start(AP)
end(AP)
frequency(AP)
summary(AP)

# TIME SERIES PLOT
plot(AP)
cycle(AP)
aggregate(AP)
plot(aggregate(AP), type="o")
aggregate(AP, FUN=mean)
boxplot(AP)
boxplot(AP~cycle(AP))
hist(AP, main="Density Plot",
     xlab="Airline passengers",
     ylab="density", col=NULL, prob=TRUE)
lines(density(AP, kernel="epanechnikov", bw="ucv"), col=2)

# ACF & PACF
par(mfrow=c(2,1))
acf(AP, main="International Airline Passengers")
pacf(AP, main="International Airline Passengers")
acf2(AP)
```

Quantifying Dependence - Example 2

Notes:

- The ACF does not tail off quickly
- There is a lot of persistence in air travel from one period to another
- This suggests that the series is likely **nonstationary** (need to stationarize the series)
- We also see **seasonality**: air travel today is much more correlated with the air travel of exactly one year ago
- Blue lines are confidence intervals and any ACF outside the interval is statistically significant (different from 0)

Quantifying Dependence - Example 3

```
# US MONTHLY UNEMPLOYMENT RATE DATA, (1948-2012)
data<-read.csv("http://rci.rutgers.edu/~rwomack/UNRATE.csv", row.names=1)
Urate<-ts(data$VALUE, start=c(1948,1), freq=12)
class(Urate)
time(Urate)

# TIME SERIES PLOT
plot(Urate,
     xlab="Time",
     ylab="Unemployment Rate",
     col=4)
abline(reg=lm(Urate~time(Urate)),
       col=2)

hist(Urate, main="Density Plot", xlab="Unemployment rate",
     ylab="density", col=NULL, prob=TRUE, ylim=c(0,0.40))
lines(density(Urate, kernel="epanechnikov", bw="ucv"), col=2)

# ACF & PACF
par(mfrow=c(2,1))
acf(Urate, main="Unemployment Rate")
pacf(Urate, main="Unemployment Rate")
acf2(Urate)
```

Quantifying Dependence - Example 2

Notes:

- There is a lot of persistence in unemployment rate from one period to another
- ACF decays slowly over time, which is indicative of **nonstationary** process
- A more formal test is obtained through the Dickey-Fuller test

Types of Time Series

Two types of time series:

1. **Stationary time series**
2. **Nonstationary time series**

Stationary Time Series

- **Basic idea:** The statistical properties of the observations tend to remain the same over time
- Why stationarity assumption:
  - The results of classical econometric theory are derived under the assumption that variables of concern are stationary
  - Standard techniques are largely invalid where data is nonstationary
- Two forms of stationarity:
  1. **Strict (or strong) stationarity**
  2. **Weak (or covariance) stationarity**

Stationary Time Series - Strict Stationarity

Strict Stationarity

- A time series Y_t is **strictly stationary** if the joint distribution of every finite collection of variables remain the same under time shifts
- That is, the joint distribution of $\{Y_{t_1}, \ldots, Y_{t_k}\}$ is the **same** as that of $\{Y_{t_1+h}, \ldots, Y_{t_k+h}\}$ for all
  - $t_1, \ldots, t_k \in T$
  - time shifts h (positive or negative)
  - $k \geq 1$

- If Y_t is strictly stationary, then
  - For $k = 1$, the distribution of Y_t is the same as that of Y_{t+h}, for all t, h
  - For $k = 1$ and $h = -1$, the dist of Y_t is the same as that of Y_{t-1} for all t
  - For $k = 1$ and $h = -t$, the dist of Y_t is the same as that of Y_0 for all t
- Thus, under strict stationarity, every Y_t has the same distribution (the distribution is time invariant)
- This implies that the mean, variance, and covariance of the series Y_t are also time invariant

Stationary Time Series - Strict Stationarity

- Similarly, strict stationarity implies:
  - For $k = 2$, the joint bivariate distribution of (Y_{t_1}, Y_{t_2}) is the **same** as that of (Y_{t_1+h}, Y_{t_2+h}) for all t_1, t_2, h
  - For $k = 2$ and $h = -1$, the joint bivariate distribution of (Y_{t_1}, Y_{t_2}) is the **same** as that of (Y_{t_1-1}, Y_{t_2-1}) for all t_1, t_2
  - For $k = 2$ and $h = -t_1$, the joint bivariate distribution of (Y_{t_1}, Y_{t_2}) is the **same** as that of $(Y_0, Y_{t_2-t_1})$ for all t_1, t_2
    - The joint distribution of (Y_{t_1}, Y_{t_2}) depends on t_1 and t_2 only through $t_1 - t_2$
- Strong stationarity is hard to verify

Stationary Time Series - Weak Stationarity

Weak Stationarity

A time series Y_t is **weakly (covariance) stationary** if:

1. The mean function $E(Y_t) = \mu_Y$ is constant (every Y_t has the same mean)
2. The variance function $Var(Y_t) = \sigma_Y^2$ is constant (every Y_t has the same variance)
3. The autocovariance function $Cov(Y_t, Y_{t-h})$ depends on t and $t - h$ only through their difference $|t - (t - h)| = |h|$ (i.e., lag h for $h > 0$)

- Weak stationarity depends only on the first and second moment functions, so is also called **second-order stationarity**
- Strong stationarity (plus finite variance) $\implies$ weak stationarity
- Weak stationarity $\not\Rightarrow$ strong stationarity
  - Unless Y_t is normally distributed

Stationary Time Series - Weak Stationarity

- If Y_t is weakly stationary, then the covariance between Y_t and Y_{t-h} depends on h but not on t or $t-h$, so that we may write the autocovariance as

$$\mathrm{Cov}(Y_t, Y_{t-h}) = \gamma_Y(h) \quad \forall t, h$$

- Similarly, the autocorrelation between Y_t and Y_{t-h} can be written as

$$\rho(Y_t, Y_{t-h}) = \frac{\gamma_Y(t, t-h)}{\sqrt{\gamma_Y(t,t)\gamma_Y(t-h, t-h)}} = \frac{\gamma_Y(h)}{\gamma_Y(0)} = \rho_Y(h)$$

- If the series Y_t is stationary, then it has a tendency to return to a constant mean
- Therefore large values will tend to be followed by smaller values (negative changes), and small values by larger values (positive changes)

Stationary Time Series - Weak Stationarity

Notes:

- White noise process is weakly stationary
- A moving average (MA) process is weakly stationary
- A auto-regressive (AR) process is weakly stationary (under certain conditions)
- A random walk (with or without a drift) process is **not** weakly stationary

Stationary Time Series - Weak Stationarity - Example 1

```
# # STATIONARY SERIES (WHITE NOISE PROCESS)
set.seed(11)
e<-ts(rnorm(500, mean=0, sd=1))
plot(e, main="iid Gaussian white noise", xlab="Time", ylab="e",
     col=2, xlim=c(0,500), ylim=c(-10,10))

# DATA GENERATION PROCESS ANALYSIS
# INSERT BELL SHAPE FOR NORMAL DISTRIBUTION
y<-rnorm(length(seq(1,3,0.0000001)))
d.y<-density(y)
layout(matrix(c(1,2,2,1), ncol=3))
plot(d.y$y, d.y$x, xlab="", ylab="", type='l', ylim=c(-10,10), xlim=c(0,1.5),
     col=1, lwd=2, axes=FALSE, frame.plot=FALSE)
abline(h=0, lty=2, col="grey")
abline(h=c(2*sd(e)), lty=2, col="grey")
abline(h=c(-2*sd(e)), lty=2, col="grey")

# OVERLAY THE DATA
plot(NA, NA, main="iid Gaussian white noise", xlab="Time", ylab="e",
     type="p", ylim=c(-10,10), xlim=c(0,500), col=2, pch=20, cex=1.7,
     cex.lab=1.7, cex.axis=1.7, cex.main=1.9)
for(i in 1:length(e)){
  points(i, e[i], col=2, pch=20, cex=1.7, add=TRUE)
  readline(prompt="Press [enter] to continue")
}
dev.off()
```

Stationary Time Series - Weak Stationarity - Example 2

How does stationarity/nonstationarity affect the analysis?

```
# STATIONARY SERIES (WHITE NOISE PROCESS)
set.seed(10)
series1<-rnorm(n=1000,mean=0,sd=1)
plot(seq(1,1000,1),series1)

abline(h=0, col=2, lty=1)
abline(h=I(0-2*1), col=2, lty=2)
abline(h=I(0+2*1), col=2, lty=2)

abline(h=mean(series1), col=4, lty=1)
abline(h=I(mean(series1)-2*sd(series1)), col=4, lty=2)
abline(h=I(mean(series1)+2*sd(series1)), col=4, lty=2)

ooc<-series1[series1<I(mean(series1)-2*sd(series1)) | I(mean(series1)+series1>2*sd(series1))]
length(ooc)/length(series1)
```

Stationary Time Series - Weak Stationarity - Example 3

How does stationarity/nonstationarity affect the analysis?

```r
# NON-STATIONARY SERIES
set.seed(10)
series2<-c(rnorm(n=500,mean=0,sd=1),rnorm(n=500,mean=1,sd=1))
plot(seq(1,1000,1),series2)
abline(v=500, col=1, lty=1)

lines(seq(1,500,1), rep(0,500), col=2, lty=1)
lines(seq(501,1000,1), rep(1,500), col=2, lty=1)
lines(seq(1,500,1),rep(I(0-2*1),500), col=2, lty=2)
lines(seq(1,500,1),rep(I(0+2*1),500), col=2, lty=2)
lines(seq(501,1000,1),rep(I(1-2*1),500), col=2, lty=2)
lines(seq(501,1000,1),rep(I(1+2*1),500), col=2, lty=2)

abline(h=mean(series2), col=4, lty=1)
abline(h=I(mean(series2)-2*sd(series2)), col=4, lty=2)
abline(h=I(mean(series2)+2*sd(series2)), col=4, lty=2)
```

Test for Stationarity

1. ACF/PACF
2. **Unit root test** (H_0 : Nonstationarity vs H_1 : Stationarity)
   - Dickey-Fuller (DF) test (Dickey and Fuller, 1979, *JASA*)
   - Augmented Dickey-Fuller (ADF) test (Said and Dickey, 1984, *Biometrika*)
   - Unit root (PP) tests (Phillips and Perron, 1988, *Biometrika*)
   - Efficient unit root (ERS) test statistics (Elliot, Rothenberg, and Stock, 1996, *Econometrica*)

- In general, the approach of the tests is to consider Y_t as a sum

$$Y_t = T_t + Z_t + e_t$$

  - T_t is a deterministic component (e.g., time trend) - **source of nonstationarity**
  - Z_t is a stochastic component (e.g., past realizations of Y_t) - **source of nonstationarity**
  - e_t is a white noise process ($e_t \sim$ iid $\mathrm{WN}(0, \sigma^2)$)
- Tests then investigate whether Z_t is present (nonstationarity) or not (stationarity)

Implementation in R

tseries package:

- `adf.test()`: Computes the (augmented) Dickey-Fuller test
- `jarque.bera.test()`: Runs Jarque-Bera test of normality
- `garch()`: Fits a GARCH time series model

Test for Stationarity

```
library(tseries)
library(astsa)

# EXAMPLE 1 - JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
johnson<-jj
adf.test(johnson, alternative="stationary", k=0)    #k=0: Dickey-Fuller test
adf.test(johnson, alternative="stationary")         #Nonzero k: Augmented Dickey-Fuller test

# EXAMPLE 2 - GLOBAL MEAN LAND-OCEAN TEMPERATURE DEVIATIONS (1880-2009)
temp<-gtemp
adf.test(temp, alternative="stationary", k=0)
adf.test(temp, alternative="stationary")

# EXAMPLE 3 - RETURNS OF THE NEW YORK STOCK EXCHANGE (1984-1991)
stock<-nyse
adf.test(stock, alternative="stationary", k=0)
adf.test(stock, alternative="stationary")

# EXAMPLE 4 - BOX & JENKINS MONTHLY AIRLINE DATA (1949-1960)
data(AirPassengers)
AP<-AirPassengers
adf.test(AP, alternative="stationary", k=0)
adf.test(AP, alternative="stationary")

# EXAMPLE 5 - US MONTHLY UNEMPLOYMENT RATE (1948-2012)
data2<-read.csv("http://rci.rutgers.edu/~rwomack/UNRATE.csv", row.names=1)
Urate<-ts(data2$VALUE, start=c(1948,1), freq=12)
adf.test(Urate, alternative="stationary", k=0)
adf.test(Urate, alternative="stationary")
```

Nonstationary Time Series

- A lot of series in economics and finance are **not** weakly stationary and instead show:
  - linear or exponential trend
  - stochastic trend - grow or fall over time or meander without a constant long-run mean
  - increasing variance over time
- Example:
  - Johnson and Johnson earnings
  - Global mean temperature deviations
  - GDP, personal consumption expenditures, investment, exports, imports
  - Industrial production, retail sales
  - Interest rates, foreign exchange rates, prices of assets/commodities
  - Unemployment rate, labor force participation rate
  - Loans, federal debt
- A very slowly decaying ACF suggests nonstationarity and presence of deterministic or stochastic trend in the time series

How to Deal with Nonstationary Series?

- A series that is nonstationary can be made stationary after:
  1. **Transformation**
  2. **Detrending**
  3. **Differencing** (popular approach)

Transformation

- Common transformations include:
  1. **Logarithm** is a proper treatment if Y_t grows exponentially and shows increasing variability over time (or if we want to model growth rate not level)
  2. **Power transformation** (square root, Box-Cox)
- Example:
  - Consider the Johnson & Johnson quarterly returns (1960-1980)
  - The marginal ranges of the Y_t values increase out of proportion over time
  - Either use a transformation to reduce the variability or incorporate it in the model explicitly
  - Log-transformation significantly reduces the marginal ranges of the original time series

Transformation - Example

```r
# TRANSFORMATION

# JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
library(astsa)
data<-jj

# SEASONAL DECOMPOSITION
plot(stl(data,s.window="periodic"), col=4,
     main="Decomposition of J&J quarterly earnings")

# LOG TRANSFORMED DATA
ldata<-log(data)
output<-ts(ldata,start=c(1960,1), frequency=4)

# SEASONAL DECOMPOSITION OF LOG TRANSFORMED DATA
plot(stl(output,s.window="periodic"), col=4,
     main="Decomposition of transformed J&J quarterly earnings")
```

Detrending

- **Detrending** is a proper treatment if Y_t is **trend stationary**
- Parametric functional forms for time trend
  - linear: $\alpha + \delta \cdot t$
  - higher order polynomials: $\alpha + \delta \cdot t^b$
  - logistic: $\frac{\alpha}{1+\beta \exp(\delta t)}$
- A common approach to modelling time series data is based on a decomposition of the form

$$Y_t = \alpha + \underbrace{\delta \cdot t}_{\delta_t} + e_t \quad t \in T$$

  - $\delta \cdot t$ or δ_t is a deterministic **time trend** (upward/downward)
  - α is the intercept
  - e_t is a white noise process ($e_t \sim$ iid $\mathrm{WN}(0, \sigma^2)$) and weakly stationary

Detrending

- Notice that ▸ Return

$$
\begin{aligned}
\mathrm{E}(Y_t) &= \alpha + \delta t \quad \times \text{ (not constant)} \\
\mathrm{Var}(Y_t) &= \mathrm{Var}(e_t) = \sigma^2 \quad \checkmark \text{ (constant)} \\
\mathrm{Cov}(Y_t, Y_{t-h}) &= \mathrm{E}[Y_t - \mathrm{E}(Y_t)][Y_{t-h} - \mathrm{E}(Y_{t-h})] \\
&= \mathrm{E}[\alpha + \delta t + e_t - (\alpha + \delta t)][\alpha + \delta(t-h) + e_{t-h} - (\alpha + \delta(t-h)] \\
&= \mathrm{E}[e_t e_{t-h}] \\
&= \sigma^2 1(t = t - h) \\
&= \sigma^2 1(h = 0) = \gamma_Y(h) \quad \checkmark
\end{aligned}
$$

- Hence Y_t is **not** weakly stationary

Detrending

- Y_t can be made stationary by removing the time trend using a regression of Y_t on intercept and time trend
- Let $\hat{\alpha} + \hat{\delta} \cdot t$ be an (OLS) estimator of $\alpha + \delta \cdot t$
- **Detrending** (i.e., removing the trend from Y_t) is

$$\hat{e}_t = Y_t - (\hat{\alpha} + \hat{\delta} \cdot t)$$

- The residuals $\hat{e}_t$ now help us identify the time series structure and/or plausible model for the time series
- It might be the case that detrending does not produce the residuals that are weakly stationary
- In that case, we need to further model it

Detrending - Example 1

```r
# DETRENDING

# EXAMPLE 1 - JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
library(astsa)
data<-jj

# LOG TRANSFORMATION
ldata<-log(data)

# FIT OLS & DETREND THE DATA
output<-lm(ldata~time(ldata))
ehat<-residuals(output)

# PLOT THE SERIES
plot(ts(data,start=1880), ylim=c(-1,16), col=1, xlab="Time",
     ylab="Quarterly earnings per share",
     main="J&J quarterly earnings per share")
lines(ts(ldata,start=1880), col=2)
lines(ts(ehat,start=1880), col=4)
legend("topleft", c("Data","Log-Transformed Data", "Detrended Log-Transformed Data"),
       col=c(1,2,4), lty=1, bty="n")
```

Detrending - Example 1

```r
# ACF & PACF
par(mfrow=c(3,2))
acf(data, lag.max=40, main="Data")
pacf(data, lag.max=40, main="Data")
acf(ldata, lag.max=40, main="Log-Data")
pacf(ldata, lag.max=40, main="Log-Data")
acf(ts(ehat), lag.max=40, main="Detrended Log-Data")
pacf(ts(ehat), lag.max=40, main="Detrended Log-Data")
```

Detrending - Example 1

Notes:

- Does this series look stationary?
- Not quite yet! There is still some (mild) persistence in the ACF of the residuals
  - Run the Dickey-Fuller test on the residuals
- Remember:
  - We are extracting signal from raw series $\{Y_t\}$
  - Each layer of model components (transformation, time trend, etc.) leaves less and less signal in the error term
  - Hence, eventually error term should become a white noise (**main goal**)
  - That is, the ACF of the residuals must be approximately zero at all lags $h > 1$

Detrending - Example 2

```r
# EXAMPLE 2 - GLOBAL MEAN LAND-OCEAN TEMPERATURE DEVIATIONS (1880-2009)
library(astsa)
data<-gtemp

# FIT OLS & DETREND THE DATA
output<-lm(data~time(data))
ehat<-residuals(output)

# PLOT THE SERIES
plot(ts(data, start=1880), col=1,
     xlab="Time",
     ylab="Mean temperature deviations",
     main="Global mean land-ocean temperature deviations")
lines(ts(ehat, start=1880), col=2)
legend("topleft", c("Data","Detrended Data"), col=c(1,2), lty=1, bty="n")

# ACF & PACF
par(mfrow=c(2,2))
acf(data, lag.max=40, main="Data")
pacf(data, lag.max=40, main="Data")
acf(ts(ehat), lag.max=40, main="Detrended Data")
pacf(ts(ehat), lag.max=40, main="Detrended Data")
```

Differencing

- **Differencing** is a proper treatment if Y_t is **difference stationary**
- Consider a series Y_t evolving according to the following function

$$\begin{aligned} Y_t &= Y_{t-1} + e_t \\ &= Y_0 + \sum_{j=1}^{t} e_j \quad \text{(by repeated substitution)} \end{aligned}$$

where e_t is a white noise process ($e_t \sim \text{iid WN}(0, \sigma^2)$)

- Notice that

$$\begin{aligned} \mathrm{E}(Y_t) &= Y_0 \quad \checkmark \\ \mathrm{Var}(Y_t) &= \sum_{j=1}^{t} \mathrm{Var}(e_j) = t \cdot \sigma^2 \quad \times \end{aligned}$$

Differencing

$$
\begin{aligned}
\mathrm{Cov}(Y_t, Y_{t-h}) &= \mathrm{E}[Y_t - \mathrm{E}(Y_t)][Y_{t-h} - \mathrm{E}(Y_{t-h})] \\
&= \mathrm{E}\left(Y_0 + \sum_{i=h+1}^{t} e_i - Y_0\right)\left(Y_0 + \sum_{i=h+1}^{t} e_{i-h} - Y_0\right) \\
&= \mathrm{E}\left(\sum_{i=h+1}^{t} e_i \sum_{i=h+1}^{t} e_{i-h}\right) \\
&= \mathrm{E}\left(\sum_{i=h+1}^{t} e_i e_{i-h}\right) \quad \forall i = i - h \\
&= \sum_{i=h+1}^{t} \sigma^2 \quad \forall i = i - h \\
&= (t-h)\sigma^2 \neq \gamma_Y(h) \quad \times
\end{aligned}
$$

- Hence Y_t is **not** weakly stationary
- Y_t can be made stationary by first order differencing

Differencing

- The **first order differencing** is

$$\Delta Y_t = Y_t - Y_{t-1} = e_t$$

  which is a **white noise process** and stationary
  - The differenced series is the change between each observation in the original series
  - The differenced series will have only $T - 1$ observations since it is not possible to calculate a difference for the first observation Y_1
  - Differencing helps to stabilize the mean
- We can denote first order differencing with the **backshift operator** as

$$\Delta Y_t = (1 - B)Y_t \quad \text{where} \quad BY_t = Y_{t-1}$$

Differencing

- **Second order differencing** is

$$
\begin{aligned}
\Delta^2 Y_t &= (1-B)^2 Y_t \\
&= (1-2B+B^2)Y_t \\
&= Y_t - 2Y_t B + Y_t BB \\
&= Y_t - 2Y_{t-1} + Y_{t-1}B \\
&= Y_t - 2Y_{t-1} + Y_{t-2}
\end{aligned}
$$

- **Higher order differencing** can be expressed as ▸ Return

$$
\begin{aligned}
\Delta^d Y_t &= (1-B)^d Y_t \\
&= \sum_{j=0}^{d} \binom{d}{j} (-B)^j Y_t, \quad d > 0
\end{aligned}
$$

Differencing

Notes:

- A series which is stationary after being differenced **once** is said to be **integrated of order 1**, or **I(1)**
- In general, a series which is stationary after being differenced d times is said to be **integrated of order d**, or **I(d)**
- A series which is sationary **without** differencing is said to be I(0)
- If the process $\{Y_t\}$ has a linear trend in time, then the first-differenced process $\{\Delta Y_t\}$ has **no** trend
- If the process $\{Y_t\}$ has a quadratic trend in time, then the second-differenced process $\{\Delta^2 Y_t\}$ has **no** trend
- Hence, no need to detrend the series that will be differenced

Differencing - Example 1

```
# DIFFERENCING

# EXAMPLE 1 - GLOBAL MEAN LAND-OCEAN TEMPERATURE DEVIATIONS (1880-2009)
library(astsa)
data<-window(gtemp)

# FIRST DIFFERENCING
ddata<-diff(data, differences=1, lag=1)

# PLOT THE SERIES
plot(data, col=1,
     xlab="Time",
     ylab="Mean temperature deviations",
     main="Global mean land-ocean temperature deviations (1880-2009)")
lines(ddata, col=2)
legend("topleft", c("Data","First-Differenced Data"), col=c(1,2), lty=1, bty="n")

# ACF & PACF
par(mfrow=c(2,2))
acf(data, main="Data")
pacf(data, main="Data")
acf(ddata, main="First Differenced Data")
pacf(ddata, main="First Differenced Data")
```

Differencing - Example 2

```
# EXAMPLE 2 - JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
library(astsa)
data<-jj

# FIRST DIFFERENCING
ddata<-diff(data, differences=1, lag=1)

plot(data, ylim=c(-5,16), col=1, xlab="Time",
     ylab="Earnings per share",
     main="Johnson & Johnson Quarterly Earnings Per Share")
lines(ddata, col=2)
legend("topleft", c("Data","First-Differenced Data"), col=c(1,2), lty=1, bty="n")

# ACF & PACF
par(mfrow=c(2,2))
acf(data, main="Data")
pacf(data, main="Data")
acf(ddata, main="First Differenced Data")
pacf(ddata, main="First Differenced Data")

# FIRST DIFFERENCING THE LOG TRANSFORMED SERIES
dldata<-diff(log(data), differences=1, lag=1)

plot(data, ylim=c(-5,16), col=1, xlab="Time",
     ylab="Earnings per share",
     main="Johnson & Johnson Quarterly Earnings Per Share")
lines(dldata, col=2)
legend("topleft", c("Data","First-Differenced log-Data"), col=c(1,2), lty=1, bty="n")
```

Differencing - Example 2

Notes:

- First-differencing removes the trend
- The series (residuals) oscillate around zero with non-constant (heteroscedastic) variance
- The ACF of the first-differenced series decays quickly
- Try first-differencing on the log-transformed data
  - Logarithmic transformation helps with non-constant variance

Differencing - Example 3

```r
# EXAMPLE 3 - US MONTHLY UNEMPLOYMENT RATE (1948-2012)
data<-read.csv("http://rci.rutgers.edu/~rwomack/UNRATE.csv", row.names=1)
Urate<-ts(data$VALUE, start=c(1948,1), freq=12)

# FIRST DIFFERENCING
ddata<-diff(Urate, differences=1, lag=1)

plot(Urate, ylim=c(-5,16), col=1, xlab="Time",
     ylab="Unemployment rate",
     main="US monthly unemployment rate")
lines(ddata, col=2)
legend("topleft", c("Data","First-Differenced Data"), col=c(1,2), lty=1, bty="n")

# ACF & PACF
par(mfrow=c(2,2))
acf(Urate, main="Data")
pacf(Urate, main="Data")
acf(ddata, main="First Differenced Data")
pacf(ddata, main="First Differenced Data")
```

Differencing - Example 3

Notes:

- The ACF of the first-differenced series tails of quickly
- There are regular spikes at every 12-month (1-year) period, which corresponds to the series being a monthly data
- Each 12-month period is correlated with previous 12-month period, which suggests that there is a periodicity to the series
- We need to control for the seasonal component of the series

Tests for Stationarity

```
# (AUGMENTED) DICKEY-FULLER TEST ON DIFFERENCED SERIES

# EXAMPLE 1 - JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
library(astsa)
johnson<-jj
ddata<-diff(johnson, differences=1, lag=1)
adf.test(ddata, alternative="stationary", k=0)    #k=0: Dickey-Fuller test
adf.test(ddata, alternative="stationary")         #Nonzero k: Augmented Dickey-Fuller test

# EXAMPLE 2 - GLOBAL MEAN LAND-OCEAN TEMPERATURE DEVIATIONS (1880-2009)
temp<-gtemp
ddata<-diff(temp, differences=1, lag=1)
adf.test(ddata, alternative="stationary", k=0)
adf.test(ddata, alternative="stationary")

# EXAMPLE 3 - RETURNS OF THE NEW YORK STOCK EXCHANGE (1984-1991)
stock<-nyse
ddata<-diff(nyse, differences=1, lag=1)
adf.test(ddata, alternative="stationary", k=0)
adf.test(ddata, alternative="stationary")
```

Tests for Stationarity

```
# EXAMPLE 4 - BOX & JENKINS MONTHLY AIRLINE DATA (1949-1960)
data(AirPassengers)
AP<-AirPassengers
ddata<-diff(AP, differences=1, lag=1)
adf.test(ddata, alternative="stationary", k=0)
adf.test(ddata, alternative="stationary")

# EXAMPLE 5 - US MONTHLY UNEMPLOYMENT RATE (1948-2012)
data<-read.csv("http://rci.rutgers.edu/~rwomack/UNRATE.csv", row.names=1)
Urate<-ts(data$VALUE, start=c(1948,1), freq=12)
ddata<-diff(Urate, differences=1, lag=1)
adf.test(ddata, alternative="stationary", k=0)
adf.test(ddata, alternative="stationary")
```

The Box-Jenkins Methodology

The Box-Jenkins methodology

1. Identification
   - Determine the components of time series
   - Whether the series is stationary or not (if not take actions)
   - The underlying process that generates the series
2. Estimation
   - Estimate the parameters of the model
   - Prediction/forecast
3. Diagnostics
   - Check model adequacy

We will spend the bulk of the time on step 1

The Box-Jenkins Methodology - Identification

- Graph data to determine if any transformations are necessary (logarithms, differencing, etc.) to get **weakly stationary** time series
- Examine series for trend (linear/nonlinear), periods of higher volatility, outliers, seasonal patterns, structural breaks, missing data, etc.
- Examine the autocorrelation function (ACF) and the partial autocorrelation (PACF) of the transformed data to determine plausible models to be estimated
  - Comparison of the correlograms (plot of sample ACFs/PACFs on lags) of the time series with the theoretical ACFs and PACFs leads to the selection of the appropriate ARIMA(p, q) model
  - The ACF and the PACF are also used to identify the correct number of lags to include in the model
  - Orders of the AR and MA process is determined by the **significant** autocorrelation and partial autocorrelation coefficients
  - Use Q-statistics to test whether groups of autocorrelations are statistically significant

Components of Time Series Data

- The series Y_t can generally be decomposed into two parts

$$Y_t = \text{ predictable part}_t \ + \ \text{unpredictable part}_t$$

- In other words, given information $Y_1, Y_2, \ldots, Y_{t-1}$

$$\begin{aligned} Y_t &= \mu_{Y,t} + e_t \\ &= \mathrm{E}(Y_t | Y_1, Y_2, \ldots, Y_{t-1}) + e_t \end{aligned}$$

  - $\mu_{Y,t}$ is the conditional mean of Y_t
  - e_t is an iid sequence with $\mu = 0, \sigma^2 = 1$ (often)
- Traditional time series modeling is concerned with $\mu_{Y,t}$
  - Model for $\mu_{Y,t}$: **mean equation**
- Volatility modeling concerns σ_t^2
  - Model for σ_t^2: **volatility equation**

Components of Time Series Data

The pattern in a time series can be broadly classified into 4 components:

1. **Trend**: A long term, relatively smooth, pattern that usually persists for more than one year
2. **Seasonal**: A pattern that appears in a regular interval wherein the frequency of occurrence is within a year or even shorter (e.g., quarterly, monthly)
   - The series oscillates with spikes and valleys throghout
3. **Cyclical**: The repeated pattern that appears in a time series but beyond a frequency of one year
   - It is a wavelike pattern about a long-term trend that is apparent over a number of years
   - Cycles are rarely regular and appear in combination with other components
   - Example: Business cycles

Components of Time Series Data

4. **Random**: The component of a time series that is obtained after the above three components have been extracted out of the series
   - The plot of the residual series should be devoid of any pattern
   - This would indicate that only a random pattern is present

- The idea is to create separate models for these four elements and then combine them, either additively

$$Y_t = T_t + S_t + C_t + e_t$$

or multiplicatively

$$Y_t = T_t \cdot S_t \cdot C_t \cdot e_t$$

Components of Time Series Data

```r
# SEASONAL DECOMPOSITION OF TIME SERIES

# BOX & JENKINS MONTHLY AIRLINE DATA (1949-1960)
data(AirPassengers)

# METHOD 1: SEASONAL DECOMPOSTION OF TIME SERIES BY LOESS
plot(stl(AirPassengers, s.window="periodic"),
     col=4,
     main="Decomposition of Air Passenger Data")

# METHOD 2: SEASONAL DECOMPOSTION OF TIME SERIES BY MOVING AVERAGES
plot(decompose(AirPassengers, type="additive"),
     col=4)
```

Components of Time Series Data

Notes:

- A bar at the right hand side of each graph is there for a relative comparison of the magnitudes of each component
  - For this data the change in trend is greater than the variation due to the monthly variation
- The residuals do not look convincingly random, which they should
- There is still some (seasonal) patterns that should be extracted out of the residuals
- Need more rigorous time series models

Univariate Time Series Models

- **White noise process**
- **Auto-regressive (AR) model**
- **Auto-regressive distributed lag (ARDL) model**
- **Random walk (with and without drift) model**
- **Moving average (MA) model**
- **Auto-regressive moving average (ARMA) model**
- **Auto-regressive integrated moving average (ARIMA) model**
  - Box and Jenkins (1970) *Time Series Analysis: Forecasting and Control*
- **Seasonal auto-regressive integrated moving average (SARIMA) model**
- **Exponential smoothing**
- **Auto-regressive conditional heteroscedasticity (ARCH) model**
  - Engle (1982) *Econometrica*
- **Generalized auto-regressive conditional heteroscedasticity (GARCH) model**
  - Bollerslev (1986) *Journal of Econometrics*

White Noise

White Noise

- A series is called **white noise** if it is purely random in nature
- The white noise is uncorrelated random variable e_t, $e_t \sim \text{WN}(\mu, \sigma^2)$

$$\text{E}(e_t) = \mu_{e,t} = \mu_e = \mu \quad \forall t \quad \checkmark$$
$$\text{Var}(e_t) = \sigma^2_{e,t} = \sigma^2 < \infty \quad \forall t \quad \checkmark$$

- If, in addition, the e_t's are independent and identically distributed (iid), then $e_t \sim \text{iid WN}(\mu, \sigma^2)$
  - Often times $e_t \sim \text{iid WN}(0, \sigma^2)$
  - Often times $e_t \sim \text{iid N}(0, \sigma^2)$
- The scatter plot of such series across time will indicate no pattern
  - Thus, forecasting the future values of such series is **not** possible
  - The mean is the best forecast of future values of the series

White Noise

- The lag-h *theoretical autocovariance function* for the white noise (and for the weakly stationary processes in general) is

$$\begin{aligned}\mathrm{Cov}(e_t, e_{t-h}) &= \mathrm{E}[(e_t - \mu_{e,t})(e_{t-h} - \mu_{e,t-h})] \\ &= \begin{cases} \sigma^2 & \text{if } t = t-h \\ 0 & \text{if } t \neq t-h \end{cases}\end{aligned}$$

  for $h \geq 0$

- If $\mu_{e,t} = \mathrm{E}(e_t) = 0\ \forall t$, then

$$\mathrm{Cov}(e_t, e_{t-h}) = \mathrm{E}[e_t e_{t-h}] = \sigma^2 1(t = t-h) \quad \checkmark$$

- The *sample autocovariance function* is

$$\hat{\gamma}_e(t, t-h) = \frac{1}{T-1} \sum_{t=h+1}^{T} (e_t - \bar{e})(e_{t-h} - \bar{e})$$

White Noise

- The *theoretical autocorrelation function* for the white noise (and for the weakly stationary processes in general) is

$$\rho_e(h) = \frac{\text{Cov}(e_t, e_{t-h})}{\sqrt{\text{Var}(e_t)\text{Var}(e_{t-h})}} = \frac{\text{Cov}(e_t, e_{t-h})}{\text{Var}(e_t)} = \frac{\gamma_h}{\gamma_0}$$

and the ACFs are given by $\{\rho_1, \rho_2, \dots\}$

- The *sample autocorrelation function* is

$$\hat{\rho}_e(h) = \frac{\sum_{t=h+1}^{T}(e_t - \bar{e})(e_{t-h} - \bar{e})}{\sum_{t=h+1}^{T}(e_t - \bar{e})^2}$$

White Noise

- If e_t is **white noise** and T is large (and some mild conditions hold), then

$$\hat{\rho}_e(h) \sim \mathrm{N}(\rho_e(h), T^{-1}) \quad \text{for each fixed } h$$

- So, we can look for autocorrelations (ACF) outside $\pm\frac{2}{\sqrt{T}}$ as evidence of non-zero autocorrelation

```
# WHITE NOISE PROCESS
set.seed(11)
data<-rnorm(100)

# PLOT ACF AND PACF
par(mfrow=c(2,1))
acf(ts(data), type="correlation")
pacf(ts(data))

library(astsa)
acf2(ts(data))
```

Auto-Regressive Model

AR(p) Process

Auto-regressive model of order p, or AR(p), is

$$Y_t = \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + e_t, \quad t \geq 1$$

- $\beta_1, \ldots, \beta_p$ are real numbers (constants), with $\beta_p \neq 0$
- Can include the intercept term β_0 as well
- $e_t \sim$ iid WN$(0, \sigma^2)$ is the white noise ($e_t \sim$ iid N$(0, \sigma^2)$)
- e_t is uncorrelated with Y_{t-j}, $j = 1, 2, \ldots$

- The RHS contains past (lagged) LHS variables, hence **auto-regression**
- This model shows different types of behavior for different choices of coefficients $\beta_1, \ldots, \beta_p$
  1. If $|\beta| < 1$, the series is stationary
  2. If $|\beta| = 1$, the series is nonstationary (unit-root, random walk process)
  3. If $|\beta| > 1$, the series is explosive

Auto-Regressive Model

- Using the **backshift operator**, $BY_t = Y_{t-1}$, we can write AR(p) process as

$$\begin{aligned} Y_t - \beta_1 Y_{t-1} - \beta_2 Y_{t-2} - \cdots - \beta_p Y_{t-p} &= e_t \\ \underbrace{(1 - \beta_1 B - \beta_2 B^2 - \cdots - \beta_p B^p)}_{\text{auto-regressive operator}} Y_t &= e_t \end{aligned}$$

- The **auto-regressive operator** (or **AR-polynomial**) is

$$\begin{aligned} \beta(B) &= 1 - \beta_1 B - \beta_2 B^2 - \cdots - \beta_p B^p \\ &= 1 - \sum_{j=1}^{p} \beta_j B^j \end{aligned}$$

- In the operator form, the AR(p) model can be written as

$$\beta(B) Y_t = e_t$$

Auto-Regressive Model

- Alternatively, some literature uses the **lag operator**

$$
\begin{aligned}
L^0(Y_t) &= Y_t \\
L^1(Y_t) &= L(Y_t) = Y_{t-1} \\
L^2(Y_t) &= L(L(Y_t)) = Y_{t-2} \\
&\vdots \\
L^p(Y_t) &= L(L^{p-1}(Y_t)) = Y_{t-p}
\end{aligned}
$$

- The inverse of the lag operator is also well defined

$$L^{-p}(Y_t) = Y_{t+p} \quad \text{for } p = 1, 2, \ldots$$

Auto-Regressive Model - Stationarity

- Return to the auto-regressive operator of AR(p) process

$$\beta(B) = 1 - \beta_1 B - \beta_2 B^2 - \cdots - \beta_p B^p$$

- Let $\lambda_1, \lambda_2, \ldots, \lambda_p$ be the p roots that solve the **characteristic equation** of the AR(p) process

$$\begin{aligned}\beta(B) &= 1 - \beta_1 B - \beta_2 B^2 - \cdots - \beta_p B^p = 0 \\ &= (1 - \frac{1}{\lambda_1} B) \cdot (1 - \frac{1}{\lambda_2} B) \cdots (1 - \frac{1}{\lambda_p} B) = 0\end{aligned}$$

- The inverses of p roots, $\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_p}$, are called **characteristic roots** of the AR(p) process

Auto-Regressive Model - Stationarity

Claim

$\{Y_t\}$, $Y_t \sim \text{AR}(p)$, is a covariance stationary if and only if all the roots of $\beta(B) = 0$ lie outside the unit circle $|\lambda_j| > 1, j = 1, 2, \ldots, p$, which implies $|\frac{1}{\lambda_j}| < 1$, $j = 1, 2, \ldots, p$

- The AR(p) model is **weakly stationary** if its characteristic roots lie inside the unit circle

Return

Auto-Regressive Model - Stationarity

- Consider an AR(1) model

$$Y_t = \beta Y_{t-1} + e_t, \quad e_t \sim \text{iid WN}(0, \sigma^2)$$

- The characteristic equation for AR(1) is

$$\beta(B) = 1 - \beta B = 0$$

with a root $\lambda = \frac{1}{\beta}$ and a characteristic root $\frac{1}{\lambda} = \beta$

- The AR(1) model is covariance stationary if and only if

$$|\lambda| > 1$$

or, equivalently,

$$|\beta| < 1$$

Auto-Regressive Model - Testing for Stationarity

Dickey-Fuller Test (Dickey and Fuller, 1979, *JASA*)

- Suppose Y_t follows AR(1) process

$$Y_t = \beta Y_{t-1} + e_t \quad e_t \sim \mathrm{WN}(0, \sigma^2)$$

- Auto-regressive unit root test is

$$\begin{array}{ll} H_0 : \beta = 1 & \text{(unit root, nonstationarity)} \\ H_1 : |\beta| < 1 & \text{stationarity} \end{array}$$

- Fit AR(1) model by least squares and define the test statistic

$$t_{\beta=1} = \frac{\hat{\beta} - 1}{\mathrm{SE}(\hat{\beta})}$$

where $\hat{\beta}$ is the least-squares estimate of β and $\mathrm{SE}(\hat{\beta})$ is the least-squares estimate of the standard error of $\hat{\beta}$

- If $|\beta| < 1$, then $\sqrt{T}(\hat{\beta} - \beta) \xrightarrow{d} \mathrm{N}(0, (1 - \beta^2)^{-1})$

Auto-Regressive Model - Testing for Stationarity

Alternative representation:

- Consider a simple AR(1) model

$$Y_t = \beta Y_{t-1} + e_t$$

- The regression model can be written as

$$\begin{aligned} \Delta Y_t = Y_t - Y_{t-1} &= \beta Y_{t-1} - Y_{t-1} + e_t \\ &= (\beta - 1) Y_{t-1} + e_t \\ &= \eta Y_{t-1} + e_t \end{aligned}$$

- Auto-regressive unit root test is

$$\begin{aligned} &H_0 : \eta = 0 \quad \text{(unit root, nonstationarity)} \\ &H_1 : \eta < 0 \quad \text{stationarity} \end{aligned}$$

- Critical values for the Dickey-Fuller statistic $\text{DF} = \frac{\hat{\eta}}{\text{SE}(\hat{\eta})}$ is obtained from the Dickey–Fuller table

Auto-Regressive Model - Testing for Stationarity

- If $\eta = 0$ (unit root), then Y_t is random walk (nonstationary) process, which is difference stationary, i.e., I(1)
- If $\eta < 0$, then Y_t is stationary
- There are three main versions of the test
  1. Test for a unit root
  $$\Delta Y_t = \eta Y_{t-1} + e_t$$
  2. Test for a unit root with drift
  $$\Delta Y_t = \mu + \eta Y_{t-1} + e_t$$
  3. Test for a unit root with drift and deterministic time trend
  $$\Delta Y_t = \mu + \delta t + \eta Y_{t-1} + e_t$$

  where $\eta = \beta - 1$

Auto-Regressive Model - Testing for Stationarity

Augmented Dickey-Fuller Test (Said and Dickey, 1984, *Biometrika*)

- The testing procedure is similar to the Dickey–Fuller test but it is applied to

$$
\begin{aligned}
\text{Model } A: &\quad \Delta Y_t = \eta Y_{t-1} + \textstyle\sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t \\
\text{Model } B: &\quad \Delta Y_t = \eta Y_{t-1} + \mu + \textstyle\sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t \\
\text{Model } C: &\quad \Delta Y_t = \eta Y_{t-1} + \mu + \delta t + \textstyle\sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t
\end{aligned}
$$

  where the lags ΔY_{t-i} are used to control for the possible higher order autocorrelation

- Unit root test is

$$
\begin{aligned}
H_0 : \eta = 0 &\quad \text{(unit root, nonstationarity)} \\
H_1 : \eta < 0 &\quad \text{stationarity}
\end{aligned}
$$

- Conclusion under H_1:
  1. Model A: Stationary (with zero mean)
  2. Model B: Level stationary (with non-zero mean)
  3. Model C: Trend stationary (stationary around a deterministic trend)

Auto-Regressive Model - Testing for Stationarity

- The number of lags ΔY_{t-i}, $i = 1, 2, \ldots, p$ can be determined based on AIC or BIC
  - Run models with different number of lags
  - The model with the lowest AIC or BIC is the best model
- ADF has very low power against I(0) process that is close to being I(1)
  - That is, it cannot distinguish well highly persistent stationary processes from nonstationary processes
- Including constant and trend in the regression weakens the test
  - Hence model C is the weakest, while model A is the strongest
- If possible, exclude the constant and/or the trend
  - But if they are incorrectly excluded, the test will be biased

Auto-Regressive Model - Testing for Stationarity

Step-by-step approach:

- Step 1. Estimate: $\Delta Y_t = \eta Y_{t-1} + \mu + \delta t + \sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t$
  1. ADF test: If you reject $H_0 : \eta = 0$, Y_t is trend stationary. STOP the process.
  2. ADF test: If you fail to reject $H_0 : \eta = 0$, check the significance of the estimated trend ($H_0 : \eta = \delta = 0$).
     - 2.1 If the trend is significant, Y_t has unit root. STOP the process and repeat the process on difference data from Step 1.
     - 2.2 If the trend is not significant, go to Step 2.
- Step 2. Estimate: $\Delta Y_t = \eta Y_{t-1} + \mu + \sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t$
  1. ADF test: If you reject $H_0 : \eta = 0$, Y_t is stationary. STOP the process.
  2. ADF test: If you fail to reject $H_0 : \eta = 0$, check the significance of the estimated intercept ($H_0 : \eta = \mu = 0$).
     - 2.1 If the intercept is significant, Y_t has unit root. STOP the process and repeat the process on difference data from Step 1.
     - 2.2 If the intercept is not significant, go to Step 3.
- Step 3. Estimate: $\Delta Y_t = \eta Y_{t-1} + \sum_{i=1}^{p-1} \rho_i \Delta Y_{t-i} + e_t$
  1. ADF test: If you reject $H_0 : \eta = 0$, Y_t is stationary. STOP the process.
  2. ADF test: If you fail to reject $H_0 : \eta = 0$, Y_t is random walk. STOP the process and repeat the process on difference data from Step 1.

Auto-Regressive Model - AR(1)

How does $|\beta| < 1$ stationarize the AR(1) series?

- Write out AR(1) model for period t and $t-1$

$$\begin{aligned} Y_t &= \beta Y_{t-1} + e_t \\ Y_{t-1} &= \beta Y_{t-2} + e_{t-1} \end{aligned}$$

- Then,

$$\begin{aligned} Y_t &= \beta(\beta Y_{t-2} + e_{t-1}) + e_t \\ &= \beta^2 Y_{t-2} + \beta e_{t-1} + e_t \end{aligned}$$

- Iterating this k-steps

$$Y_t = \beta^k Y_{t-k} + \sum_{j=0}^{k} \beta^j e_{t-j} \quad k = t-1$$

Auto-Regressive Model - AR(1)

- If Y_t is second-order stationary, indicating $|\beta| < 1$ (according to the claim), then

$$\beta^k Y_{t-k} \xrightarrow{\text{ms}} 0$$

and we are left with model that depends on iid white noises only

$$Y_t = \sum_{j=0}^{t-1} \beta^j e_{t-j}$$

- This model is known as an **infinite order MA** (MA(∞)) process

Auto-Regressive Model - AR(1)

- The mean (first moment) of $\{Y_t\}$ is

$$\begin{aligned} \mu_{Y,t} &= \mathrm{E}(Y_t) \\ &= \sum_{j=0}^{t-1} \beta^j \mathrm{E}(e_{t-j}) \\ &= \mu \cdot \sum_{j=0}^{t-1} \beta^j \end{aligned}$$

- If $\mathrm{E}(e_t) = \mu = 0$, $\forall t$, then

$$\mu_{Y,t} = \mathrm{E}(Y_t) = 0 \quad \forall t \quad \checkmark$$

Auto-Regressive Model - AR(1)

- If $\mathrm{E}(e_t) = \mu \neq 0$, $\forall t$, then

$$\mu_{Y,t} = \mathrm{E}(Y_t) = \mu \cdot (1 + \beta + \beta^2 + \dots + \beta^{t-1})$$

  which depends on t and thus is **not** stationary
- However, for $|\beta| < 1$ (according to the claim) and large enough t,

$$\begin{aligned}\mu_{Y,t} = \mathrm{E}(Y_t) &= \mu \cdot \underbrace{(1 + \beta + \beta^2 + \dots)}_{\frac{1\cdot(1-\beta^{\infty})}{1-\beta}} \\ &= \frac{\mu}{1-\beta} \quad \forall t \quad \checkmark\end{aligned}$$

Auto-Regressive Model - AR(1)

```
#AR(1): E(Y) STABILIZES AT HIGHER t
t<-seq(from=0,to=20,by=1)
beta1<-0.5
beta2<-0.3
mu<-2       #E(e)
EY1<-mu*(cumsum(beta1^t))
EY2<-mu*(cumsum(beta2^t))
plot(t, EY1,
     type="l",
     col=4,
     xlab="t",
     ylab="E(Y)",
     main="Convergence of E(Y)")
lines(t,EY2,col=1)
legend("bottomright", c(expression(paste(beta,"=0.5, ", mu,"=2")),
                        expression(paste(beta,"=0.3, ", mu,"=2"))),
                        col=c(4,1), lty=1, bty="n")
```

Auto-Regressive Model - AR(1)

Notes:

- Nonstationarity is present only for small t, but it vanishes quickly when t is large
- Hence, for simulation purposes, leave out the first few time periods (lags) as the process is likely nonstationary

$$
\begin{aligned}
Y_0 &= 0 \\
Y_1 &= e_1 \\
Y_2 &= \beta Y_1 + e_2 \\
Y_3 &= \beta Y_2 + e_3 \\
&\vdots
\end{aligned}
$$

where $e_t \sim \mathrm{N}(0, \sigma^2)$

- This is called **burn-in** or **spin-up** period (typically, a few hundreds)
- Focus on later periods (lags) as the process becomes stationary

Auto-Regressive Model - AR(1)

- The variance (second moment) of $\{Y_t\}$ is

$$\gamma_Y(0) = \mathrm{Var}(Y_t) = \mathrm{Var}\left(\sum_{j=0}^{t-1} \beta^j e_{t-j}\right) = \sigma^2(1 + \beta^2 + \beta^4 + \cdots + \beta^{2(t-1)})$$

  which depends on t and thus is **not** stationary
- However, for $|\beta| < 1$ (according to the claim) and large enough t,

$$\mathrm{Var}(Y_t) = \sigma^2 \underbrace{(1 + \beta^2 + \beta^4 + \cdots)}_{\frac{1\cdot(1-(\beta^2)^\infty)}{1-\beta^2}} = \frac{\sigma^2}{1-\beta^2} \quad \forall t \quad \checkmark$$

Auto-Regressive Model - AR(1)

- The autocovariance function of $\{Y_t\}$ can be obtained using the MA(∞) representation, for $h \geq 0$,

$$\begin{aligned}\gamma_Y(h) &= \mathrm{Cov}(Y_{t+h}, Y_t) = \mathrm{E}(Y_{t+h}Y_t) \\ &= \mathrm{E}\left[\left(\sum_{j=0}^{\infty}\beta^j e_{t+h-j}\right)\left(\sum_{k=0}^{\infty}\beta^k e_{t-k}\right)\right]\end{aligned}$$

Since $\mathrm{Cov}(e_{t+h-j}, e_{t-k}) = \mathrm{E}(e_{t+h-j}e_{t-k}) = \sigma^2$ if $t+h-j = t-k$ $\implies j = h+k$, then

$$\begin{aligned} &= \sum_{k=0}^{\infty}\beta^{k+h}\beta^k\sigma^2 \\ &= \beta^h\sigma^2\sum_{k=0}^{\infty}\beta^{2k}\end{aligned}$$

Since the summation is a geometric sequence, $\frac{1\cdot(1-(\beta^2)^{\infty})}{1-\beta^2}$ and $|\beta| < 1$,

$$= \frac{\beta^h\sigma^2}{1-\beta^2} \quad \checkmark$$

Auto-Regressive Model - AR(1)

- Therefore,

$$\begin{aligned} \gamma_Y(h) &= \frac{\beta^h \sigma^2}{1-\beta^2} \\ \gamma_Y(h-1) &= \frac{\beta^{h-1} \sigma^2}{1-\beta^2} \\ \gamma_Y(h) &= \beta \cdot \gamma_Y(h-1) \end{aligned}$$

- Autocovariance is a function of h, not t nor $t+h$

Auto-Regressive Model - AR(1)

- The autocorrelation function (ACF) for AR(1) process is

$$\rho_Y(h) = \frac{\gamma_Y(h)}{\gamma_Y(0)} = \frac{\frac{\beta^h \sigma^2}{1-\beta^2}}{\frac{\sigma^2}{1-\beta^2}} = \beta^h, \quad h \geq 0$$

- This implies that for $|\beta| < 1$

$$\begin{array}{l|l|l}
\rho_Y(1) = \rho_Y(t, t-1) = \beta & \rho_Y(1) = \beta = 0.7 & \rho_Y(1) = \beta = -0.7 \\
\rho_Y(2) = \rho_Y(t, t-2) = \beta^2 & \rho_Y(2) = \beta^2 = 0.49 & \rho_Y(2) = \beta^2 = 0.49 \\
\rho_Y(3) = \rho_Y(t, t-3) = \beta^3 & \rho_Y(3) = \beta^3 = 0.343 & \rho_Y(3) = \beta^3 = -0.343 \\
\rho_Y(4) = \rho_Y(t, t-4) = \beta^4 & \rho_Y(4) = \beta^4 = 0.2401 & \rho_Y(4) = \beta^4 = 0.2401 \\
\vdots & \vdots & \vdots
\end{array}$$

- Also,

$$\begin{aligned}
\rho_Y(h) &= \beta^h \\
\rho_Y(h-1) &= \beta^{h-1} \\
\rho_Y(h) &= \beta \cdot \rho_Y(h-1)
\end{aligned}$$

Auto-Regressive Model - AR(1)

- Since $|\beta| < 1$ for weakly stationary $\{Y_t\}$ (according to the claim), then the theoretical ACF of AR(1) process **decays exponentially (tails off quickly)** in either direct or oscilating way
  - $0 < \beta < 1$: exponential decay
  - $-1 < \beta < 0$: oscillating exponential decay (dampened sine wave)
- Whether the observed time series follows AR process or not can thus be determined by observing the variation of the ACF over different lags

Auto-Regressive Model

Partial autocorrelation function (PACF)

- Consider the following system of AR models that can be estimated by OLS

$$
\begin{aligned}
Y_t &= \beta_{0,1} + \beta_{1,1} Y_{t-1} + e_{1,t} \\
Y_t &= \beta_{0,2} + \beta_{1,2} Y_{t-1} + \beta_{2,2} Y_{t-2} + e_{2,t} \\
Y_t &= \beta_{0,3} + \beta_{1,3} Y_{t-1} + \beta_{2,3} Y_{t-2} + \beta_{3,3} Y_{t-3} + e_{3,t} \\
&\vdots
\end{aligned}
$$

- Estimated coefficients $\hat{\beta}_{1,1}, \hat{\beta}_{2,2}, \hat{\beta}_{3,3}, \ldots$ form the sample **partial autocorrelation functions**
  - $\hat{\beta}_{i,i}$ is lag-i sample PACF of Y_t
  - $\hat{\beta}_{i,i}$ shows the added contribution of lag-i, Y_{t-i}, to Y_t over the AR(i-1) model
- If the time series process $\{Y_t\}$ comes from an AR(p) process, sample PACF should have $\hat{\beta}_{j,j}$ close to zero for $j > p$
- The **order** (p) of the AR process can thus be determined by finding the lag after which PACF cuts off to zero

Auto-Regressive Model

- For an AR(p) process with Gaussian white noise

$$T \to \infty \implies \begin{cases} \hat{\beta}_{p,p} \to \beta_p \\ \hat{\beta}_{j,j} \to 0 \text{ for } j > p \end{cases}$$

- In addition, the asymptotic variance of $\hat{\beta}_{j,j}$ for $j > p$ is $\frac{1}{T}$
- This is the reason why R uses 95% confidence interval given by

$$0 \pm \frac{2}{\sqrt{T}}$$

Implementation in R

`ARMAacf()` (Built-in command in R)

- Computes the *theoretical* (*population*) ACF or PACF function for an ARMA process
- `ar=c()`: numeric vector of AR coefficients
- `ma=c()`: numeric vector of MA coefficients
- `pacf = FALSE`: For ACF plot
- `pacf = TRUE`: For PACF plot

`arima.sim()` (Built-in command in R)

- Produces stationary simulated series via 'burn-in'
- `model`: used to specify the coefficients of ARIMA process
- Error variables are assumed $N(0, 1)$
- See R-documentation of the function for further details

Auto-Regressive Model - AR(1) - Example 1

```
# AR(1): BETA=0.7
set.seed(11)
data1<-arima.sim(model=list(ar=0.7), n=100)   #for AR(p), ar=c(...)
plot(data1, xlab="Time", ylab="Y",
     main=expression(paste("AR(1), ",beta,"=0.7")))

# ALTERNATIVE METHOD
set.seed(11)                          #The data will not be quite similar to that produced by
y<-e<-rnorm(100,mean=0, sd=1)  #arima.sim. That's because arima.sim drops first few obs
for(t in 2:100){                      #(burn-in). There is a point, however, after which the
  y[t]<-0.7*(y[t-1]) + e[t]       #two data sets are identical.
}
plot(ts(y))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf<-ARMAacf(ar=c(0.7), lag.max=20, pacf=FALSE)
plot(ar_acf, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("AR(1), ",beta,"=0.7")))
ar_pacf<-ARMAacf(ar=c(0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("AR(1), ",beta,"=0.7")))
acf(data1, ylab="Sample ACF",
    main=expression(paste("AR(1), ",beta,"=0.7")))
pacf(data1, ylab="Sample PACF",
     main=expression(paste("AR(1), ",beta,"=0.7")))
```

Auto-Regressive Model - AR(1) - Example 2

```
# AR(1): BETA=-0.7
set.seed(11)
data2<-arima.sim(model=list(ar=-0.7), n=100)
plot(data2, xlab="Time", ylab="Y",
     main=expression(paste("AR(1), ",beta,"=-0.7")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf2<-ARMAacf(ar=c(-0.7), lag.max=20, pacf=FALSE)
plot(ar_acf2, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("AR(1), ",beta,"=-0.7")))
ar_pacf2<-ARMAacf(ar=c(-0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf2, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("AR(1), ",beta,"=-0.7")))
acf(data2, ylab="Sample ACF",
    main=expression(paste("AR(1), ",beta,"=-0.7")))
pacf(data2, ylab="Sample PACF",
     main=expression(paste("AR(1), ",beta,"=-0.7")))
```

Auto-Regressive Model - AR(2) - Example 3

```r
# AR(2): BETA=(0.2, 0.7)
set.seed(11)
data3<-arima.sim(model=list(order=c(2,0,0), ar=c(0.2,0.7)), n=200)
plot(data3, xlab="Time", ylab="Y",
     main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))

# ALTERNATIVE METHOD
set.seed(11)                          #The data will not be quite similar to that produced by
e<-ts(rnorm(200, mean=0, sd=1))       #arima.sim. That's because arima.sim drops first few obs.
y<-filter(e, method="recursive",      #convolution=MA, recursive=AR
          filter=c(0.2,0.7))          #a vector of filter coefficients (param vals)
plot(y, xlab="Time", ylab="Y",
     main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf3<-ARMAacf(ar=c(0.2,0.7), lag.max=20, pacf=FALSE)
plot(ar_acf3, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))
ar_pacf3<-ARMAacf(ar=c(0.2,0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf3, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))
acf(data3, ylab="Sample ACF",
    main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))
pacf(data3, ylab="Sample PACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,0.7)")))
```

Auto-Regressive Model - AR(2) - Example 4

```
# AR(2): BETA=(0.2, -0.7)
set.seed(11)
data4<-arima.sim(model=list(order=c(2,0,0), ar=c(0.2,-0.7)), n=200)
plot(data4, xlab="Time", ylab="Y",
     main=expression(paste("AR(2), ",beta,"=(0.2,-0.7)")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf4<-ARMAacf(ar=c(0.2,-0.7), lag.max=20, pacf=FALSE)
plot(ar_acf4, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,-0.7)")))
ar_pacf4<-ARMAacf(ar=c(0.2,-0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf4, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,-0.7)")))
acf(data4, ylab="Sample ACF",
    main=expression(paste("AR(2), ",beta,"=(0.2,-0.7)")))
pacf(data4, ylab="Sample PACF",
     main=expression(paste("AR(2), ",beta,"=(0.2,-0.7)")))
```

Auto-Regressive Model - Yule-Walker Equations

- Consider the AR(p) model

$$(Y_t - \mu_Y) = \beta_1(Y_{t-1} - \mu_Y) + \beta_2(Y_{t-2} - \mu_Y) + \cdots + \beta_p(Y_{t-p} - \mu_Y) + e_t$$

- The **Yule-Walker equations** are

$$\begin{aligned}
\mathrm{E}[(Y_t - \mu_Y)(Y_{t-j} - \mu_Y)] &= \beta_1 \mathrm{E}[(Y_{t-1} - \mu_Y)(Y_{t-j} - \mu_Y)] \\
&\quad + \beta_2 \mathrm{E}[(Y_{t-2} - \mu_Y)(Y_{t-j} - \mu_Y)] \\
&\quad \vdots \\
&\quad + \beta_p \mathrm{E}[(Y_{t-p} - \mu_Y)(Y_{t-j} - \mu_Y)] \\
&\quad + \mathrm{E}[e_t(Y_{t-j} - \mu_Y)] \\
\gamma_Y(j) &= \beta_1 \gamma_Y(j-1) \\
&\quad + \beta_2 \gamma_Y(j-2) \\
&\quad \vdots \\
&\quad + \beta_p \gamma_Y(j-p) \\
&\quad + \delta_{0,j}\sigma^2
\end{aligned}$$

Auto-Regressive Model - Yule-Walker Equations

- Equations $j = 1, 2, \ldots, p$ yield a system of p linear equations in unknowns β_j
- In matrix form

$$\begin{bmatrix} \gamma(1) \\ \gamma(2) \\ \vdots \\ \gamma(p) \end{bmatrix} = \begin{bmatrix} \gamma(0) & \gamma(-1) & \gamma(-2) & \cdots & \gamma(-(p-1)) \\ \gamma(1) & \gamma(0) & \gamma(-1) & \cdots & \gamma(-(p-2)) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \gamma(p-1) & \gamma(p-2) & \gamma(p-3) & \cdots & \gamma(0) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}$$

  where $\gamma(-j) = \gamma(j)$
- Given estimates $\hat{\gamma}(j)$, $j = 0, \ldots, p$, the solutions of these equations are the Yule-Walker estimates of β_j

Auto-Regressive Distributed Lag Model

ARDL(p, q) or ADL(p, q)

Auto-regressive distributed lag model of order p, q, or ARDL(p, q) is

$$\begin{aligned} Y_t &= \beta_1 Y_{t-1} + \cdots + \beta_p Y_{t-p} + \delta_1 X_{t-1} + \cdots + \delta_p X_{t-p} + e_t \\ \beta(B) Y_t &= \delta(B) X_{t-1} + e_t \end{aligned}$$

where $\beta(B) = 1 - \beta_1 B - \cdots - \beta_p B^p$ and $\delta(B) = \delta_1 + \delta_2 B + \cdots + \delta_q B^{q-1}$

- k additional predictors can be added, ARDL($p, q_1, \ldots, q_k$)
- Assumptions
  - AR model assumptions
  - $\mathrm{E}(e_t | Y_{t-1}, Y_{t-2}, \ldots, X_{t-1}, X_{t-2}, \ldots) = 0$
  - $\mathrm{Cov}(e_t, Y_{t-j}) = 0$, $\mathrm{Cov}(e_t, X_{t-j}) = 0$, $\mathrm{Cov}(e_t, e_{t-j}) = 0 \ \forall j > 0$
  - (Y_t, X_t) are (strictly) stationary
  - (Y_t, X_t) are ergodic, i.e., (Y_t, X_t) and (Y_{t-j}, X_{t-j}) become independent for $j \to \infty$
  - No perfect multicollinearity

Auto-Regressive Distributed Lag Model

- Estimation
  - OLS is consistent if X_t is treated as given, and uncorrelated with e_t
    - In other words, as long as it can be assumed that e_t is a white noise process (hence stationary), and independent of X_{t-h} and Y_{t-h}, $h = 0, 1, \ldots$
  - OLS is inconsistent if X_t is *simultaneously* determined with Y_t and $\mathrm{E}(X_t e_t) \neq 0$

Granger causality ("predictability")

- Whether forecast of Y_t can be improved by considering X_t
- More formally, $X_t \overset{Gr}{\rightarrow} Y_t$ if, for at least one forecast horizon $h > 0$,

$$\mathrm{MSPE}(Y_{t+h}|F_t) > \mathrm{MSPE}(Y_{t+h}|F_t^*)$$

where $F_t = \{Y_1, \ldots, Y_t\}$ and $F_t^* = \{Y_1, \ldots, Y_t, X_1, \ldots, X_t\}$

- If $X_t \overset{Gr}{\nrightarrow} Y_t$, X_t has no predictive content for Y_t
  - Test $H_0 : X_t \overset{Gr}{\nrightarrow} Y_t$ vs $H_1 : X_t \overset{Gr}{\rightarrow} Y_t$
  - Run F-test on the significance of lags of X_t in ARDL(p, q)

Random Walk

Random Walk without Drift

Random walk without drift is a version of AR(1) model with $\beta_0 = 0$ and $\beta_1 = 1$

$$
\begin{aligned}
Y_t &= Y_{t-1} + e_t \\
&= Y_{t-2} + e_{t-1} + e_t \\
&= Y_{t-3} + e_{t-2} + e_{t-1} + e_t \\
&\vdots \\
&= Y_0 + \sum_{j=1}^{t} e_j \quad \text{(by repeated substitution - MA represt.)}
\end{aligned}
$$

- e_t is a white noise process ($e_t \sim$ iid WN$(0, \sigma^2)$)
- Often $Y_0 = 0$
- The "drift" relates to the time trend (upward/downward)

- Since the coefficient of Y_{t-1} is unity, the weak stationarity condition of an AR(1) model is not satisfied
- Random walk is unit-root nonstationarity process

Random Walk

- The series has a strong memory as it remembers all of the past shocks
  - That is, the shocks are said to have a permanent effect on the series
  - With AR models (infinite order rep.), shocks decay
- In each time period, going from left to right, the value of the variable takes an independent random step up or down, a so-called random walk
  - Present value of the series does not affect the future value of the series
  - A commonly-used analogy is that of a drunkard who staggers randomly to the left or right as he tries to go forward: the path he traces will be a random walk
- It assumes that, at each point in time, the series merely takes a random step away from its last recorded position, with steps whose mean value is zero
  - If the mean step size is some nonzero value μ, the process is said to be a random-walk-with-drift
- The random walk model can also be viewed as a special case of an ARIMA model
  - Specifically, it is an ARIMA(0,1,0) model

Random Walk

- For a random walk process $Y_t = \sum_{j=1}^{t} e_j$, $t \geq 1$, the moments are

$$
\begin{aligned}
\mu_{Y,t} &= \mathrm{E}(Y_t) = \sum_{j=1}^{t} \mathrm{E}(e_j) = 0, \quad t \geq 1 \quad \checkmark \\
\mathrm{Var}(Y_t) &= \sum_{j=1}^{t} \mathrm{Var}(e_j) = t \cdot \sigma^2, \quad t \geq 1 \quad \times \\
\mathrm{Cov}(Y_t, Y_s) &= \mathrm{Cov}\left(\sum_{j=1}^{t} e_j, \sum_{j=1}^{s} e_j\right) \\
&= \mathrm{E}\left[\left(\sum_{j=1}^{t} e_j\right)\left(\sum_{j=1}^{s} e_j\right)\right] - \mathrm{E}\left(\sum_{j=1}^{t} e_j\right)\mathrm{E}\left(\sum_{j=1}^{s} e_j\right) \\
&= \mathrm{E}\left[\left(\sum_{j=1}^{t} e_j\right)\left(\sum_{j=1}^{s} e_j\right)\right] = \min\{t, s\}\sigma^2 \quad \times
\end{aligned}
$$

- Hence Y_t is not weakly stationary but is **difference** stationary
- Hence pure random walk is ARIMA(0,1,0)

Random Walk

```
# RANDOM WALK WITHOUT DRIFT
set.seed(11)
e<-rnorm(500, mean=0, sd=1)
y<-ts(cumsum(e))
plot(ts(y), main="Random Walk without Drift", xlab="Time", ylab="Y")

# ALTERNATIVE METHOD
set.seed(11)
y.a<-e.a<-rnorm(500, mean=0, sd=1)
for(t in 2:500){
  y.a[t]<-y.a[t-1] + e.a[t]
}
plot(ts(y.a), main="Random Walk without Drift", xlab="Time", ylab="Y")

# ACF & PACF
acf(y, main="")
pacf(y, main="")

# DICKEY-FULLER TEST
library(tseries)
adf.test(y, alternative="stationary", k=0)   #k=0: Dickey-Fuller test
adf.test(y, alternative="stationary")

# ACF & PACF FOR THE FIRST-DIFFERENCED SERIES
acf(diff(y, differences=1, lag=1), main="")
pacf(diff(y, differences=1, lag=1), main="")
```

Random Walk

Notes:

- Autocorrelation for random walk is very high and **persistent**
- ACF decays very slowly, which signals that the process is nonstationary
- However, the first difference easily yields white noise process

Random Walk with Drift

Random Walk with Drift

Random walk with drift μ is a version of AR(1) model with $\beta_1 = 1$

$$\begin{aligned}
Y_t &= \mu + Y_{t-1} + e_t \\
&= \mu + \mu + Y_{t-2} + e_{t-1} + e_t \\
&= \mu + \mu + \mu + Y_{t-3} + e_{t-2} + e_{t-1} + e_t \\
&\vdots \\
&= Y_0 + \underbrace{\mu \cdot t}_{\text{time trend}} + \sum_{j=1}^{t} e_j \quad \text{(by repeated substitution)}
\end{aligned}$$

- e_t is a white noise process ($e_t \sim \text{iid WN}(0, \sigma^2)$)
- $\mu \in \mathbb{R}$ is the *time trend* (drift)
- Often $Y_0 = 0$

Random Walk with Drift

- For a random walk with drift process $Y_t = \mu \cdot t + \sum_{j=1}^{t} e_j$, $t \geq 1$, the moments are

$$\begin{aligned} \mu_{Y,t} &= \mathrm{E}(Y_t) = \mu \cdot t + \sum_{j=1}^{t} \mathrm{E}(e_j) = \mu \cdot t, \quad t \geq 1 \quad \times \\ \mathrm{Var}(Y_t) &= \sum_{j=1}^{t} \mathrm{Var}(e_j) = t \cdot \sigma^2, \quad t \geq 1 \quad \times \end{aligned}$$

- Hence Y_t is not weakly stationary, nor trend stationary, but is **difference** stationary
- **Note**: there is a difference between random walk with drift ($Y_t = \mu + Y_{t-1} + e_t = \mu \cdot t + \sum_{j=1}^{t} e_j$) and trend-stationary series ($Y_t = \delta \cdot t + e_t$)
  - The random-walk model with drift: Both the mean and variance are time-dependent
  - The trend-stationary series: Only the mean is time-dependent Details
  - Hence random walk with drift is also ARIMA(0,1,0)

Random Walk with Drift

```
# RANDOM WALK WITH DRIFT
set.seed(11)
e<-rnorm(500, mean=0, sd=1)
y<-ts(cumsum(0.2 + e))          #mu=0.2 (drift size)
plot(y, xlab="Time", ylab="Y", main=expression(paste("Random walk with drift, ", mu,"=0.2")))

# ALTERNATIVE METHOD
set.seed(11)
y<-e<-rnorm(500,mean=0,sd=1)
for(t in 2:500){
  y[t]<-0.2 + y[t-1] + e[t]
}
plot(ts(y), xlab="Time", ylab="Y",
     main=expression(paste("Random walk with drift, ", delta,"=0.2")))

# ACF & PACF
par(mfrow=c(2,2))
acf(y, main="Series")
pacf(y, main="Series")

# DICKEY-FULLER TEST
library(tseries)
adf.test(y, alternative="stationary", k=0)
adf.test(y, alternative="stationary")

# ACF & PACF FOR THE FIRST-DIFFERENCES SERIES
acf(diff(y, differences=1, lag=1), main="First-Differenced Series")
pacf(diff(y, differences=1, lag=1), main="First-Differenced Series")
```

Moving Average Model

MA(q) Process

Moving average model of order q, or MA(q), is

$$Y_t = e_t + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q}, \quad t \geq 1$$

- $\theta_1, \ldots, \theta_q$ are real numbers (constants), with $\theta_q \neq 0$
- $e_t \sim$ iid WN$(0, \sigma^2)$ is the white noise ($e_t \sim$ iid N$(0, \sigma^2)$)
- e_t is uncorrelated with all Y_{t-j} for $j \geq 1$
- We require $\theta_0 = 1$ to ensure identifiability

Moving Average Model

- Using the *backshift operator*, $Be_t = e_{t-1}$, we can write MA(q) process as

$$\begin{aligned} Y_t &= e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} \\ &= \underbrace{\left(1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q\right)}_{\text{moving average operator}} e_t \end{aligned}$$

- The **moving average operator** (or **MA-polynomial**) is

$$\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q = 1 + \sum_{j=1}^{q} \theta_j B^j$$

- In the operator form, the MA(q) model can be written as

$$Y_t = \theta(B) e_t$$

Moving Average Model

- The MA(q) process is always **weakly stationary** for any values of $\theta_1, \dots, \theta_q$, $q \geq 1$
  - Neither the mean nor the variance nor the autocovariance of time series Y_t depend on t
- The mean (first moment) of $\{Y_t\}$ is

$$
\begin{aligned}
\mu_{Y,t} &= \mathrm{E}(Y_t) \\
&= \mathrm{E}(e_t) + \theta_1 \mathrm{E}(e_{t-1}) + \cdots + \theta_q \mathrm{E}(e_{t-q}) \\
&= (1 + \theta_1 + \cdots + \theta_q)\mu \\
&= 0 \qquad [\text{since } e_t \sim \text{iid WN}(0, \sigma^2) \quad \forall t] \quad \checkmark
\end{aligned}
$$

- The variance (second moment) of $\{Y_t\}$ is

$$
\begin{aligned}
\mathrm{Var}(Y_t) &= \mathrm{Var}(e_t) + \theta_1^2 \mathrm{Var}(e_{t-1}) + \cdots + \theta_q^2 \mathrm{Var}(e_{t-q}) \\
&= \sigma^2 (1 + \theta_1^2 + \cdots + \theta_q^2) \\
&= \sigma^2 \sum_{k=0}^{q} \theta_k^2 \quad (\gamma_Y(0)) \quad \checkmark
\end{aligned}
$$

Moving Average Model

- The autocovariance for the MA(q) process is ($\mathrm{E}(e_t) = 0$)

$$
\begin{aligned}
\gamma_Y(h) = \mathrm{Cov}(Y_{t+h}, Y_t) &= \mathrm{E}[(Y_{t+h} - \mu_{Y,t+h})(Y_t - \mu_{Y,t})] \\
&= \mathrm{E}[Y_{t+h} Y_t] \\
&= \mathrm{E}\left[\underbrace{\left(\sum_{j=0}^{q} \theta_j e_{t+h-j} \right)}_{q \text{ lags of } e_{t+h}} \underbrace{\left(\sum_{k=0}^{q} \theta_k e_{t-k} \right)}_{q \text{ lags of } e_t} \right] \\
&= \sum_{j=0}^{q} \sum_{k=0}^{q} \theta_j \theta_k \mathrm{E}(e_{t+h-j} e_{t-k}) \\
&= \sum_{j=0}^{q} \sum_{k=0}^{q} \theta_j \theta_k \sigma^2 \cdot 1(t + h - j = t - k) \\
&= \sum_{k+h=0}^{q} \sum_{k=0}^{q} \theta_{k+h} \theta_k \sigma^2 \cdot 1(j = k + h)
\end{aligned}
$$

Moving Average Model

- Since $h \lessgtr 0$, then

$$\gamma_Y(h) = \begin{cases} \sum_{k=0}^{q-|h|} \theta_{k+h}\theta_k\sigma^2 & \text{if } 0 \leq |h| \leq q \\ 0 & \text{if } |h| > q \end{cases} \quad \checkmark$$

Return

- For the MA(q) process, the above expression reduces to

$$\begin{array}{ll} \gamma_Y(q) = \theta_q\sigma^2 \neq 0 & \forall h = q \\ \gamma_Y(h) = 0 & \forall h > q \end{array} \quad \checkmark$$

- The ACF of the MA(q) model is characterized by

$$\begin{array}{ll} \rho_Y(q) = \frac{\gamma_Y(q)}{\gamma_Y(0)} = \frac{\theta_q}{\sum_{k=0}^{q} \theta_k^2} \neq 0 & \forall h = q \\ \rho_Y(h) = \frac{\gamma_Y(h)}{\gamma_Y(0)} = 0 & \forall h > q \end{array}$$

Moving Average Model

- The theoretical ACF of MA(q) process cuts off to zero after lag q
  - The ACF is thus useful for identifying the order of an MA model in the same way the PACF is useful in identifying the order of an AR model
- The theoretical PACF of MA(q) process decays exponentially (tails off quickly) in either direct or oscilating way
  - $\theta > 0$: oscillating exponential decay (dampened sine wave)exponential decay
  - $\theta < 0$: exponential decay

Moving Average Model - Theoretical Example

- Consider the moving average process $\{Y_t : t \in \mathbb{Z}\}$

$$Y_t = e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2}$$

- The expected value of $\{Y_t : t \in \mathbb{Z}\}$ is

$$\begin{aligned} \mathrm{E}(Y_t) = \mu_{Y,t} &= \mathrm{E}(e_t) + \theta_1 \mathrm{E}(e_{t-1}) + \theta_2 \mathrm{E}(e_{t-2}) \\ &= (1 + \theta_1 + \theta_2)\mu \\ &= 0 \qquad [\text{since } e_t \sim \text{iid WN}(0, \sigma^2) \quad \forall t] \end{aligned}$$

- The variance of $\{Y_t : t \in \mathbb{Z}\}$ is

$$\begin{aligned} \mathrm{Var}(Y_t) &= \mathrm{Var}(e_t) + \theta_1^2 \mathrm{Var}(e_{t-1}) + \theta_2^2 \mathrm{Var}(e_{t-2}) \\ &= \sigma^2(1 + \theta_1^2 + \theta_2^2) \\ &= \sigma^2 \sum_{k=0}^{2} \theta_k^2 \quad (\gamma_Y(0)) \end{aligned}$$

Moving Average Model - Theoretical Example

- Assuming $\mathrm{E}(e_t) = 0$, the autocovariance of $\{Y_t : t \in \mathbb{Z}\}$ is ▸ Formula

$$
\begin{aligned}
\mathrm{Cov}(Y_t, Y_{t-1}) &= \mathrm{E}[(Y_t - \mu_{Y,t})(Y_{t-1} - \mu_{Y,t-1})] \\
&= \mathrm{E}[Y_t Y_{t-1}] \\
&= \mathrm{E}[(e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2})(e_{t-1} + \theta_1 e_{t-2} + \theta_2 e_{t-3})] \\
&= \theta_1 \mathrm{E}[e_{t-1} e_{t-1}] + \theta_2 \theta_1 \mathrm{E}(e_{t-2} e_{t-2}) \\
&= (\theta_1 + \theta_2 \theta_1)\sigma^2 \quad (\gamma_Y(1))
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Cov}(Y_t, Y_{t-2}) &= \mathrm{E}[(Y_t - \mu_{Y,t})(Y_{t-2} - \mu_{Y,t-2})] \\
&= \mathrm{E}[Y_t Y_{t-2}] \\
&= \mathrm{E}[(e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2})(e_{t-2} + \theta_1 e_{t-3} + \theta_2 e_{t-4})] \\
&= \theta_2 \mathrm{E}[e_{t-2} e_{t-2}] \\
&= \theta_2 \sigma^2 \quad (\gamma_Y(2))
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Cov}(Y_t, Y_{t-3}) &= \mathrm{E}[(Y_t - \mu_{Y,t})(Y_{t-3} - \mu_{Y,t-3})] \\
&= \mathrm{E}[(e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2})(e_{t-3} + \theta_1 e_{t-4} + \theta_2 e_{t-5})] \\
&= 0 \quad (\gamma_Y(3))
\end{aligned}
$$

Moving Average Model - Theoretical Example

- The ACF of $\{Y_t : t \in \mathbb{Z}\}$ is

$$\begin{aligned}
\rho_Y(1) &= \frac{\gamma_Y(1)}{\gamma_Y(0)} = \frac{(\theta_1 + \theta_2\theta_1)\sigma^2}{\sigma^2 \sum_{k=0}^{2} \theta_k^2} = \frac{\theta_1 + \theta_2\theta_1}{\sum_{k=0}^{2} \theta_k^2} \neq 0 \\
\rho_Y(2) &= \frac{\gamma_Y(2)}{\gamma_Y(0)} = \frac{\theta_2\sigma^2}{\sigma^2 \sum_{k=0}^{2} \theta_k^2} = \frac{\theta_2}{\sum_{k=0}^{2} \theta_k^2} \neq 0 \\
\rho_Y(h) &= \frac{\gamma_Y(h)}{\gamma_Y(0)} = \frac{0}{\sigma^2 \sum_{k=0}^{2} \theta_k^2} = 0 \quad \forall h > 2
\end{aligned}$$

Moving Average Model

- Consider the first order model, or a MA(1),

$$Y_t = e_t + \theta e_{t-1}$$

where

$$\begin{aligned} e_t &= Y_t - \theta e_{t-1} \\ e_{t-1} &= Y_{t-1} - \theta e_{t-2} \\ e_{t-2} &= Y_{t-2} - \theta e_{t-3} \\ &\vdots \end{aligned}$$

- Then,

$$\begin{aligned} & Y_t = e_t + \theta(Y_{t-1} - \theta e_{t-2}) = e_t + \theta Y_{t-1} - \theta^2 e_{t-2} \\ & Y_t = e_t + \theta Y_{t-1} - \theta^2(Y_{t-2} - \theta e_{t-3}) = e_t + \theta Y_{t-1} - \theta^2 Y_{t-2} + \theta^3 e_{t-3} \\ & \quad \vdots \end{aligned}$$

Moving Average Model

- Thus, Y_t has the **infinite order** auto-regressive (AR(∞)) representation

$$Y_t = \sum_{j=1}^{\infty} [-(-\theta)^j] Y_{t-j} + e_t$$

where $[-(-\theta)^j]$ ensures that when

  - j is odd, the sign on θ^j is positive
  - j is even, the sign on θ^j is negative

- This equation expresses the current shock e_t as a linear combination of the present and past realizations of Y_t
- Intuitively, θ^j should go to zero as j increases because the remote y_{t-j} should have very little impact on the current shock, if any
- Consequently, for an MA(1) model to be plausible, we require $|\theta| < 1$
  - If $|\theta| < 1$, an MA(1) model is said to be **invertible**
  - If $|\theta| = 1$, an MA(1) model is said to be **noninvertible**
- The invertibility of an MA(q) process ($|\theta| < 1$) is the **counterpart (flipside)** of stationarity of an AR(p) process ($|\beta| < 1$)

Moving Average Model - Implementation

Key question:

- Assume an MA(1) process

$$Y_t = e_t + \theta e_{t-1} \quad t = 1, \ldots, 100$$

- Notice that we **don't observe** the error term
- Then, how do we estimate/evaluate the MA model?
- The error term is computed recursively by

$$\begin{aligned}
\hat{e}_t &= Y_t - \theta \hat{e}_{t-1} \\
\hat{e}_{t-1} &= Y_{t-1} - \theta \hat{e}_{t-2} \\
\hat{e}_{t-2} &= Y_{t-2} - \theta \hat{e}_{t-3} \\
&\vdots \\
\hat{e}_2 &= Y_2 - \theta \hat{e}_1 \\
\hat{e}_1 &= Y_1 - \theta \hat{e}_0 \quad \text{(starting point)}
\end{aligned}$$

Return

Moving Average Model - Implementation

What about θ and $\hat{e}_0$?

- The initial estimate of θ ($\hat{\theta}$) is obtained from theoretical autocorrelation function (for MA(q))

$$\rho_Y(h) = \frac{\theta_h}{\sum_{k=0}^{h} \theta_k^2} \quad h = 1, 2, \ldots, q$$

  - q equations and q unknowns
  - $\rho_Y(h)$ can be replaced with estimated (sample) autocorrelation
  - Example: For an MA(1) process ($q = 1$)

Formula

$$\hat{\rho}_Y(1) = \frac{\theta_1}{\theta_0^2 + \theta_1^2} = \frac{\theta_1}{1 + \theta_1^2} \implies \hat{\theta} = \frac{1 \pm \sqrt{1 - 4\hat{\rho}_Y(1)^2}}{2\hat{\rho}_Y(1)}$$

- The value for $\hat{e}_0$ can be obtained using two methods
  - *Conditional Likelihood*: $\hat{e}_0 = 0$ (for moderate/large T)
  - *Unconditional Likelihood*: back-forecasting (Box et al, 1994)

Moving Average Model - MA(1) - Example 1

```
# MA(1): THETA=0.7
set.seed(11)
data1<-arima.sim(model=list(ma=0.7), n=100)   #for MA(p), ma=c(...)
plot(data1, xlab="Time", ylab="Y",
     main=expression(paste("MA(1), ",theta,"=0.7")))

# ALTERNATIVE METHOD
set.seed(11)
y<-e<-rnorm(100,mean=0, sd=1)
for(t in 2:100){
  y[t]<-0.7*(e[t-1]) + e[t]
}
plot(ts(y))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf1<-ARMAacf(ma=c(0.7), lag.max=20, pacf=FALSE)
plot(ar_acf1, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("MA(1), ",theta,"=0.7")))
ar_pacf1<-ARMAacf(ma=c(0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf1, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("MA(1), ",theta,"=0.7")))
acf(data1, ylab="Sample ACF",
    main=expression(paste("MA(1), ",theta,"=0.7")))
pacf(data1, ylab="Sample PACF",
     main=expression(paste("MA(1), ",theta,"=0.7")))
```

Moving Average Model - MA(1) - Example 2

```r
# MA(1): THETA=-0.7
set.seed(11)
data2<-arima.sim(model=list(ma=-0.7), n=100)
plot(data2, xlab="Time", ylab="Y",
     main=expression(paste("MA(1), ",theta,"=-0.7")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf2<-ARMAacf(ma=c(-0.7), lag.max=20, pacf=FALSE)
plot(ar_acf2, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("MA(1), ",theta,"=-0.7")))
ar_pacf2<-ARMAacf(ma=c(-0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf2, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("MA(1), ",theta,"=-0.7")))
acf(data2, ylab="Sample ACF",
    main=expression(paste("MA(1), ",theta,"=-0.7")))
pacf(data2, ylab="Sample PACF",
     main=expression(paste("MA(1), ",theta,"=-0.7")))
```

Moving Average Model - MA(2) - Example 3

```r
# MA(2): THETA=(0.2, 0.7)
set.seed(11)
data3<-arima.sim(model=list(ma=c(0.2,0.7)), n=200)
plot(data3, xlab="Time", ylab="Y",
     main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))

# ALTERNATIVE METHOD
set.seed(11)
e<-ts(rnorm(200, mean=0, sd=1))
y<-filter(e, method="convolution",   #convolution=MA, recursive=AR
          filter=c(0.2,0.7),         #a vector of filter coefficients
          sides=2)                   #for MA process only
plot(y, xlab="Time", ylab="Y",
     main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf3<-ARMAacf(ma=c(0.2,0.7), lag.max=20, pacf=FALSE)
plot(ar_acf3, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))
ar_pacf3<-ARMAacf(ma=c(0.2,0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf3, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))
acf(data3, ylab="Sample ACF",
    main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))
pacf(data3, ylab="Sample PACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,0.7)")))
```

Moving Average Model - MA(2) - Example 4

```
# MA(2): THETA=(0.2, -0.7)
set.seed(11)
data4<-arima.sim(model=list(ma=c(0.2,-0.7)), n=200)
plot(data4, xlab="Time", ylab="Y",
     main=expression(paste("MA(2), ",theta,"=(0.2,-0.7)")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf4<-ARMAacf(ma=c(0.2,-0.7), lag.max=20, pacf=FALSE)
plot(ar_acf4, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,-0.7)")))
ar_pacf4<-ARMAacf(ma=c(0.2,-0.7), lag.max=20, pacf=TRUE)
plot(ar_pacf4, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,-0.7)")))
acf(data4, ylab="Sample ACF",
    main=expression(paste("MA(2), ",theta,"=(0.2,-0.7)")))
pacf(data4, ylab="Sample PACF",
     main=expression(paste("MA(2), ",theta,"=(0.2,-0.7)")))
```

Auto-Regressive Moving Average Model

ARMA(p, q) Process

Auto-regressive moving average model of order p, q, or ARMA(p, q), is

$$\begin{aligned} Y_t &= \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + \\ &\quad + e_t + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} \end{aligned}$$

- $\beta_1, \ldots, \beta_p$ are real numbers (constants), with $\beta_p \neq 0$
- $\theta_1, \ldots, \theta_q$ are real numbers (constants), with $\theta_q \neq 0$
- $e_t \sim$ iid WN$(0, \sigma^2)$ is the white noise ($e_t \sim$ iid N$(0, \sigma^2)$)
- e_t is uncorrelated with all Y_{t-j} for $j \geq 1$
- We require $\theta_0 = 1$ to ensure identifiability

Auto-Regressive Moving Average Model

- In the operator form, the ARMA model can be represented as

$$\beta(B)Y_t = \theta(B)e_t$$

  where the AR-polynomial $\beta(B)$ and the MA-polynomial $\theta(B)$ are defined as before
- Notes
  - $\text{ARMA}(0, q) = \text{MA}(q)$
  - $\text{ARMA}(p, 0) = \text{AR}(p)$

Auto-Regressive Moving Average Model - Some Issues

- **Parameter redundancy**: If $\beta(B)$ (AR-polynomial) and $\theta(B)$ (MA-polynomial) have any common factors, they can be cancelled out, so the model is the same as one with lower orders
- **Causality**: If $\beta(B) \neq 0$ for all $|B| \leq 1$, then Y_t can be written in terms of the present and the past e_t's (MA(∞))
- **Invertibility**: If $\theta(B) \neq 0$ for all $|B| \leq 1$, then e_t can be written in terms of the present and the past Y_t's, and Y_t can be written as an infinite order auto-regression (AR(∞))

Auto-Regressive Moving Average Model - Some Issues

- Consider the process

$$Y_t = 0.4Y_{t-1} + 0.45Y_{t-2} + e_t + e_{t-1} + 0.25e_{t-2}$$

- In the operator form,

$$(1 - 0.4B - 0.45B^2)Y_t = (1 + B + 0.25B^2)e_t$$

- This appears to be an ARMA(2,2) model
- However, when we factorize the AR- and MA-polynomials, we get

$$(1 + 0.5B)(1 - 0.9B)Y_t = (1 + 0.5B)^2 e_t$$

- Thus the factor $(1 + 0.5B)$ is common to both AR- and MA-polynomials, which leads to **redundancy**

Auto-Regressive Moving Average Model - Some Issues

- After cancelling the common factor, the simplified model becomes

$$\begin{aligned} (1-0.9B)Y_t &= (1+0.5B)e_t \\ Y_t &= 0.9Y_{t-1} + e_t + 0.5e_{t-1} \end{aligned}$$

- This is an ARMA(1,1) model
- Since for the reduced model $\beta(B) = (1-0.9B) \neq 0$ for all $|B| \leq 1$, then the process is **causal**

$$Y_t = e_t + \sum_{j=0}^{\infty} \phi_j e_{t-j}$$

Auto-Regressive Moving Average Model

- In general, there is no simple formula for the ACF of the ARMA(p, q) process
- However, when the process is also causal, we can derive an expression for the ACF
- Suppose that $Y_t \sim \text{ARMA}(p, q)$ and is causal
- Then,

$$\text{Cov}(Y_{t+h}, Y_t) = \sigma^2 \sum_{k=1}^{\infty} \phi_k \phi_{k+h} \quad \forall h \geq 0$$

where $\phi_0 = 1$

Auto-Regressive Moving Average Model - Example 1

```
# ARMA(1,1): BETA=0.5, THETA=0.9
set.seed(11)
data1<-arima.sim(model=list(order=c(1,0,1), ar=0.5, ma=0.9), n=100)
plot(data1, xlab="Time", ylab="Y",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=0.9")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf1<-ARMAacf(ar=0.5, ma=0.9, lag.max=20, pacf=FALSE)
plot(ar_acf1, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=0.9")))
ar_pacf1<-ARMAacf(ar=0.5, ma=0.9, lag.max=20, pacf=TRUE)
plot(ar_pacf1, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=0.9")))
acf(data1, ylab="Sample ACF",
    main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=0.9")))
pacf(data1, ylab="Sample PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=0.9")))
```

Auto-Regressive Moving Average Model - Example 2

```r
# ARMA(1,1): BETA=0.5, THETA=-0.9
set.seed(11)
data2<-arima.sim(model=list(order=c(1,0,1), ar=0.5, ma=-0.9), n=100)
plot(data2, xlab="Time", ylab="Y",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=-0.9")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf2<-ARMAacf(ar=0.5, ma=-0.9, lag.max=20, pacf=FALSE)
plot(ar_acf2, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=-0.9")))
ar_pacf2<-ARMAacf(ar=0.5, ma=-0.9, lag.max=20, pacf=TRUE)
plot(ar_pacf2, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=-0.9")))
acf(data2, ylab="Sample ACF",
    main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=-0.9")))
pacf(data2, ylab="Sample PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=0.5, ",theta,"=-0.9")))
```

Auto-Regressive Moving Average Model - Example 3

```r
# ARMA(1,1): BETA=-0.5, THETA=0.9
set.seed(11)
data3<-arima.sim(model=list(order=c(1,0,1), ar=-0.5, ma=0.9), n=100)
plot(data3, xlab="Time", ylab="Y",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=0.9")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf3<-ARMAacf(ar=-0.5, ma=0.9, lag.max=20, pacf=FALSE)
plot(ar_acf3, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=0.9")))
ar_pacf3<-ARMAacf(ar=-0.5, ma=0.9, lag.max=20, pacf=TRUE)
plot(ar_pacf3, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=0.9")))
acf(data3, ylab="Sample ACF",
    main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=0.9")))
pacf(data3, ylab="Sample PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=0.9")))
```

Auto-Regressive Moving Average Model - Example 4

```r
# ARMA(1,1): BETA=-0.5, THETA=-0.9
set.seed(11)
data4<-arima.sim(model=list(order=c(1,0,1), ar=-0.5, ma=-0.9), n=100)
plot(data4, xlab="Time", ylab="Y",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=-0.9")))

# ACF & PACF
par(mfrow=c(2,2))
ar_acf4<-ARMAacf(ar=-0.5, ma=-0.9, lag.max=20, pacf=FALSE)
plot(ar_acf4, type='h', xlab="Lag", ylab="Theoretical ACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=-0.9")))
ar_pacf4<-ARMAacf(ar=-0.5, ma=-0.9, lag.max=20, pacf=TRUE)
plot(ar_pacf4, type='h', xlab="Lag", ylab="Theoretical PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=-0.9")))
acf(data4, ylab="Sample ACF",
    main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=-0.9")))
pacf(data4, ylab="Sample PACF",
     main=expression(paste("ARMA(1,1), ",beta,"=-0.5, ",theta,"=-0.9")))
```

Identification Summary

Linear dependence between Y_t (or e_t) and Y_{t-h} (or e_{t-h})

Process		ACF $\rho(h)$	PACF $\beta_{h,h}$
White noise		$\rho(h) = 0$ for $h > 0$	$\beta_{h,h} = 0$ for $h > 0$
AR(1)	$\beta_1 > 0$	exponential decay $\rho(h) = \beta_1^h$	$\beta_{1,1} = \beta_1$, $\beta_{h,h} = 0$ for $h > 1$
	$\beta_1 < 0$	oscillating decay $\rho(h) = \beta_1^h$	$\beta_{1,1} = \beta_1$, $\beta_{h,h} = 0$ for $h > 1$
AR(p)		Decays towards zero, may oscillate or have a shape of a dampened sine wave	Cuts off after lag p (order) $\beta_{h,h} = 0$ for $h > p$
MA(1)	$\theta_1 > 0$	$\rho(1) > 0$, $\rho(h) = 0$ for $h > 1$	oscillating decay $\beta_{1,1} > 0$, $\beta_{2,2} < 0$, ..
	$\theta_1 < 0$	$\rho(1) < 0$, $\rho(h) = 0$ for $h > 1$	exponential decay $\beta_{h,h} < 0$ for $h > 1$
MA(q)		Cuts off after lag q (order) $\rho(h) = 0$ for $h > q$	Decays towards zero, may oscillate or have a shape of a dampened sine wave

Identification Summary

Process		ACF $\rho(h)$	PACF $\beta_{h,h}$
ARMA(1,1)	$\beta_1 > 0$, $\theta_1 > 0$	exponential decay	oscilating exponential decay
	$\beta_1 > 0$, $\theta_1 < 0$	exponential decay after lag 1	exponential decay
	$\beta_1 < 0$, $\theta_1 > 0$	oscillating exponential decay	oscillating exponential decay
	$\beta_1 < 0$, $\theta_1 > 0$	oscillating exponential decay	exponential decay
ARMA(p, q)		Decays (direct or oscillatory) after lag p or dampened sine wave	Decays (direct or oscillatory) after lag q or dampened sine wave

Auto-Regressive Integrated Moving Average Model

ARIMA(p, d, q) Process

The time series $\{Y_t\}$ follows an auto-regressive integrated moving average model of order p, d, q, or ARIMA(p, d, q) or integrated ARMA(p, q), if $\Delta^d Y_t = (1 - B)^d Y_t$ is:

- stationary (and nonstationary for lower-order differencing)
- ARMA(p, q) process

- Nonstationary time series is said to be integrated of order one, or I(1), if it can be made stationary by applying first differences
- Problem: Determining the order of differencing required to remove time trends (deterministic or stochastic)

Auto-Regressive Integrated Moving Average Model

- Consider ARMA(p, q) process

$$\begin{aligned} Y_t &= \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + \\ &\quad + e_t + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} \end{aligned}$$

- In the backshift operator form

$$\begin{aligned} (1 - \beta_1 B - \beta_2 B^2 - \cdots - \beta_p B^p) Y_t &= (1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q) e_t \\ \beta(B) Y_t &= \theta(B) e_t \end{aligned}$$

  - This is ARIMA$(p, 0, q)$ process, where $\Delta^0 Y_t = Y_t$ $\Delta^d Y_t$

- ARIMA(p, d, q) process is then defined as

$$\beta(B) \Delta^d Y_t = \theta(B) e_t$$

or

$$\begin{aligned} \Delta^d Y_t &= \beta_1 \Delta^d Y_{t-1} + \beta_2 \Delta^d Y_{t-2} + \cdots + \beta_p \Delta^d Y_{t-p} + \\ &\quad + e_t + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} \end{aligned}$$

Seasonal ARIMA Model

SARIMA$(p, d, q)(P, D, Q)_s$ Process

A (multiplicative) seasonal auto-regressive integrated moving average model with non-seasonal order p, d, q and seasonal order P, D, Q, or ARIMA$(p, d, q)(P, D, Q)_s$, is

$$\underbrace{\beta(B)}_{\text{Non-seas AR}} \cdot \underbrace{\beta_s(B_s)}_{\text{Seas AR}} \cdot \underbrace{\Delta_s^d}_{\text{Seas diff}} \cdot \underbrace{\Delta^d}_{\text{Non-seas diff}} \cdot Y_t = \underbrace{\theta(B)}_{\text{Non-seas MA}} \cdot \underbrace{\theta_s(B_s)}_{\text{Seas MA}} \cdot e_t$$

$$(1 - \beta_1 B - \cdots - \beta_p B^p)(1 - \beta_{s,1} B_s - \cdots - \beta_{s,P} B_s^P)\Delta_s^d \Delta^d Y_t =$$
$$(1 + \theta_1 B + \cdots + \theta_q B^q)(1 + \theta_{s,1} B_s + \cdots + \theta_{s,Q} B_s^Q)e_t$$

- Example: $s = 4$ (quarterly), 12 (monthly), 52 (weekly), $\dots$
- If $s = 1$, we are back to non-seasonal (yearly) model
- $BY_t = Y_{t-1}$ and $B_s Y_t = Y_{t-s}$

- Shumway and Stoffer (2011, ch.3.9)

Seasonal ARIMA Model

- Seasonality refers to a regular pattern of changes (i.e., cyclical or periodic behavior) that repeat for s time periods, where s defines the number of time periods until the pattern repeats again
- Example:
  - If we were selling ice cream, we might predict August sales using last years August sales Y_{t-12}
  - Similarly, you could use the past two Augusts and include Y_{t-24}
- In practice most time series contain a seasonal AR or MA component at the same time as regular AR or MA component
- Seasonality usually causes the series to be nonstationary
  - Need to examine differenced data (often seasonality is of secondary importance)
  - **Seasonal differencing**: removes seasonal trend and can also get rid of a seasonal random walk type of nonstationarity
  - **Non-seasonal differencing**: removes overall trend
  - Apply both when both trend and seasonality are present

Seasonal ARIMA Model

- ARIMA(1,1,1)(1,1,1)$_{s=12}$

$$\begin{aligned}
\beta(B)\beta_s(B_s)\Delta_s\Delta Y_t &= \theta(B)\theta_s(B_s)e_t \\
(1-\beta_1 B)(1-\beta_{s,1}B_s)\Delta_s\Delta Y_t &= (1+\theta_1 B)(1+\theta_{s,1}B_s)e_t \\
(1-\beta_1 B-\beta_{s,1}B_s+\beta_1\beta_{s,1}BB_s)\Delta_s\Delta Y_t &= (1+\theta_1 B+\theta_{s,1}B_s+\theta_1\theta_{s,1}BB_s)e_t
\end{aligned}$$

$$\Delta_s\Delta Y_t-\beta_1\Delta_s\Delta Y_{t-1}-\beta_{s,1}\Delta_s\Delta Y_{t-12}+\beta_1\beta_{s,1}\Delta_s\Delta Y_{t-13}=$$
$$e_t+\theta_1 e_{t-1}+\theta_{s,1}e_{t-12}+\theta_1\theta_{s,1}e_{t-13}$$
$$\Delta_s\Delta Y_t=\beta_1\Delta_s\Delta Y_{t-1}+\beta_{s,1}\Delta_s\Delta Y_{t-12}-\beta_1\beta_{s,1}\Delta_s\Delta Y_{t-13}+$$
$$e_t+\theta_1 e_{t-1}+\theta_{s,1}e_{t-12}+\theta_1\theta_{s,1}e_{t-13}$$

- Can further simplify the left hand side using $\Delta=(1-B)$ and $\Delta_s=(1-B_s)$

Seasonal ARIMA Model - Example 1

Simple pure seasonal AR model, or ARIMA(0,0,0)(1,0,0)$_s$,

$$\begin{aligned}
\beta_s(B_s)Y_t &= e_t \\
(1-\beta_{s,1}B_s)Y_t &= e_t \\
Y_t &= \beta_{s,1}Y_{t-s} + e_t
\end{aligned}$$

- ACF: spike at each multiple of s
- PACF: single spike at lag s

```r
# PURE SEASONAL AR MODEL - SARIMA(0,0,0)(1,0,0)[s=12]
set.seed(11)
data1<-arima.sim(list(order=c(12,0,0), ar=c(rep(0,11),0.6)), n=100)
par(mfrow=c(3,1))
plot(data1, xlab="Time", ylab="Y",
     main=expression(paste("ARIMA(0,0,0)(1,0,0)[s=12], ",beta[s],"=0.6")))

# ACF & PACF
acf(data1, main="")
pacf(data1, main="")
```

Seasonal ARIMA Model - Example 2

Simple pure seasonal MA model, or ARIMA(0,0,0)(0,0,1)$_s$,

$$\begin{aligned} Y_t &= \theta_s(B_s)e_t \\ Y_t &= (1+\theta_{s,1}B_s)e_t \\ Y_t &= e_t + \theta_{s,1}e_{t-s} \end{aligned}$$

- ACF: single spike at lag s
- PACF: spike at each multiple of s

```
# PURE SEASONAL MA MODEL - SARIMA(0,0,0)(0,0,1)[s=12]
set.seed(11)
data3<-arima.sim(list(order=c(0,0,12), ma=c(rep(0,11),0.6)), n=100)
par(mfrow=c(3,1))
plot(data3, xlab="Time", ylab="Y",
     main=expression(paste("ARIMA(0,0,0)(0,0,1)[s=12], ",theta[s],"=0.6")))

# ACF & PACF
acf(data3, main="")
pacf(data3, main="")
```

Seasonal ARIMA Model - Example 3

An AR model with seasonal and non-seasonal components, or ARIMA(1,0,0)(1,0,0)$_s$,

$$
\begin{aligned}
\beta(B)\beta_s(B_s)Y_t &= e_t \\
(1-\beta_1 B)(1-\beta_{s,1}B_s)Y_t &= e_t \\
(1-\beta_1 B-\beta_{s,1}B_s+\beta_1\beta_{s,1}BB_s)Y_t &= e_t \\
Y_t &= \beta Y_{t-1}+\beta_{s,1}Y_{t-s}-\beta_1\beta_{s,1}Y_{t-s-1}+e_t
\end{aligned}
$$

```r
# AR MODEL WITH SEASONAL AND NON-SEASONAL COMPONENTS - SARIMA(1,0,0)(1,0,0)[s=12]
set.seed(11)
data2<-arima.sim(list(order=c(12,0,0), ar=c(0.5,rep(0,10),0.4)), n=100)
par(mfrow=c(3,1))
plot(data2, xlab="Time", ylab="Y",
     main=expression(paste("ARIMA(1,0,0)(1,0,0)[s=12], ",beta,"=0.5, ",beta[s],"=0.4")))

# ACF & PACF
acf(data2, main="")
pacf(data2, main="")
```

Seasonal ARIMA Model - Example 4

An MA model with seasonal and non-seasonal components, or ARIMA(0,0,1)(0,0,1)$_s$,

$$
\begin{aligned}
Y_t &= \theta(B)\theta_s(B_s)e_t \\
Y_t &= (1+\theta_1 B)(1+\theta_{s,1}B_s)e_t \\
Y_t &= (1+\theta_1 B+\theta_{s,1}B_s+\theta_1\theta_{s,1}BB_s)e_t \\
Y_t &= e_t+\theta_1 Y_{t-1}+\theta_{s,1}e_{t-s}+\theta_1\theta_{s,1}e_{t-s-1}
\end{aligned}
$$

```
# MA MODEL WITH SEASONAL AND NON-SEASONAL COMPONENTS - SARIMA(0,0,1)(0,0,1)[s=12]
set.seed(11)
data4<-arima.sim(list(order=c(0,0,12), ma=c(0.5,rep(0,10),0.4)), n=100)
par(mfrow=c(3,1))
plot(data4, xlab="Time", ylab="Y",
     main=expression(paste("ARIMA(0,0,1)(0,0,1)[s=12], ",theta,"=0.5, ",theta[s],"=0.4")))

# ACF & PACF
acf(data4, main="")
pacf(data4, main="")
```

Seasonal ARIMA Model - Example 5

An AR model with (an additive) seasonal MA component, or ARIMA(1,0,0)(0,0,1)$_s$,

$$
\begin{aligned}
\beta(B)Y_t &= \theta_s(B_s)e_t \\
(1-\beta_1 B)Y_t &= (1+\theta_{s,1}B_S)e_t \\
Y_t &= \beta_1 Y_{t-1} + e_t + \theta_{s,1}e_{t-s}
\end{aligned}
$$

```r
# AR MODEL WITH SEASONAL MA COMPONENTS - SARIMA(1,0,0)(0,0,1)[s=12]
set.seed(11)
data5<-arima.sim(list(order=c(1,0,12), ar=0.6, ma=c(rep(0,11),0.5)), n=100)
par(mfrow=c(3,1))
plot(data5, xlab="Time", ylab="Y",
     main=expression(paste("ARIMA(1,0,0)(0,0,1)[s=12], ",beta,"=0.6, ",theta[s],"=0.5")))

# ACF & PACF
acf(data5, main="")
pacf(data5, main="")
```

Seasonal ARIMA Model

Seasonal differencing

- Seasonality usually causes the time series to be nonstationary
- A common approach is to transform data using a *logarithm* and apply regular differencing (i.e., I(1))

$$\Delta \log(Y_t) = (1 - B) \log(Y_t) = \log(Y_t) - \log(Y_{t-1})$$

  - This will remove an overall trend that is present in the data
  - It is important to note that non-seasonal behavior will still matter in seasonal models
- For time series that is both nonstationarity and shows seasonal pattern, the approach is to transform data using a logarithm and apply both regular and seasonal differencing

$$\Delta_s \Delta \log(Y_t) = (1 - B_s)(1 - B) \log(Y_t)$$

- Seasonal differencing is defined as a difference between a value and a value with lag that is a multiple of s

Seasonal ARIMA Model

Seasonal differencing (con't)

- Occasionally data has to be differenced more than once by applying

$$\Delta^d = (1 - B)^d \quad \text{or} \quad \Delta_s^D = (1 - B_s)^D$$

- Seasonal differencing removes a seasonal trend and can also get rid of a seasonal random walk (another type of nonstationarity)
- In practice ARIMA(1,1,0)(0,1,1)$_s$ and ARIMA(0,1,1)(0,1,1)$_s$ occur routinely

Seasonal ARIMA Model

Step-by-step approach:

1. Do a time series plot of the data and examine it for global trends and seasonality
2. Do any necessary differencing
   - If there is no obvious trend or seasonality, do not take any differences
   - If there is seasonality and no trend take a difference of lag s
   - If there is a linear trend and no obvious seasonality, take a first difference
   - If there is a curved trend, consider a transformation of the data before differencing
   - If there is both trend and seasonality, apply both a non-seasonal and seasonal difference to the data, as two successive operations
     - Example:

```
diff1<-diff(Y, lag=1, differences=1)                #non-seasonal differencing
diff1_and_12<-diff(diff1, lag=12, differences=1)    #seasonal (monthly) differencing
```

Seasonal ARIMA Model

Step-by-step approach (con't):

3. Examine the ACF and PACF of the differenced data and make some basic guesses about the most appropriate model at this time
    - *Non-seasonal terms*: Examine the early lags (1, 2, 3,...) to judge non-seasonal terms
        - Spikes in the ACF (at low lags) indicate non-seasonal MA terms
        - Spikes in the PAC (at low lags) indicated possible non-seasonal AR terms
    - *Seasonal terms*: Examine the patterns across lags that are multiples of s
        - Judge the ACF and PACF at the seasonal lags in the same way you do for the earlier lags
        - Example: for monthly data, look at lags 12, 24, 36 (usually the first two or three seasonal multiples)
4. Estimate the model(s) that might be reasonable for the data based on the previous steps
5. Examine the residuals (with ACF, Q-statistics, etc.) to see if the model seems good and compare AIC or BIC values to determine the best of several models

Estimation

- This is the second step of the Box-Jenkins methodology
  - Step 2 (Estimation) and step 3 (Diagnostics) usually go together in the analysis (iterative process)
- Three main methods to estimate the parameters of an ARIMA model:
  1. Method of moments (MOM)
  2. Maximum likelihood (MLE)
  3. Yule-Walker procedure (Y-W)
- The choice of the estimation method depends on the assumptions imposed on the error term
- The model is often estimated using either conditional likelihood method or exact likelihood method

- See Greene (2011, Chapters 20.5-20.6)

Estimation - MOM

- Let $Y_1, \ldots, Y_T$ be a sample from a population with pdf/pmf $f(y; \boldsymbol{\theta})$, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)' \in \Theta$
- The MOM estimators of $\theta_1, \ldots, \theta_p$ are obtained by solving p equations where the first p sample and the population moments are matched
- Specifically, for $r \geq 1$,

$$\begin{array}{ll} \mu_r' \equiv \mu_r(\boldsymbol{\theta}) = \mathrm{E}(Y^r) & \text{the } r\text{th population moment} \\ m_r = \frac{1}{T}\sum_{i=1}^{T} y_i^r & \text{the } r\text{th sample moment} \end{array}$$

- Then, the MOM estimators of $(\theta_1, \ldots, \theta_p)$ are recovered by solving the p equations

$$\begin{aligned} \mu_1(\boldsymbol{\theta}) &= m_1 \\ \mu_2(\boldsymbol{\theta}) &= m_2 \\ &\vdots \\ \mu_p(\boldsymbol{\theta}) &= m_p \end{aligned}$$

Estimation - MOM

- Properties of MOM estimators:
  - Easy to implement
  - Computationally simple
  - Converges to the parameter with increasing probability (i.e., consistency)
  - MOM estimator is **not** necessarily asymptotically most efficient

Estimation - MOM - Example

- Consider a linear time series model

$$\underset{T\times 1}{\mathbf{Y}} = \underset{T\times k}{\mathbf{X}}\underset{k\times 1}{\beta} + \underset{T\times 1}{\varepsilon}$$

  where $\mathbf{X}$ contains k different components of the model
- Classical assumptions

$$\begin{aligned} \mathrm{E}(\varepsilon) &= \mathbf{0}_T \\ \mathrm{Var}(\varepsilon|\mathbf{X}) &= \boldsymbol{\sigma}^2\mathbf{I}_T \\ \mathrm{Cov}(\varepsilon, \mathbf{X}) &= \mathrm{E}(\underset{T\times k}{\mathbf{X}}{}' \underset{T\times 1}{\varepsilon}) = \mathbf{0}_k \end{aligned}$$

- Each of these equations are moment equations
- Use the last equation to construct population and sample moment conditions, and obtain MOM estimates of model parameters
  - Last equation implies that each of k regressors ($\mathbf{X}$'s) are exogenous from ε (strict exogeneity)
  - Hence we can obtain k equations in k unknown parameters

Estimation - MOM - Example

- Population moment

$$
\begin{aligned}
\mathrm{E}(\underset{T\times k}{\mathbf{X}}{}'\underset{T\times 1}{\varepsilon}) &= \mathbf{0}_k \\
\mathrm{E}(\underset{T\times k}{\mathbf{X}}{}'(\underset{T\times 1}{\mathbf{Y}} - \underset{T\times k}{\mathbf{X}}\underset{k\times 1}{\beta})) &= \mathbf{0}_k
\end{aligned}
$$

- Sample moment

$$
\frac{1}{T}\sum_{t=1}^{T}\mathbf{X}_{t.}'(Y_t - \mathbf{X}_{t.}\beta)
$$

- By the law of large numbers

$$
\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\mathbf{X}_{t.}'(Y_t - \mathbf{X}_{t.}\beta) &\approx \mathrm{E}(\underset{T\times k}{\mathbf{X}}{}'(\underset{T\times 1}{\mathbf{Y}} - \underset{T\times k}{\mathbf{X}}\underset{k\times 1}{\beta})) = \mathbf{0}_k \\
T^{-1}[\mathbf{X}'(\mathbf{Y} - \mathbf{X}\beta)] &= \mathbf{0}_k
\end{aligned}
$$

- The MOM estimate is

$$
\begin{aligned}
\hat{\beta}_{\mathrm{MOM}} &= \arg_{\beta}\{T^{-1}[\mathbf{X}'(\mathbf{Y} - \mathbf{X}\beta)] = \mathbf{0}_k\} \\
&= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}
\end{aligned}
$$

Estimation - MLE

- The MLE estimators $\hat{\theta}_1, \ldots, \hat{\theta}_p$ of $\theta_1, \ldots, \theta_p$ are the location at which the likelihood function, $L(\theta_1, \ldots, \theta_p | \mathbf{Y})$, attains its global maximum as a function of $\theta_1, \ldots, \theta_p$
- The likelihood function is

$$L(\theta_1, \ldots, \theta_p | \mathbf{Y}) = \prod_{i=1}^{n} f(\mathbf{Y} | \theta_1, \ldots, \theta_p)$$

- If $L(\theta_1, \ldots, \theta_p | \mathbf{Y})$ is differentiable in $\theta_1, \ldots, \theta_p$, we solve

$$\frac{\partial}{\partial \theta_j} L(\theta_1, \ldots, \theta_p | \mathbf{Y}) = 0, \quad j = 1, \ldots, p$$

- The solutions to these likelihood equations locate only extreme points in the interior of Θ and provide possible candidates for the MLE
- They can be local or global minima, local or global maxima, or inflection points

Estimation - MLE

- $\frac{d^2}{d\theta_j^2} L(\theta_1, \ldots, \theta_p | \mathbf{Y})|_{\theta=\hat{\theta}} < 0$ is sufficient for local maxima
- If there is only one local maxima, then that must be the unique global maxima
- Many examples falls in this category, so no further work will be needed then

Theorem (Invariance Principle)

If $\hat{\theta}$ is the MLE of θ, then for any function $\tau(\theta)$, the MLE of $\tau(\theta)$ is $\tau(\hat{\theta})$

Forecasting

- Forecasting/prediction is an important application of time series analysis
- Suppose that we are at the time index t and are interested in forecasting $y_{t+\ell}$, where $\ell \geq 1$
  - The time index t is called the **forecast origin**
  - The positive integer ℓ is the **forecast horizon**
- The forecast $\hat{y}_t(\ell)$ of $y_{t+\ell}$ is chosen to minimize the squared error loss function

$$\mathrm{E}[(y_{t+\ell} - \hat{y}_t(\ell))^2 | F_t] \leq \min_{g} \mathrm{E}[(y_{t+\ell} - g)^2 | F_t]$$

  - $F_t = \{y_t, y_{t-1}, y_{t-2}, \dots\}$ is a collection of information available at the forecast origin t
  - g is some function of F_t (an alternative estimator)
- $\hat{y}_t(\ell)$ is known as the ℓ-**step ahead forecast** of series Y_t
  - $\hat{y}_t(1)$ is the forecast of y_{t+1} (1-step ahead from y_t)
  - $\hat{y}_t(2)$ is the forecast of y_{t+2} (2-step ahead from y_t)

Forecasting - AR(p) Process - 1-Step-Ahead Forecast

1-Step-Ahead Forecast ($\ell = 1$)

- Suppose that we are at the time index t and are interested in forecasting y_{t+1}
- From AR(p) model, the equation for y_{t+1} is

$$y_{t+1} = \beta_0 + \beta_1 y_t + \beta_2 y_{t-1} + \cdots + \beta_p y_{t-p+1} + e_{t+1}$$

- Under the minimum squared error loss function, the point forecast of y_{t+1} given F_t (the information available at time t) is the conditional expectation

$$\hat{y}_t(1) = \mathrm{E}(y_{t+1}|F_t) = \beta_0 + \beta_1 y_t + \beta_2 y_{t-1} + \beta_3 y_{t-2} + \cdots + \beta_p y_{t-p+1}$$

with the associated forecast error of

$$\hat{e}_t(1) = y_{t+1} - \hat{y}_t(1) = e_{t+1}$$

and

$$\mathrm{Var}(\hat{e}_t(1)) = \mathrm{Var}(e_{t+1}) = \sigma^2$$

Forecasting - AR(p) Process - 2-Step-Ahead Forecast

2-Step-Ahead Forecast ($\ell = 2$)

- Suppose that we are at the time index t and are interested in forecasting y_{t+2}
- From AR(p) model, the equation for y_{t+2} is

$$y_{t+2} = \beta_0 + \beta_1 y_{t+1} + \beta_2 y_t + \beta_3 y_{t-1} + \cdots + \beta_p y_{t-p+2} + e_{t+2}$$

- Under the minimum squared error loss function, the point forecast of y_{t+2} given F_t (the information available at time t) is the conditional expectation

$$\hat{y}_t(2) = \mathrm{E}(y_{t+2}|F_t) = \beta_0 + \beta_1 \hat{y}_t(1) + \beta_2 y_t + \beta_3 y_{t-1} + \cdots + \beta_p y_{t-p+2}$$

with the associated forecast error of

$$\hat{e}_t(2) = y_{t+2} - \hat{y}_t(2) = \beta_1 \underbrace{[y_{t+1} - \hat{y}_t(1)]}_{e_{t+1}} + e_{t+2}$$

and

$$\mathrm{Var}(\hat{e}_t(2)) = \mathrm{Var}\{\beta_1[y_{t+1} - \hat{y}_t(1)] + e_{t+2}\} = (1 + \beta_1^2)\sigma^2$$

Forecasting - AR(p) Process - 2-Step-Ahead Forecast

- Note that

$$\begin{aligned} \text{Var}(\hat{e}_t(2)) &\geq \text{Var}(\hat{e}_t(1)) \\ (1+\beta_1^2)\sigma^2 &\geq \sigma^2 \end{aligned}$$

meaning that as the forecast horizon increases ($\ell \uparrow$), the uncertainty in forecast also increases

Forecasting - MA(1) - 1-Step-Ahead Forecast

1-Step-Ahead Forecast ($\ell = 1$)

- Suppose that we are at the time index t and are interested in forecasting y_{t+1}
- From MA(1) model, the equation for y_{t+1} is ▸ Details

$$y_{t+1} = \beta_0 + e_{t+1} + \theta e_t$$

- Under the minimum squared error loss function, the point forecast of y_{t+1} given F_t (the information available at time t) is the conditional expectation

$$\hat{y}_t(1) = \mathrm{E}(y_{t+1}|F_t) = \beta_0 + \theta e_t$$

with the associated forecast error of

$$\hat{e}_t(1) = y_{t+1} - \hat{y}_t(1) = e_{t+1}$$

and

$$\mathrm{Var}(\hat{e}_t(1)) = \mathrm{Var}(e_{t+1}) = \sigma^2$$

Forecasting - MA(1) - 2-Step-Ahead Forecast

2-Step-Ahead Forecast ($\ell = 2$)

- Suppose that we are at the time index t and are interested in forecasting y_{t+1}
- From MA(1) model, the equation for y_{t+2} is

$$y_{t+2} = \beta_0 + e_{t+2} + \theta e_{t+1}$$

- Under the minimum squared error loss function, the point forecast of y_{t+2} given F_t (the information available at time t) is the conditional expectation

$$\hat{y}_t(2) = \mathrm{E}(y_{t+2}|F_t) = \beta_0$$

with the associated forecast error of

$$\hat{e}_t(2) = y_{t+2} - \hat{y}_t(2) = e_{t+2} + \theta e_{t+1}$$

and

$$\mathrm{Var}(\hat{e}_t(2)) = \mathrm{Var}\{e_{t+2} + \theta e_{t+1}\} = (1 + \theta_1^2)\sigma^2$$

Forecasting - MA(1) - 2-Step-Ahead Forecast

- Note that

$$\begin{aligned} \mathsf{Var}(\hat{e}_t(2)) &\geq \mathsf{Var}(\hat{e}_t(1)) \\ (1+\theta_1^2)\sigma^2 &\geq \sigma^2 \end{aligned}$$

meaning that as the forecast horizon increases ($\ell \uparrow$) the uncertainty in forecast also increases

Diagnostics

- This is the third step of the Box-Jenkins methodology

1. Identify the best model
   - The model with the lowest value of **Akaike Information Criterion** (AIC) and **Schwarz-Bayesian Information Criterion** (BIC) can be chosen as the best model

$$\begin{aligned} \text{AIC} &= -\frac{2}{T}\log(L) + \frac{2}{T}k \\ \text{BIC} &= -\frac{2}{T}\log(L) + \frac{\log(T)}{T}k \end{aligned}$$

   where T is the sample size, L is the value of the likelihood function, and k is the number of parameters in the model

Diagnostics

2. Examine the residuals of the fitted (best) model to confirm the goodness of fit
    - If the model is well specified, residuals should be *very close* to white-noise
    - Plot residuals, look for outliers, periods in which the model does not fit the data well (evidence of structural change)
    - Examine ACF and PACF of the residuals to check for significant autocorrelations
    - If most of the sample autocorrelation coefficients of the residuals lies within the confidence limit, then the residuals are the white noise indicating that the model fit is appropriate
    - Use Q-statistics to determine if groups of autocorrelations are statistically significant
3. Estimated coefficients should be consistent with the underlying assumption of stationarity

Diagnostics - Q-Statistics

Portmanteau Test: To formally test

$$
\begin{aligned}
&H_0 : \rho(1) = \rho(2) = \cdots = \rho(H) = 0 \\
&H_1 : \rho(h) \neq 0 \quad \text{for some } h = \{1, 2, \ldots, H\}
\end{aligned}
$$

- **Box-Pierce test statistic** (Box and Pierce, 1970, *JASA*)

$$
Q^*(H) = T \sum_{h=1}^{H} \hat{\rho}^2(h)
$$

- **Ljung-Box test statistic** (Ljung and Box, 1978, *Biometrika*)

$$
Q(H) = T(T+2) \sum_{h=1}^{H} \frac{\hat{\rho}^2(h)}{T-h} \sim \chi^2_{H-k}
$$

  where k is the number of estimated model parameters

- Ljung-Box statistic tends to perform better in smaller samples
- In general, the recommendation is to use $H \approx \log(T)$

Diagnostics - In-Sample Accuracy

- Split the series into two subsamples:
  1. *Estimation sample*: $y_1, y_2, \dots, y_t$
  2. *Prediction sample*: $y_{t+1}, y_{t+2}, \dots, y_T$

- Use the first subsample to estimate the model
- Evaluate **in-sample accuracy** by comparing fitted values $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_t$ with actual values $y_1, y_2, \dots, y_t$
- Given the fitted values $\hat{y}_j$, for $j = 1, \dots, t$ from the model, in-sample residuals are

$$\hat{e}_j = y_j - \hat{y}_j$$

Diagnostics - In-Sample Accuracy

- *Mean Error* - measure of the average bias

$$\text{ME} = \frac{1}{t}\sum_{j=1}^{t} \hat{e}_j$$

- *(Residual) Mean Squared Error* - sample average loss for quadratic loss function

$$\text{MSE} = \frac{1}{t}\sum_{j=1}^{t} \hat{e}_j^2$$

- *Mean Absolute Error* - sample average loss for absolute value loss function

$$\text{MAE} = \frac{1}{t}\sum_{j=1}^{t} |\hat{e}_j|$$

- *Mean Absolute Percentage Error*

$$\text{MAPE} = \frac{1}{t}\sum_{j=1}^{t} \left|\frac{\hat{e}_j}{y_j}\right|$$

- *Mean Absolute Scaled Error* - compares in sample MAE of the model forecast with in sample MAE for one-step naive forecast method $\hat{y}_{j+1} = y_j$

$$\text{MASE} = \frac{\frac{1}{t}\sum_{j=1}^{t} |\hat{e}_j|}{\frac{1}{t-1}\sum_{j=1}^{t-1} |\hat{y}_{j+1} - y_j|}$$

Diagnostics - Out-of-Sample Accuracy

- Use the second subsample to construct set of ℓ-step ahead forecasts $\hat{y}_t(\ell), \hat{y}_{t+1}(\ell), \ldots, \hat{y}_{T-\ell}(\ell)$
- Evaluate **out-of-sample accuracy** by comparing forecasts $\hat{y}_t(\ell), \hat{y}_{t+1}(\ell), \ldots, \hat{y}_{T-\ell}(\ell)$ with actual values $y_{t+\ell}, y_{t+1+\ell}, \ldots, y_T$
- Given the forecast values $\hat{y}_{t+j}(\ell)$, for $j = 0, 1, \ldots, T - t - \ell$ from the model, out-of-sample residuals are

$$\hat{e}_{t+j}(\ell) = y_{t+j+\ell} - \hat{y}_{t+j}(\ell)$$

Diagnostics - Out-of-Sample Accuracy

- *Mean Error* - measure of the average bias

$$\text{ME} = \frac{1}{T-t-\ell+1} \sum_{j=0}^{T-t-\ell} e_{t+j}(\ell)$$

- *(Residual) Mean Squared Error* - sample average loss for quadratic loss function

$$\text{MSE} = \frac{1}{T-t-\ell+1} \sum_{j=0}^{T-t-\ell} e_{t+j}(\ell)^2$$

- *Mean Absolute Error* - sample average loss for absolute value loss function

$$\text{MAE} = \frac{1}{T-t-\ell+1} \sum_{j=0}^{T-t-\ell} |e_{t+j}(\ell)|$$

- *Mean Absolute Percentage Error*

$$\text{MAPE} = \frac{1}{T-t-\ell+1} \sum_{j=0}^{T-t-\ell} \left| \frac{e_{t+j}(\ell)}{y_{t+j+\ell}} \right|$$

- *Mean Absolute Scaled Error*

$$\text{MASE} = \frac{\frac{1}{T-t-\ell+1} \sum_{j=0}^{T-t-\ell} |e_{t+j}(\ell)|}{\frac{1}{T-t-\ell} \sum_{j=0}^{T-t-\ell-1} |\hat{y}_{t+j+1}(\ell) - y_{t+j+\ell}|}$$

Step-by-Step Analysis

Step-by-step approach:

1. Plot the data and check the ACF/PACF
2. Run stationarity test (Dickey-Fuller and/or Augmented Dickey-Fuller test)
3. To get stationarity, transform the data (e.g., log())
4. If the ACF does not decay fast enough, take successive (non-seasonal and/or seasonal) differences of the log-values
5. This should look more like a stationary time series
6. Fit suitable candidate models to the data
7. Choose the model with the mimimum value of the AIC or BIC values
8. Run diagnostic checks on the residuals
9. Forecast

Implementation in R

`ar()`: (Build-in command in R)

- Fits an auto-regressive a univariate or multivariate time series model to the data
- The order of the model is automatically determined based on AIC
- `method = c("yule-walker", "burg", "ols", "mle", "yw")`

`ar.ols()`: (Build-in command in R)

- Fit an auto-regressive a univariate or multivariate time series model to the data by OLS

`arima()`: (Build-in command in R)

- Fits an ARIMA model by MLE to a univariate time series
- `order = c(ar, i, ma)`
- `seasonal = list(order = c(ar, i, ma), period=NA)`
- `include.mean = TRUE`: includes intercept

Implementation in R

`predict()`: (Build-in command in R)

- Forecasts from time series or time series models

forecast package:

- `auto.arima()`: Identifies best ARIMA model to univariate time series based on either AIC, AICc or BIC
- `forecast()`: Forecasts from time series or time series models
- `accuracy()`: Returns in-sample and out-of-sample measures of forecast accuracy
- `forecast.HoltWinters()`: Forecasts for univariate Holt-Winters time series models

Estimation - Example 1

```r
# EXAMPLE 1 - SIMULATED AR(3)
set.seed(11)
data<-arima.sim(model=list(order=c(3,0,0), ar=c(-0.2,0.1,-0.4)), n=500)

# MLE - DETERMING OPTIMAL AR ORDER
output<-ar(data, method="mle")
print(output)

# ACF & PACF OF THE SERIES
par(mfrow=c(3,2))
acf(data, main="Data")
pacf(data, main="Data")

# ACF & PACF OF THE RESIDUALS
acf(output$resid[4:500], main="Residuals")
pacf(output$resid[4:500], main="Residuals")

# NAIVE OLS METHOD
output_n<-lm(data~time(data))
summary(output_n)

# ACF & PACF OF THE RESIDUALS
acf(output_n$resid, main="Naive Residuals")
pacf(output_n$resid, main="Naive Residuals")

# OLS - LAG ORDER 3
output_ols<-ar.ols(data, order=3, demean=FALSE, intercept=FALSE)
print(output_ols)
```

Estimation - Example 2

```r
# EXAMPLE 2 - BOX & JENKINDS MONTHLY AIRLINE DATA (1949-1960)
data(AirPassengers)
AP<-AirPassengers

# ACF & PACF OF THE SERIES
par(mfrow=c(2,2))
acf(AP, main="Data")
pacf(AP, main="Data")

# MLE - MA(3)
AP.ma<-arima(AP, order=c(0,0,3))    #include.mean = FALSE: removes intercept
print(AP.ma)

# MLE - ARMA(1,1)
AP.arma<-arima(AP, order=c(1,0,1))
print(AP.arma)

# MLE - ARIMA(1,1,1)
AP.arima<-arima(AP, order=c(1,1,1))
print(AP.arima)

# MLE - ARIMA(1,1,1)(0,1,0)[s]
AP.sarima<-arima(AP, order=c(1,1,1), seasonal=list(order=c(0,1,0), period=12))
print(AP.sarima)

# ACF & PACF OF THE RESIDUALS FROM ARIMA(1,1,1)(0,1,0)[s]
acf(AP.sarima$residuals, main="Residuals, ARIMA(1,1,1)-seasonal")
pacf(AP.sarima$residuals, main="Residuals, ARIMA(1,1,1)-seasonal")
```

Estimation - Example 2

```
# PREDICTION - ARIMA(1,1,1)(0,1,0)[s] - USING PREDICT() COMMAND
AP.predict<-predict(AP.sarima, n.ahead=10*12)   #n.ahead=year*month
print(AP.predict)

# PLOT THE SERIES
ts.plot(AP,AP.predict$pred, ylim=c(-200,1200), col=1:2,
        xlab="Time",
        ylab="Observed/Predicted",
        main="Monthly international airline passengers")
lines(AP.predict$pred+2*AP.predict$se, col=4, lty=2)
lines(AP.predict$pred-2*AP.predict$se, col=4, lty=2)
legend("topleft", c("data","prediction", "CI"),
       col=c(1,2,4), lty=c(1,1,2), bty="n")

# PREDICTION - ARIMA(1,1,1)(0,1,0)[s] - USING FORECAST() COMMAND
install.packages("forecast")
library(forecast)
AP.predict<-forecast(AP.sarima, h=10*12)   #h=year*month
print(AP.predict)

# PLOT THE SERIES
plot(AP.predict,          #include=10: includes 10 obs from the original series
     ylim=c(-200,1200),
     xlab="Time",
     ylab="Observed/Predicted",
     main="Monthly international airline passengers")
legend("topleft", c("data","prediction","95% CI", "80% CI"),
       col=c(1,4, "gray", "lightsteelblue3"), lty=1, lwd=2, bty="n")
```

Estimation - Example 3

```
# EXAMPLE 3 - JOHNSON & JOHNSON QUARTERLY EARNINGS PER SHARE (1960-1980)
library(astsa)
yall<-jj

# SPLIT THE SERIES INTO TWO PARTS: (i) ESTIMATION SAMPLE (ii) PREDICTION SAMPLE
y1<-window(yall, end=c(1978,4))
y2<-window(yall, start=c(1979,1))
y<-y1

# TRANSFORM & DIFFERENCE THE SERIES
ly<-log(y)                                          #log
dly1<-diff(ly, differences=1, lag=1)                #log-differenced (non-seasonal)
dly4<-diff(ly, differences=1, lag=4)                #log-differenced (seasonal)
dly4_1<-diff(diff(ly), differences=1, lag=4)  #log-differenced (non-seasonal+seasonal)

# PLOT THE SERIES
par(mfrow=c(2,3))
plot(y, main=expression(y))
plot(ly, main=expression(log(y)))
plot.new()
plot(dly1, main=expression(paste(Delta, "log(y)")))
plot(dly4, main=expression(paste(Delta[4], "log(y)")))
plot(dly4_1, main=expression(paste(Delta, Delta[4], "log(y)")))

# STATIONARITY TEST
library(tseries)
adf.test(y, alternative="stationary", k=0)
adf.test(y, alternative="stationary")
```

Estimation - Example 3

```
adf.test(ly, alternative="stationary", k=0)
adf.test(ly, alternative="stationary")
adf.test(dly1, alternative="stationary", k=0)
adf.test(dly1, alternative="stationary")
adf.test(dly4, alternative="stationary", k=0)
adf.test(dly4, alternative="stationary")
adf.test(dly4_1, alternative="stationary", k=0)
adf.test(dly4_1, alternative="stationary")

# ACF & PACF
library(zoo)        #for coredata command
maxlag<-24          #=4quarters*6years
par(mfrow=c(2,4))
plot(acf(coredata(ly), type='correlation', lag=maxlag, plot=FALSE), ylab="",
     main=expression(paste("ACF for log(y)")))
plot(acf(coredata(dly1), type='correlation', lag=maxlag, plot=FALSE), ylab="",
     main=expression(paste("ACF for ", Delta,"log(y)")))
plot(acf(coredata(dly4), type='correlation', lag=maxlag, plot=FALSE), ylab="",
     main=expression(paste("ACF for ", Delta[4], "log(y)")))
plot(acf(coredata(dly4_1), type='correlation', lag=maxlag, plot=FALSE), ylab="",
     main=expression(paste("ACF for ", Delta[4], Delta, "log(y)")))
acf(coredata(ly), type='partial', lag=maxlag, ylab="",
    main=expression(paste("PACF for log(y)")))
acf(coredata(dly1), type='partial', lag=maxlag, ylab="",
    main=expression(paste("PACF for ", Delta, "log(y)")))
acf(coredata(dly4), type='partial', lag=maxlag, ylab="",
    main=expression(paste("PACF for ", Delta[4], "log(y)")))
acf(coredata(dly4_1), type='partial', lag=maxlag, ylab="",
    main=expression(paste("PACF for ", Delta[4], Delta, "log(y)")))
```

Estimation - Example 3

```r
# AUTO SEARCH FOR THE BEST MODEL (TRY DIFFERENT CRITERIA)
library(forecast)
auto1<-auto.arima(ly, ic="bic", seasonal=TRUE,
                  stationary=FALSE, stepwise=FALSE, trace=TRUE)
summary(auto1)
auto2<-auto.arima(dly1, ic="bic", seasonal=TRUE,
                  stationary=TRUE, stepwise=FALSE, trace=TRUE)
summary(auto2)    #auto1 and auto2 are the same model/coefficients
auto3<-auto.arima(dly4, ic="bic", seasonal=TRUE,
                  stationary=FALSE, stepwise=FALSE, trace=TRUE)
summary(auto3)
auto4<-auto.arima(dly4_1, ic="bic", seasonal=TRUE,
                  stationary=TRUE, stepwise=FALSE, trace=TRUE)
summary(auto4)

# MODEL 1 - GOOD CANDIDATE
m1<-arima(ly, order=c(0,1,1), seasonal=list(order=c(2,0,0), period=4))
print(m1)

# Z-STATISTICS
z1<-m1$coef/sqrt(diag(m1$var.coef))
print(z1)

# P-VALUES
p1<-2*(1-pnorm(abs(m1$coef)/sqrt(diag(m1$var.coef))))
print(p1)

# 95% CONFIDENCE INTERVAL
confint(m1, level=0.95)
```

Estimation - Example 3

```
# DIAGNOSTICS

# IN-SAMPLE ACCURACY
bic1<-BIC(m1)
aic1<-AIC(m1)
is<-accuracy(m1)                #if command fails, re-install forecast

# Q-STATISTICS - AUTOCORRELATION TEST
tsdiag(m1, gof.lag=36)
LB1<-Box.test(m1$residuals, lag=36, type="Ljung")
print(LB1)

# NORMALITY TEST
qqnorm(m1$residuals)
qqline(m1$residuals)
shapiro.test(m1$residuals)      #Shapiro-Wilk test: H0: Normality
library(tseries)
jarque.bera.test(m1$residuals) #Jarque-Bera test: H0: Normality
install.packages("nortest")
library(nortest)
ad.test(m1$residuals)           #Anderson-Darling test: H0: Normality
cvm.test(m1$residuals)          #Cramer-von Mises test: H0: Normality
lillie.test(m1$residuals)       #Kolmogorov-Smirnov test: H0: Normality

# TEST FOR CONDITIONAL HETEROSCEDASTICTY (ARCH)
install.packages("TSA")
library(TSA)
McLeod.Li.test(m1,ly)
```

Estimation - Example 3

```
# PREDICTION
m1.fcast<-forecast(m1, h=8)        #=4(quarters)*2(years)
plot(m1.fcast, xlim=c(1970,1981))
lines(log(yall))

# OUT-OF-SAMPLE ACCURACE
os<-accuracy(m1.fcast, y2)         #if command fails, re-install forecast

# MODEL 2
m2<-arima(dly1, order=c(0,0,1), seasonal=list(order=c(2,0,0), period=4))
print(m2)

# DIAGNOSTICS
bic2<-BIC(m2)
aic2<-AIC(m2)
tsdiag(m2, gof.lag=36)
LB2<-Box.test(m2$residuals, lag=36, type="Ljung")
print(LB2)

# PREDICTION
m2.fcast<-forecast(m2, h=8)
plot(m2.fcast, xlim=c(1970,1981))
lines(diff(log(yall), differences=1, lag=1))

# MODEL 3
m3<-arima(dly4, order=c(1,0,0))
print(m3)
```

Estimation - Example 3

```
# DIAGNOSTICS
bic3<-BIC(m3)
aic3<-AIC(m3)
tsdiag(m3, gof.lag=36)
LB3<-Box.test(m3$residuals, lag=36, type="Ljung")
print(LB3)

# PREDICTION
m3.fcast<-forecast(m3, h=8)
plot(m3.fcast, xlim=c(1970,1981))
lines(diff(log(yall), differences=1, lag=4))

# MODEL 4
m4<-arima(dly4_1, order=c(0,0,1), seasonal=list(order=c(1,0,0), period=4))
print(m4)

# DIAGNOSTICS
bic4<-BIC(m4)
aic4<-AIC(m4)
tsdiag(m4, gof.lag=36)
LB4<-Box.test(m4$residuals, lag=36, type="Ljung")
print(LB4)

# PREDICTION
m4.fcast<-forecast(m4, h=8)
plot(m4.fcast, xlim=c(1970,1981))
lines(diff(diff(log(yall), differences=1, lag=1), differences=1, lag=4))
```

Exponential Smoothing

- Suppose the available data are $y_t, y_{t-1}, y_{t-2}, \ldots$ and we are interested in predicting y_{t+1}
- A natural estimate for predicting the next value of a given time series Y_t at the period t is to take weighted sums of past observations
- The data y_t should be more relevant than y_{t-1} in predicting y_{t+1}; y_{t-1} is more relevant than y_{t-2}; etc.
- A simple formulation of the 1-step ahead prediction of y_{t+1} is

$$\hat{y}_{t+1} = y_t(1) = \lambda_0 \cdot y_t + \lambda_1 \cdot y_{t-1} + \lambda_2 \cdot y_{t-2} + \ldots$$

  - $\lambda_i = \alpha(1-\alpha)^i$ is geometric weights, for $i = 0, 1, 2, \ldots$
  - $\alpha \in (0, 1)$ is the smoothing parameter
  - The weights (λ_i's) must sum to 1

Exponential Smoothing

Simple Exponential Smoothing

The simple exponential smoothing is

$$\begin{aligned} Y_t(1) &= \alpha \cdot Y_t + \alpha(1-\alpha) \cdot Y_{t-1} + \alpha(1-\alpha)^2 \cdot Y_{t-2} + \dots \\ &= \alpha \cdot Y_t + (1-\alpha) \cdot Y_{t-1}(1) \end{aligned}$$

- If α is close to zero, more weight is placed on past observations
- If α is close to 1, more weight is placed on recent observations

- The term “exponential” comes from the fact that the weights decay exponentially
- The method is also known as the **exponentially weighted moving average** (EWMA) method
- Unknown parameters can be determined by minimizing the squared prediction error

Exponential Smoothing

- Assume that e_{t+1} is the forecast error, then we have

$$\begin{aligned} y_{t+1} &= y_t(1) + e_{t+1} \\ &= \alpha \cdot y_t + (1-\alpha) \cdot y_{t-1}(1) + e_{t+1} \\ &= y_t - (1-\alpha)[y_t - y_{t-1}(1)] + e_{t+1} \\ &= y_t - (1-\alpha)e_t + e_{t+1} \end{aligned}$$

  which can be written as

$$\begin{aligned} y_{t+1} - y_t &= e_{t+1} - (1-\alpha)e_t \\ (1-B)y_{t+1} &= (1-(1-\alpha)B)e_{t+1} \end{aligned}$$

  which is ARIMA(0,1,1) process
- **Limitation**: Exponential smoothing in its basic form should only be used for time series with no systematic trend and/or seasonal components

Exponential Smoothing

- **Holt-Winters Filtering**
  - Holt (1957) *Forecasting Trends and Seasonals by Exponentially Weighted Moving Averages*
  - Winters (1960) *Management Science*
- It is the generalization of the simple exponential smoothing that deals with time series containing trend and seasonal variation
- Smoothing is controlled by three parameters:
  - α (for the level)
  - β (for the trend)
  - γ (for the seasonal variation)

Implementation in R

- `HoltWinters()` (contained in `ts` library)
  - Computes Holt-Winters Filtering of a given time series
  - Smoothing parameters are determined automatically by minimizing the squared prediction error from one–step forecasts
  - `seasonal="additive"` (by default): Uses additive seasonal model
  - Can change this to `"multiplicative"`

Exponential Smoothing - Example

```
# HOLT-WINTERS FILTERING
# BOX & JENKINS MONTHLY AIRLINE DATA (1949-1960)
data(AirPassengers)
AP<-AirPassengers
par(mfrow=c(3,1))

AP.hw<-HoltWinters(AP)
str(AP.hw)
print(AP.hw$fitted)
print(c(AP.hw$alpha, AP.hw$beta, AP.hw$gamma))
plot(AP.hw$fitted, main="H-W Time Series Components")

# PREDICTION
plot(AP.hw,col=1:2)
legend("topleft", c("data","fitted"), col=c(1,2), lty=1, bty="n")
#plot(AP.hw$x,col=1)
#lines(AP.hw$fitted[,1],col=2)

AP.predict<-predict(AP.hw, n.ahead=10*12)   #n.ahead=year*month
ts.plot(AP, AP.predict, col=1:2, xlab="Time", ylab="Observed/Predicted")
legend("topleft", c("data","prediction"), col=c(1,2), lty=1, bty="n")

# ALTERNATIVE APPROACH
library(forecast)
AP.predict2<-forecast.HoltWinters(AP.hw, h=10*12)
print(AP.predict2)
plot.forecast(AP.predict2, main="")
legend("topleft", c("data","prediction","95% CI", "80% CI"),
       col=c(1,4, "gray", "lightsteelblue3"), lty=1, lwd=2, bty="n")
```

Conditional Heteroscedasticity

- So far we assumed that the **volatility** of time series is constant (independent)
- Volatility is not directly observable, but it has some characteristics
  - Volatility of many time series does not appear to be constant over time
    - There are clusters of high and low volatility periods
  - Volatility evolves over time in a continuous manner
    - Volatility jumps are rare
  - Volatility does not diverge to infinity
    - Volatility varies within some fixed range
- There are several measures of volatility, but *conditional standard deviation (variance)* is commonly used
- The variance of the error term at t is a function of error terms of previous periods
  - Often the variance is related to the squares of the previous error term

Conditional Heteroscedasticity

- Consider the conditional mean and variance of Y_t given $F_{t-1} = \{Y_{t-1}, Y_{t-2}, \dots\}$ is the information set available at time $t-1$

$$\begin{aligned}\mu_{Y,t} &= \mathrm{E}(Y_t|F_{t-1}) \\ \sigma_t^2 &= \mathrm{Var}(Y_t|F_{t-1}) = \mathrm{E}[(Y_t - \mu_{Y,t})^2|F_{t-1}]\end{aligned}$$

- We can assume that Y_t follows a simple time series model, such as ARMA(p,q)

$$\begin{aligned}Y_t &= \mu_{Y,t} + e_t \\ &= \sum_{i=1}^{p} \beta_i Y_{t-i} + e_t + \sum_{j=1}^{q} \theta_j e_{t-j} \\ \mu_{Y,t} &= \sum_{i=1}^{p} \beta_i Y_{t-i} + \sum_{j=1}^{q} \theta_j e_{t-j}\end{aligned}$$

- Then,

$$\sigma_t^2 = \mathrm{Var}(Y_t|F_{t-1}) = \mathrm{Var}(e_t|F_{t-1})$$

Conditional Heteroscedasticity

- The conditional heteroscedastic models are concerned with the evolution of σ_t^2
- The manner under which σ_t^2 evolves over time distinguishes one volatility model from another
- Conditional heteroscedastic models can be classified into two general categories
  1. Use an **exact function** to govern the evolution of σ_t^2
  2. Use a **stochastic equation** to describe σ_t^2
- The ARCH/GARCH models belong to the *first category*, whereas the stochastic volatility model is in the second category

Conditional Heteroscedasticity

- The models used in modeling time series that exhibit time-varying volatility clustering are referred to as **conditional heteroscedastic (CH) models**
  - Auto-regressive CH (ARCH) (Engle, 1982, *Econometrica*)
  - Generalized ARCH (GARCH) (Bollerslev, 1986, *Journal of Econometrics*)
  - Integrated GARCH (IGARCH) (Engle and Bollerslev, 1986, *Econometric Reviews*)
  - GARCH-in-mean (GARCH-M) (Engle, Lilien and Robins, 1987, *Econometrica*)
  - Quadratic GARCH (QGARCH) (Sentana, 1995, *Review of Economic Studies*)
  - Nonlinear GARCH (NGARCH) (Engle and Bollerslev, 1986, *Econometric Reviews*; Engle and Ng, 1993, *Journal of Finance*)
  - Exponential GARCH (EGARCH) (Nelson, 1991, *Econometrica*)
  - Threshold GARCH (TGARCH) (Glosten et al, 1993, *Journal of Finance*; Zakoian, 1994, *Journal of Economic Dynamics and Control*)

Auto-Regressive Conditional Heteroscedasticity Model

ARCH(1) Model (Engle, 1982, *Econometrica*)

Auto-regressive conditional heteroscedasticity model of order 1, or ARCH(1), is

$$\left.\begin{aligned} e_t &= \nu_t \sigma_t \\ \sigma_t^2 &= \alpha_0 + \alpha_1 e_{t-1}^2 \end{aligned}\right\} \implies e_t = \nu_t \sqrt{\alpha_0 + \alpha_1 e_{t-1}^2}$$

- e_t is conditionally heteroscedastic error term (i.e., serially uncorrelated, but dependent)
- ν_t is a white noise process ($\nu_t \sim$ iid $\mathrm{WN}(0, \sigma^2)$)
- $\alpha_0 > 0$
- $0 \leq \alpha_1 < 1$

ARCH(m) Model

Auto-regressive conditional heteroscedasticity model of order m, or ARCH(m), is

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \alpha_2 e_{t-2}^2 + \cdots + \alpha_m e_{t-m}^2$$

Auto-Regressive Conditional Heteroscedasticity Model

Notes:

- Large past squared shocks $\{e_{t-i}^2\}_{i=1}^m$ imply a large conditional variance σ_t^2
- Consequently, e_t tends to assume a large value (in modulus)
- This means that, under the ARCH framework, large shocks tend to be followed by another large shock
  - That is, the probability of obtaining a large variate is greater than that of a smaller variance

- See Tsay (2010, Chapter 3.4) and Greene (2011, Chapter 20.10) for moments, estimation, and forecasting in ARCH models

Auto-Regressive Conditional Heteroscedasticity Model

ARMA(p, q)-ARCH(m)

We can combine the ARCH(m) model for volatility with ARMA(p, q) model for mean

$$\begin{aligned} Y_t &= \sum_{i=1}^{p} \beta_i Y_{t-i} + e_t + \sum_{j=1}^{q} \theta_j e_{t-j} \\ e_t &= \nu_t \sigma_t = \nu_t \sqrt{\alpha_0 + \sum_{k=1}^{m} \alpha_k e_{t-k}^2} \end{aligned}$$

- Advantage of ARCH models
  - Simple, but able to generate volatility clustering
- Weakness of ARCH models
  - Large number of lags often required to adequately describe the volatility process

Auto-Regressive Conditional Heteroscedasticity Model

Step-by-step approach:

1. Specify and estimate mean equation (i.e., ARIMA)
   - Use ACF, PACF, Q-statistic to identify any serial dependence in Y_t
   - Find out if there is serial dependence that can be modeled
2. Use residuals of the mean equation $(e_t = Y_t - \mu_{Y,t})$ to identify **ARCH effects**
   - Analyze ACF, PACF, and Q-statistics of **squared residuals** (e_t^2) to check for conditional heteroscedasticity
3. Specify a volatility model if ARCH effects are statistically significant, perform a joint estimation of the mean and volatility equations
4. Check the fitted model for adequacy, modify it if necessary

Auto-Regressive Conditional Heteroscedasticity Model

- If the model is properly specified, the **standardized residuals**, $\frac{e_t}{\sigma_t} = \nu_t$, should be iid, and thus
  - should not be serially correlated
  - should not exhibit periods of increased volatility
  - their squares should not be serially correlated

Implementation in R

fGarch package:

- Uses quasi-MLE to estimate ARCH/GARCH models
- Allows for several types of innovational distributions (by default Gaussian dist)
- `garchFit()`
  - Fits univariate ARCH/GARCH models
  - `cond.dist = c("norm", "snorm", "ged", "sged", "std", "sstd", "snig", "QMLE")`: the desired conditional distribution
  - `"QMLE"`: Quasi-Maximum Likelihood Estimation, which assumes normal distribution and uses robust standard errors for inference
  - `algorithm = c("nlminb", "lbfgsb", "nlminb+nm", "lbfgsb+nm")`

ARCH - Example

```
# ARCH
library(astsa)
data<-ts(nyse)
plot(data, main="Returns of the NYSE", xlab="Time", ylab="Y")

# ACF & PACF
par(mfrow=c(2,1))
acf(nyse, main="Series")
pacf(nyse, main="")

# MEAN EQUATION MODELING
library(forecast)
auto<-auto.arima(data, ic="bic", seasonal=TRUE,
                 stationary=FALSE, stepwise=FALSE, trace=TRUE)
summary(auto)
tsdiag(auto, gof.lag=36)
LB1<-Box.test(auto$residuals, lag=36, type="Ljung")
print(LB1)
LB2<-Box.test(auto$residuals^2, lag=36, type="Ljung")  #H0: No ARCH effect
print(LB2)

# VOLATILITY EQUATION MODELING
install.packages("fGarch")
library(fGarch)
mod1<-garchFit(~ arma(0,1) + garch(4,0), data = diff(data), cond.dist = "norm", trace=TRUE)
print(mod1)
summary(mod1)    #omega in the output is alpha0 from our notation
```

ARCH - Example

```
# DIAGNOSTICS
plot(mod1)        #to see plot options; 0 to exit
par(mfrow=c(2,2))
plot(mod1@residuals, type="l", xlab="", ylab="", main="residuals")
plot(mod1@sigma.t, type="l", xlab="", ylab="", main="conditional standard deviation")
plot(mod1@residuals/mod1@sigma.t, type="l", xlab="", ylab="", main="standardized residuals")
plot((mod1@residuals/mod1@sigma.t)^2, type="l", xlab="", ylab="",
     main="squared standardized residuals")

# ACF & PACF
nu<-mod1@residuals/mod1@sigma.t                    #standardized residuals
nu2<-(mod1@residuals/mod1@sigma.t)^2               #squared standardized residuals

par(mfrow=c(2,1))
acf(nu, type="correlation", lag.max=36, ylab="", main="ACF for standardized residuals")
acf(nu, type="partial", lag.max=36, ylab="", main="PACF for standardized residuals")

acf(nu2, type="correlation", lag.max=36, ylab="", main="ACF for squared standardized residuals")
acf(nu2, type="partial", lag.max=36, ylab="", main="PACF for squared standardized residuals")

# LJUNG-BOX TEST
LB3<-Box.test(nu2, lag = 12, type = "Ljung")
print(LB3)

# PREDICTION
par(mfrow=c(1,1))
mod1.fcst<-predict(mod1, n.ahead=24, plot=TRUE)
```

Auto-Regressive Conditional Heteroscedasticity Model

Weaknesses:

- The model assumes that positive and negative shocks have the same effects on volatility because it depends on the square of the previous shocks
- The ARCH model does not provide any new insight for understanding the *source* of variations of a financial time series
  - It merely provides a mechanical way to describe the behavior of the conditional variance.
- Although the ARCH model is simple, it often requires many parameters (m) to adequately describe the volatility process

Generalized Auto-Regressive Conditional Heteroscedasticity

GARCH(1,1) Model (Bollerslev, 1986, *Journal of Econometrics*)

Generalized auto-regressive conditional heteroscedasticity model of order (1,1), or GARCH(1,1), is

$$\left.\begin{array}{l} e_t = \nu_t \sigma_t \\ \sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \end{array}\right\} \implies e_t = \nu_t \sqrt{\alpha_0 + \alpha_1 e_{t-1}^2 + \beta_1 \sigma_{t-1}^2}$$

- e_t is conditionally heteroscedastic error term
- ν_t is a white noise process ($\nu_t \sim$ iid $\mathrm{WN}(0, \sigma^2)$)
- $\alpha_0 > 0$
- $0 \leq \alpha_1 < 1$

GARCH(m, s) Model

Auto-regressive conditional heteroscedasticity model of order m, or ARCH(m), is

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \cdots + \alpha_m e_{t-m}^2 + \beta_1 \sigma_{t-1}^2 + \cdots + \beta_s \sigma_{t-s}^2$$

Generalized Auto-Regressive Conditional Heteroscedasticity

Notes:

- Large $\{e_{t-i}^2\}_{i=1}^m$ or $\{\sigma_{t-j}^2\}_{j=1}^s$ imply a large conditional variance σ_t^2
- Consequently, e_t tends to assume a large value (in modulus)
- This means that a large e_{t-1}^2 tends to be followed by another large e_t^2, generating, again, the well-known behavior of *volatility clustering*

- See Tsay (2010, Chapter 3.5) and Greene (2011, Chapter 20.10) for moments, estimation, and forecasting in GARCH models

Generalized Auto-Regressive Conditional Heteroscedasticity

ARMA(p, q)-GARCH(m, s)

We can combine the GARCH(m, s) model for volatility with ARMA(p, q) model for mean

$$
\begin{aligned}
Y_t &= \sum_{i=1}^{p} \beta_i Y_{t-i} + e_t + \sum_{j=1}^{q} \theta_j e_{t-j} \\
e_t &= \nu_t \sigma_t = \nu_t \sqrt{\alpha_0 + \sum_{k=1}^{m} \alpha_k e_{t-k}^2 + \sum_{m=1}^{s} \beta_m \sigma_{t-m}^2}
\end{aligned}
$$

- Identifying the order of GARCH model is not easy
- Lower order GARCH models are used in most applications, GARCH(1,1), GARCH(2,1), GARCH(1,2)

GARCH - Example

```
# GARCH
library(astsa)
data<-ts(nyse)

# MEAN EQUATION MODELING
library(forecast)
auto<-auto.arima(data, ic="bic", seasonal=TRUE, stationary=FALSE, stepwise=FALSE, trace=TRUE)
summary(auto)

# VOLATILITY EQUATION MODELING
install.packages("fGarch")
library(fGarch)
mod1<-garchFit(~ arma(1,1) + garch(1,1), data=data, cond.dist = "norm", trace=TRUE)
#print(mod1)
summary(mod1)

# DIAGNOSTICS
plot(mod1)        #to see plot options;  0 to exit
par(mfrow=c(2,2))
plot(mod1@residuals, type="l", xlab="", ylab="", main="residuals")
plot(mod1@sigma.t, type="l", xlab="", ylab="", main="conditional standard deviation")
plot(mod1@residuals/mod1@sigma.t, type="l", xlab="", ylab="", main="standardized residuals")
plot((mod1@residuals/mod1@sigma.t)^2, type="l", xlab="", ylab="",
     main="squared standardized residuals")
```

GARCH - Example

```r
nu<-mod1@residuals/mod1@sigma.t                    #standardized residuals
nu2<-(mod1@residuals/mod1@sigma.t)^2               #squared standardized residuals

par(mfrow=c(2,1))
acf(nu, type="correlation", lag.max=36, ylab="", main="ACF for standardized residuals")
acf(nu, type="partial", lag.max=36, ylab="", main="PACF for standardized residuals")

acf(nu2, type="correlation", lag.max=36, ylab="", main="ACF for squared standardized residuals")
acf(nu2, type="partial", lag.max=36, ylab="", main="PACF for squared standardized residuals")

# LJUNG-BOX TEST
LB<-Box.test(nu2, lag = 12, type = "Ljung")
print(LB)

# PREDICTION
par(mfrow=c(1,1))
mod1.fcst<-predict(mod1, n.ahead=24, plot=TRUE)
```

Multivariate Time Series

- Generalization of the univariate time series modelling approach
- Data on multiple time series (components): $Y_{1,t}, Y_{2,t}, \ldots$
- Why multivariate analysis?
  - Interactions and dynamic relationship between variables
  - Feeback between time series
  - If there is no feeback between $Y_{1,t}$ and $Y_{2,t}$ series, $Y_{2,t}$ can be entered into the model of $Y_{1,t}$ (i.e., $Y_{2,t}$ is exogenous)

Multivariate Time Series

Example I: world financial markets (e.g., S&P 500, FTSE 100, Nikkei)

- Economic globalization and internet communication have accelerated the integration
- Price movements in one market can spread easily and instantly to another market
- Financial markets are more dependent on each other than ever before
- One must consider the markets *jointly* to better understand the dynamic structure of the global finance

Example II: returns on multiple assets (e.g., IBM, Microsoft, GM)

- The dynamic relationships between returns of the assets
- Need methods to *jointly* study multiple return series

Multivariate Time Series

- Let $\{\mathbf{Y}_t\} = \{\ldots, \mathbf{Y}_{t-1}, \mathbf{Y}_t, \mathbf{Y}_{t+1}, \ldots\}$ be an m-dimensional $(m > 1)$ stochastic process consisting of random m-vectors
- m-variate time series is

$$\mathbf{Y}_t = \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{m,t} \end{bmatrix} \quad t \in T$$

- That is, $\{\mathbf{Y}_t\}$ consists of m component time series

$$Y_{1,t}, Y_{2,t}, \ldots, Y_{m,t}$$

Multivariate Time Series - Weak Stationarity

Weak Stationarity

$\{\mathbf{Y}_t : t \in T\}$ is covariance stationary if **every** component time series, i.e., $\{Y_{1,t}\}, \{Y_{2,t}\}, \ldots, \{Y_{m,t}\}$, is covariance stationary

- This implies that the multivariate mean (first moment) and the variance-covariance matrix (second moment) are time-invariant
- The $m \times 1$ mean vector of $\mathbf{Y}_t$ is

$$\boldsymbol{\mu}_{\mathbf{Y}} = \mathrm{E}(\mathbf{Y}_t) = \begin{bmatrix} \mathrm{E}(Y_{1,t}) \\ \mathrm{E}(Y_{2,t}) \\ \vdots \\ \mathrm{E}(Y_{m,t}) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_m \end{bmatrix}$$

for all $t \in T$

Return

Multivariate Time Series

- The $m \times m$ variance-covariance matrix of $\mathbf{Y}_t$ is

$$
\begin{aligned}
\Gamma(0) &= \mathrm{Var}(\mathbf{Y}_t) \\
&= \mathrm{E}[(\mathbf{Y}_t - \boldsymbol{\mu}_{\mathbf{Y}})(\mathbf{Y}_t - \boldsymbol{\mu}_{\mathbf{Y}})'] \\
&= \begin{bmatrix}
\mathrm{Var}(Y_{1,t}) & \mathrm{Cov}(Y_{1,t}, Y_{2,t}) & \cdots & \mathrm{Cov}(Y_{1,t}, Y_{m,t}) \\
\mathrm{Cov}(Y_{2,t}, Y_{1,t}) & \mathrm{Var}(Y_{2,t}) & \cdots & \mathrm{Cov}(Y_{2,t}, Y_{m,t}) \\
\vdots & \vdots & \ddots & \vdots \\
\mathrm{Cov}(Y_{m,t}, Y_{1,t}) & \mathrm{Cov}(Y_{m,t}, Y_{2,t}) & \cdots & \mathrm{Var}(Y_{m,t})
\end{bmatrix}
\end{aligned}
$$

  - $\Gamma(0)$ is symmetric

Multivariate Time Series

- The $m \times m$ cross-correlation matrix for $\mathbf{Y}_t$ is

$$\begin{aligned} R(0) &= D^{-\frac{1}{2}}\Gamma(0)D^{-\frac{1}{2}} \\ D &= \mathrm{diag}(\Gamma(0)) \end{aligned}$$

  - $R(0)$ is symmetric

- The (i, j)'s element of $R(0)$ is

$$\rho_{ij} = \frac{\mathrm{Cov}(Y_{i,t}, Y_{j,t})}{\mathrm{std}(Y_{i,t})\mathrm{std}(Y_{j,t})}$$

- ρ_{ij} is referred to as a **concurrent** or **contemporaneous** correlation (correlation of the two series at time t)

Multivariate Time Series

- The lead-lag relationship between component series is important
- The $m \times m$, lag-h autocovariance (cross-covariance) matrix of $\mathbf{Y}_t$ does not depend on t but only on h (for a weakly stationary series)

$$
\begin{aligned}
\Gamma(h) &= \mathrm{Cov}(\mathbf{Y}_{t+h}, \mathbf{Y}_t) \\
&= \mathrm{E}[(\mathbf{Y}_{t+h} - \boldsymbol{\mu}_{\mathbf{Y}})(\mathbf{Y}_t - \boldsymbol{\mu}_{\mathbf{Y}})'] \\
&= \begin{bmatrix} \gamma_{1,1}(h) & \gamma_{1,2}(h) & \cdots & \gamma_{1,m}(h) \\ \gamma_{2,1}(h) & \gamma_{2,2}(h) & \cdots & \gamma_{1,m}(h) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{m,1}(h) & \gamma_{m,2}(h) & \cdots & \gamma_{m,m}(h) \end{bmatrix}
\end{aligned}
$$

  - $\Gamma(h)$ is not symmetric

Multivariate Time Series

- The (i, j)'s element of $\Gamma(h)$ is $\text{Cov}(Y_{i,t+h}, Y_{j,t})$
  - The diagonal elements $\gamma_{j,j}(h)$ are the *autocovariance* functions of $Y_{j,t}$, $1 \leq j \leq m$
  - The off-diagonal elements $\gamma_{j,k}(h)$ are the *cross-covariance* functions of $Y_{j,t}$ and $Y_{k,t}$, $1 \leq j \neq k \leq m$
- If $\text{Cov}(Y_{i,t+h}, Y_{j,t}) \neq 0$, then $Y_{j,t}$ **leads** $Y_{i,t}$
- If $Y_{j,t}$ leads $Y_{i,t}$ and $Y_{i,t}$ leads $Y_{j,t}$, then there is **feedback**

Multivariate Time Series

- The $m \times m$, lag-h cross-correlation matrix of $\mathbf{Y}_t$ is

$$\begin{aligned} R(h) &= D^{-\frac{1}{2}}\Gamma(h)D^{-\frac{1}{2}} \\ D &= \mathrm{diag}(\Gamma(h)) \end{aligned}$$

  - $R(h)$ is not symmetric
- The (i,j)'s element of $R(h)$ is

$$\rho_{ij}(h) = \frac{\mathrm{Cov}(Y_{i,t+h}, Y_{j,t})}{\mathrm{std}(Y_{i,t+h})\mathrm{std}(Y_{j,t})}$$

Multivariate Time Series

- An estimator of $\boldsymbol{\mu}$ is the *m*-dimensional sample mean

$$\bar{Y}_n = \frac{1}{n}\sum_{t=1}^{n} Y_t$$

- An estimator of $\Gamma(h)$, the lag-h autocovariance (cross-covariance) matrix, is the sample lag-h autocovariance

$$\hat{\Gamma}(h) = \frac{1}{n}\sum_{t=1}^{n-h}(Y_{t+h} - \bar{Y}_t)(Y_t - \bar{Y}_t)', \quad h \geq 0$$

- An estimator of $R(h)$, the lag-h cross-correlation matrix, is the sample lag-h cross-correlation

$$\begin{aligned} \hat{R}(h) &= \hat{D}^{-\frac{1}{2}}\hat{\Gamma}(h)\hat{D}^{-\frac{1}{2}} \\ \hat{D} &= \mathrm{diag}(\hat{\Gamma}(h)) \end{aligned}$$

Multivariate Time Series

Multivariate Portmanteau Test: To formally test

$$H_0 : R(1) = R(2) = \cdots = R(k) = 0$$
$$H_1 : R(j) \neq 0 \quad \text{for some } j = \{1, \dots, k\}$$

- **Multivariate Ljung-Box test statistic** (Hosking, 1980, *JASA*; Li, McLeod, 1981, *JRSS Series B*)

$$Q_m(k) = T^2 \sum_{\ell=1}^{m} \frac{1}{T-\ell} \mathrm{tr}(\hat{\Gamma}'(\ell)\hat{\Gamma}^{-1}(0)\hat{\Gamma}(\ell)\hat{\Gamma}^{-1}(0)) \sim \chi_{m^2 k}$$

  - T is the sample size
  - m is the dimension of $\mathbf{Y}_t$
  - $\mathrm{tr}(A)$ is the trace of matrix A

Multivariate Time Series - Example

```r
# EXAMPLE - MONTHLY LOG RETURNS OF IBM STOCK AND S&P 500 (JAN1926 - DEC2008)
data.r<-read.table("http://faculty.chicagobooth.edu/ruey.tsay/teaching/fts/m-ibmspln.dat")
ibm<-data.r[,1]
sp5<-data.r[,2]
data.r.2<-cbind(ibm,sp5)
data.r.3<-data.frame(data.r.2)
head(data.r.3)

# PLOT THE SERIES
par(mfrow=c(2,1))
plot(seq(1, nrow(data.r.3),1),data.r.3[,1],
     type="l",
     col="blue",
     lwd=1.5,
     xlab="Time",
     ylab= "Return",
     main="IBM")
plot(seq(1, nrow(data.r.3),1),data.r.3[,2],
     type="l",
     col="red",
     lty=1,
     lwd=1.5,
     xlab="Time",
     ylab= "Return",
     main="SP500")
```

Multivariate Time Series - Example

```
# SCATTER PLOT
par(mfrow=c(2,2))
plot(data.r.3[,2],data.r.3[,1],
     xlab="SP500",
     ylab= "IBM")
plot(data.r.3[,2][1:(nrow(data.r.3)-1)],data.r.3[,1][2:nrow(data.r.3)],
     xlab="SP500 - Lag 1",
     ylab= "IBM")
plot(data.r.3[,1][1:(nrow(data.r.3)-1)],data.r.3[,2][2:nrow(data.r.3)],
     xlab="IBM - Lag 1",
     ylab= "SP500")

# SAMPLE MOMENTS
colMeans(data.r.3)
cov(data.r.3)
cor(data.r.3)

# ACF, PACF, CCF
acf(data.r.3[,1])
pacf(data.r.3[,1])

acf(data.r.3[,2])
pacf(data.r.3[,2])

ccf(data.r.3[,1],data.r.3[,2])
ccf(data.r.3[,1],data.r.3[,2],plot=FALSE)
ccf(data.r.3[,2],data.r.3[,1])
ccf(data.r.3[,1],data.r.3[,1]) #same as acf
```

Multivariate Time Series Models

- **Vector autoregressive (VAR) model**
  - Sims (1980) *Econometrica*
- **Vector error correction (VEC) model**
  - Johansen (1991) *Econometrica*
- **Vector moving average (VMA) model**
- **Vector autoregressive moving average (VARMA) model**

Vector Autoregressive Process

VAR(p) Process (Sims, 1980, *Econometrica*)

The m-dimensional multivariate time series $\{\mathbf{Y}_t\}$ follows the VAR(p) model with auto-regressive order p if

$$\mathbf{Y}_t = \mathbf{c} + \mathbf{\Phi}_1 \mathbf{Y}_{t-1} + \mathbf{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \mathbf{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t$$

- $\mathbf{c} = (c_1, c_2, \ldots, c_m)'$ is an m-dimensional vector of constants
- $\mathbf{\Phi}_1, \mathbf{\Phi}_2, \ldots, \mathbf{\Phi}_p$ are $m \times m$ matrices of coefficients
- VAR(p) model has $m + pm^2$ parameters
- $\mathbf{e}_t$ is multivariate white noise (a.k.a. "innovation"), MWN$(\mathbf{0}_m, \mathbf{\Sigma})$

$$\begin{aligned}
\mathrm{E}(\mathbf{e}_t) &= \underset{m\times 1}{\mathbf{0}} \\
\mathrm{Var}(\mathbf{e}_t) = \mathrm{E}(\mathbf{e}_t\mathbf{e}_t') &= \underset{m\times m}{\mathbf{\Sigma}} && \equiv \Gamma(0) \\
\mathrm{Cov}(\mathbf{e}_{t+h}, \mathbf{e}_t) = \mathrm{E}(\mathbf{e}_{t+h}\mathbf{e}_t') &= \underset{m\times m}{\mathbf{0}} && \equiv \Gamma(h)
\end{aligned}$$

- VAR models represent the correlations among a set of variables

Vector Autoregressive Process - Bivariate VAR(1)

- Consider the bivariate, $m = 2$, VAR(1) model

$$\underset{2\times 1}{\mathbf{Y}_t} = \underset{2\times 1}{\mathbf{c}} + \underset{2\times 2}{\mathbf{\Phi}}\,\underset{2\times 1}{\mathbf{Y}_{t-1}} + \underset{2\times 1}{\mathbf{e}_t}$$

where

$$\mathbf{Y}_t = \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, \quad \mathbf{\Phi} = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{bmatrix},$$
$$\mathbf{Y}_{t-1} = \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix}, \quad \mathbf{e}_t = \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix}$$

- Alternatively,

$$\begin{aligned} Y_{1,t} &= c_1 + \phi_{1,1} Y_{1,t-1} + \phi_{1,2} Y_{2,t-1} + e_{1,t} \\ Y_{2,t} &= c_2 + \phi_{2,1} Y_{1,t-1} + \phi_{2,2} Y_{2,t-1} + e_{2,t} \end{aligned}$$

Vector Autoregressive Process - Bivariate VAR(1)

Notes:

- $\phi_{1,2}$ is the conditional effect of $Y_{2,t-1}$ on $Y_{1,t}$ in the presence of $Y_{1,t-2}$
- $\phi_{1,2} \neq 0$, $\beta_{2,1} \neq 0$: feeback relationship between the two series
- $\phi_{1,2} = 0$, $\phi_{2,1} \neq 0$: unidirectional relationship from $Y_{1,t}$ to $Y_{2,t}$ (i.e., $Y_{1,t}$ does not depend on $Y_{2,t}$)
- $\phi_{1,2} = \phi_{2,1} = 0$: $Y_{1,t}$ and $Y_{2,t}$ are uncoupled

- Test whether lags of one variable ($Y_{1,t}$) enter the equation for another variable ($Y_{2,t}$)

Vector Autoregressive Process

Granger Causality

- $Y_{j,t}$ **Granger causes** $Y_{i,t}$ if at least one of the coefficients on lags of $Y_{j,t}$ in the equation for $Y_{i,t}$ is non-zero
  - For the ith equation of a bivariate VAR(p), this implies

$$Y_{i,t} = c_i + \sum_{\ell=1}^{p} \phi_{\ell,i,i} Y_{i,t-\ell} + \sum_{\ell=1}^{p} \phi_{\ell,i,j} Y_{j,t-\ell} + e_{i,t}$$

$$\mathrm{E}(Y_{i,t}|Y_{i,t-1}, Y_{i,t-2}, \dots, Y_{j,t-1}, Y_{j,t-2}, \dots) \neq \mathrm{E}(Y_{i,t}|Y_{i,t-1}, Y_{i,t-2}, \dots)$$

  - Using F-statistic, test

$$H_0 : \phi_{1,i,j} = \phi_{2,i,j} = \cdots = \phi_{p,i,j} = 0$$

$$H_1 : \exists \ell \in \{1, \dots, p\} : \phi_{\ell,i,j} \neq 0$$

- Granger causality $\not\Rightarrow$ contemporeneous causality between $Y_{i,t}$ and $Y_{j,t}$
- If the innovation to $Y_{i,t}$ and the innovation to $Y_{j,t}$ are correlated we say there is **instantaneous causality**

VAR - Structural vs Reduced Form

- VAR models presented above are called a **reduced-form** model
  - Commonly used in the literature (ease in estimation)
  - It does not show explicitly the concurrent dependence between the component series (e.g., $Y_{1,t}$ and $Y_{2,t}$)
- **Structural VAR** (SVAR) form explicitly models the concurrent/contemporaneous dependence between the component series

VAR - Structural vs Reduced Form

- Consider bivariate, $m = 2$, SVAR(1) model

$$\begin{aligned} Y_{1,t} &= c_1 + \psi_{1,0} Y_{2,t} + \psi_{1,1} Y_{1,t-1} + \psi_{1,2} Y_{2,t-1} + e_{1,t} \\ Y_{2,t} &= c_2 + \psi_{2,0} Y_{1,t} + \psi_{2,1} Y_{1,t-1} + \psi_{2,2} Y_{2,t-1} + e_{2,t} \end{aligned}$$

- In matrix form,

$$\underset{2\times 2}{\mathbf{\Psi}_0}\underset{2\times 1}{\mathbf{Y}_t} = \underset{2\times 1}{\mathbf{c}} + \underset{2\times 2}{\mathbf{\Psi}_1}\underset{2\times 1}{\mathbf{Y}_{t-1}} + \underset{2\times 1}{\mathbf{e}_t}$$

where

$$\mathbf{\Psi}_0 = \begin{bmatrix} 1 & -\psi_{1,0} \\ -\psi_{2,0} & 1 \end{bmatrix}, \quad \mathbf{Y}_t = \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix},$$
$$\mathbf{\Psi}_1 = \begin{bmatrix} \psi_{1,1} & \psi_{1,2} \\ \psi_{2,1} & \psi_{2,2} \end{bmatrix}, \quad \mathbf{Y}_{t-1} = \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix}, \quad \mathbf{e}_t = \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix}$$

- **Endogeneity problem**: $Y_{1,t}$ has a contemporaneous effect on $Y_{2,t}$, and vice versa

VAR - Structural vs Reduced Form

- Premultiplying by $\mathbf{\Psi}_0^{-1}$, we can obtain a *reduced-form* expression

$$\begin{aligned}\mathbf{\Psi}_0^{-1}\mathbf{\Psi}_0\mathbf{Y}_t &= \mathbf{\Psi}_0^{-1}\mathbf{c} + \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_1\mathbf{Y}_{t-1} + \mathbf{\Psi}_0^{-1}\mathbf{e}_t \\ \underset{2\times1}{\mathbf{Y}_t} &= \underset{2\times1}{\mathbf{c}^*} + \underset{2\times2}{\mathbf{\Phi}_1}\underset{2\times1}{\mathbf{Y}_{t-1}} + \underset{2\times1}{\mathbf{e}_t^*}\end{aligned}$$

  - $\mathbf{c}^* = \mathbf{\Psi}_0^{-1}\mathbf{c}$
  - $\mathbf{\Phi}_1 = \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_0$
  - $\mathbf{e}_t^* = \mathbf{\Psi}_0^{-1}\mathbf{e}_t$
  - $\mathrm{E}(\mathbf{e}_t^*) = \mathbf{0}$
  - $\mathrm{Var}(\mathbf{e}_t^*) = \mathbf{\Psi}_0^{-1}\mathbf{\Sigma}\mathbf{\Psi}_\mathbf{0}^{\mathbf{-1}'}$
- This system can now be estimated equation by equation using standard OLS

VAR - Structural vs Reduced Form - Example

- Consider bivariate, $m = 2$, SVAR(1) model

$$\underset{2\times 2}{\mathbf{\Psi}_0}\underset{2\times 1}{\mathbf{Y}_t} = \underset{2\times 1}{\mathbf{c}} + \underset{2\times 2}{\mathbf{\Psi}_1}\underset{2\times 1}{\mathbf{Y}_{t-1}} + \underset{2\times 1}{\mathbf{e}_t}$$

where

$$\mathbf{\Psi}_0 = \begin{bmatrix} 1 & 0 \\ -0.5 & 1 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{\Psi}_1 = \begin{bmatrix} 0.6 & 0.2 \\ -0.1 & 0.5 \end{bmatrix}, \quad \mathbf{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- Premultiplying by $\mathbf{\Psi}_0^{-1} = \begin{bmatrix} 1 & 0 \\ 0.5 & 1 \end{bmatrix}$, we can obtain the *reduced-form* VAR(1)

$$\underset{2\times 1}{\mathbf{Y}_t} = \underset{2\times 1}{\mathbf{c}^*} + \underset{2\times 2}{\mathbf{\Phi}_1}\underset{2\times 1}{\mathbf{Y}_{t-1}} + \underset{2\times 1}{\mathbf{e}_t^*}$$

where

$$\mathbf{\Phi}_1 = \begin{bmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{\Sigma} = \begin{bmatrix} 2 & 1 \\ 1 & 2.5 \end{bmatrix}$$

VAR - Structural vs Reduced Form - Example

```
# SIMULATED DATA
set.seed(123)
n<-100
mu<-c(0,0)
sigma<-matrix(c(2,1,1,2.5), nrow=2, ncol=2, byrow=TRUE)
bvn<-mvrnorm(n, mu=mu, Sigma=sigma ) #from MASS package
e1<-bvn[,1]
e2<-bvn[,2]

y1<-rep(NA,100)
y1[1]<-e1[1]
y2<-rep(NA,100)
y2[1]<-e2[1]

for(i in 2:100){
  y1[i]=0.6*y1[i-1] + 0.2*y2[i-1] + e1[i]
  y2[i]=0.2*y1[i-1] + 0.6*y2[i-1] + e2[i]
}

plot(seq(1,100,1),y1,
     type="l", col="blue", lwd=1.5,
     xlab="Time",
     ylab=as.expression(bquote(y[1]~","~y[2])))
lines(seq(1,100,1), y2,
      type="l", col="red", lty=2, lwd=1.5)
abline(h=0, lty=1, col="grey")
legend("topleft", c(as.expression(bquote(y[1])),as.expression(bquote(y[2]))),
       col=c("blue","red"),
       lty=c(1,2),
       bty="n")
```

Vector Autoregressive Process - Estimation

- Building a VAR(p) model for a given time series
  - Tsay (2010) ch.8.2.4
- Similar iterative procedure as in the univariate case
  1. Order specification (choose p)
  2. Estimation (component-wise OLS, MLE)
  3. Model checking (AIC, BIC)
  4. Diagnostics of the residuals

Vector Autoregressive Process - Forecasting

- For simplicity, consider VAR(1) model
- **1-Step-Ahead Forecast**
  - From VAR(1) model, we have

$$\mathbf{y}_{t+1} = \mathbf{c} + \mathbf{\Phi}\mathbf{y}_t + \mathbf{e}_{t+1}$$

  - Under the minimum squared error loss function, the point forecast of $\mathbf{y}_{t+1}$ given $\mathbf{F}_t$ (the information available at time t) is the conditional expectation

$$\hat{\mathbf{y}}_t(1) = \mathrm{E}(\mathbf{y}_{t+1}|\mathbf{F}_t) = \mathbf{c} + \mathbf{\Phi}\mathbf{y}_t$$

with the associated forecast error of

$$\begin{aligned}
\hat{\mathbf{e}}_t(1) &= \mathbf{y}_{t+1} - \hat{\mathbf{y}}_t(1) \\
&= (\mathbf{c} + \mathbf{\Phi}\mathbf{y}_t + \mathbf{e}_{t+1}) - (\mathbf{c} + \mathbf{\Phi}\mathbf{y}_t) \\
&= \mathbf{e}_{t+1}
\end{aligned}$$

Vector Autoregressive Process - Forecasting

- **2-Step-Ahead Forecast**
  - From VAR(1) model, we have

$$\mathbf{y}_{t+2} = \mathbf{c} + \mathbf{\Phi}\mathbf{y}_{t+1} + \mathbf{e}_{t+2}$$

  - Under the minimum squared error loss function, the point forecast of $\mathbf{y}_{t+2}$ given $\mathbf{F}_t$ (the information available at time t) is the conditional expectation

$$\hat{\mathbf{y}}_t(2) = \mathrm{E}(\mathbf{y}_{t+2}|\mathbf{F}_t) = \mathbf{c} + \mathbf{\Phi}\hat{\mathbf{y}}_t(1)$$

  with the associated forecast error of

$$\begin{aligned} \hat{\mathbf{e}}_t(2) &= \mathbf{y}_{t+2} - \hat{\mathbf{y}}_t(2) \\ &= (\mathbf{c} + \mathbf{\Phi}\mathbf{y}_{t+1} + \mathbf{e}_{t+2}) - (\mathbf{c} + \mathbf{\Phi}\hat{\mathbf{y}}_t(1)) \\ &= \mathbf{\Phi}\mathbf{e}_{t+1} + \mathbf{e}_{t+2} \end{aligned}$$

VAR - Implementation in R

vars package

- Tools for lag selection, diagnostic testing, forecasting and causality analysis
- `VARselect()`: selects the order of auto-regressive (VAR) models
- `VAR()`: estimates vector auto-regressive (VAR) models
- `irf()`: computes impulse response function of a VAR/SVAR models
- `SVAR()`: estimates structural vector auto-regressive (SVAR) models
- `SVER()`: estimates structural vector error-correction (SVER) models

- Review of related R packages:
  - Pfaff (2008) *Journal of Statistical Software*

VAR - Estimation - Example 1

```
# EXAMPLE 1 - SIMULATED DATA
data.s<-cbind(y1,y2)
data.s.2<-data.frame(data.s)

# VAR MODEL ORDER SELECTION
install.packages('vars')
library(vars)
ord.choice<-VARselect(data.s.2,
                      lag.max=10,
                      type="const")
print(ord.choice)

# ESTIMATE THE BEST VAR MODEL
var1.s<-VAR(data.s.2,
          p=1, type="const")
print(var1.s)
summary(var1.s)

# GRANGER CAUSALITY
causality(var1.s, cause="y1")
causality(var1.s, cause="y2")

# PREDICTION
var1.s.f<-predict(var1.s, n.ahead=12)
print(var1.s.f)
plot(var1.s.f, lwd=1.5)
fanchart(var1.s.f, lwd=1.5)
```

VAR - Estimation - Example 2

```
# EXAMPLE 2 - MONTHLY LOG RETURNS OF IBM STOCK AND S&P 500 (JAN1926 - DEC2008)
data.r<-read.table("http://faculty.chicagobooth.edu/ruey.tsay/teaching/fts/m-ibmspln.dat")
ibm<-data.r[,1]
sp5<-data.r[,2]
data.r.2<-cbind(ibm,sp5)
data.r.3<-data.frame(data.r.2)

# VAR MODEL ORDER SELECTION
library(vars)
ord.choice<-VARselect(data.r.3,
                      lag.max=10,
                      type="const")
print(ord.choice)

# ESTIMATE THE BEST VAR MODEL
var1.r<-VAR(data.r.3,
          p=1, type="const")
print(var1.r)
summary(var1.r)

# GRANGER CAUSALITY
causality(var1.r, cause="ibm")
causality(var1.r, cause="sp5")

# PREDICTION
var1.r.f<-predict(var1.r, n.ahead=20)
print(var1.r.f)
plot(var1.r.f, lwd=1.5)
fanchart(var1.r.f, lwd=1.5)
```

VAR - Impulse Response Function

- Consider a reduced form VAR(1) model

$$\underset{m\times 1}{\mathbf{Y}_t} = \underset{m\times 1}{\mathbf{c}} + \underset{m\times m}{\mathbf{\Phi}} \underset{m\times 1}{\mathbf{Y}_{t-1}} + \underset{m\times 1}{\mathbf{e}_t}$$

- We can represent the above model in a moving-average (MA) form as

$$\begin{aligned}
\mathbf{Y}_t &= \mathbf{c} + \mathbf{\Phi}(\mathbf{c} + \mathbf{\Phi}\mathbf{Y}_{t-2} + \mathbf{e}_{t-1}) + \mathbf{e}_t \\
&= \mathbf{c} + \mathbf{\Phi}\mathbf{c} + \mathbf{\Phi}^2\mathbf{Y}_{t-2} + \mathbf{\Phi}\mathbf{e}_{t-1} + \mathbf{e}_t \\
&\vdots \\
&= \sum_{\ell=0}^{\infty} \mathbf{\Phi}^{\ell}\mathbf{c} + \sum_{\ell=0}^{\infty} \mathbf{\Phi}^{\ell}\mathbf{e}_{t-\ell}
\end{aligned}$$

or, simply,

$$\underset{m\times 1}{\mathbf{Y}_t} = \underset{m\times 1}{\mathbf{c}^*} + \underset{m\times 1}{\mathbf{e}_t} + \underset{m\times m}{\mathbf{\Theta}_1} \underset{m\times 1}{\mathbf{e}_{t-1}} + \underset{m\times m}{\mathbf{\Theta}_2} \underset{m\times 1}{\mathbf{e}_{t-2}} + \cdots$$

  - This is VMA(∞) or a Wold MA represenatation

VAR - Impulse Response Function

- In expanded form,

$$\begin{bmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{m,t} \end{bmatrix} = \begin{bmatrix} c_1^* \\ c_2^* \\ \vdots \\ c_m^* \end{bmatrix} + \begin{bmatrix} e_{1,t} \\ e_{2,t} \\ \vdots \\ e_{m,t} \end{bmatrix} + \begin{bmatrix} \theta_{1,1,1} & \theta_{1,1,2} & \cdots & \theta_{1,1,m} \\ \theta_{1,2,1} & \theta_{1,2,2} & \cdots & \theta_{1,2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{1,m,1} & \cdots & \cdots & \theta_{1,m,m} \end{bmatrix} \begin{bmatrix} e_{1,t-1} \\ e_{2,t-1} \\ \vdots \\ e_{m,t-1} \end{bmatrix}$$
$$+ \begin{bmatrix} \theta_{2,1,1} & \theta_{2,1,2} & \cdots & \theta_{2,1,m} \\ \theta_{2,2,1} & \theta_{2,2,2} & \cdots & \theta_{2,2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{2,m,1} & \cdots & \cdots & \theta_{2,m,m} \end{bmatrix} \begin{bmatrix} e_{1,t-2} \\ e_{2,t-2} \\ \vdots \\ e_{m,t-2} \end{bmatrix} + \cdots$$

- $\theta_{\ell,i,j}$:
  - ℓ: lag order ($\ell = 1, 2, \ldots$)
  - i: series of interest ($Y_{i,t}$)
  - j: series whose lag effect ($e_{j,t-\ell}$) is considred for series i ($Y_{i,t}$)

VAR - Impulse Response Function

Impulse Response Function

- The $m \times m$ MA coefficient matrix $\mathbf{\Theta}_\ell$, $\ell = 1, 2, \ldots$, is called the **impulse response function (IRF)** of $\mathbf{Y}_t$
- Used to measure the impact of:
  - a unit increase (impulse) in $\mathbf{e}_{t-\ell}$ on $\mathbf{Y}_t$, with other variables dated to $t-\ell$ or earlier held constant
  - a unit increase (impulse) in $\mathbf{e}_t$ on $\mathbf{Y}_{t+\ell}$
  - (i,j)'s element of $\mathbf{\Theta}_\ell$, $\theta_{\ell,i,j}$, measures the impact of $e_{j,t-\ell}$ on $Y_{i,t}$

- Plot $\theta_{\ell,i,j}$ as function of ℓ to get a visual impression of the dynamic interrelationships between $e_{j,t-\ell}$ and $Y_{i,t}$
  - $\{Y_{1,t}, \underbrace{e_{1,t-\ell}}_{\equiv Y_{1,t-\ell}}\}$: plot $(\theta_{1,1,1}, \theta_{2,1,1}, \theta_{3,1,1}, \ldots)$ against $(1, 2, 3, \ldots)$
  - $\{Y_{1,t}, \underbrace{e_{2,t-\ell}}_{\equiv Y_{2,t-\ell}}\}$: plot $(\theta_{1,1,2}, \theta_{2,1,2}, \theta_{3,1,2}, \ldots)$ against $(1, 2, 3, \ldots)$

VAR - Impulse Response Function - Example

```
# IMPULSE RESPONSE FUNCTION

# EXAMPLE 1 - SIMULATED DATA
var1.s.irf<-irf(var1.s, n.ahead=30)
par(mfcol=c(2,2), cex=0.8, mar=c(3,4,2,2))
plot(var1.s.irf, plot.type="single", lwd=2)

# EXAMPLE 2 - MONTHLY LOG RETURNS OF IBM STOCK AND S&P 500, JAN1926 - DEC2008
var1.r.irf<-irf(var1.r, n.ahead=10)
par(mfcol=c(2,2), cex=0.8, mar=c(3,4,2,2))
plot(var1.r.irf, plot.type="single", lwd=2)
```

VAR - Weak Stationarity

- In a VAR model all variables need to be weakly stationary for the standard estimation methods to produce non-spurious results ▸ Details
- Using the back-shift operator, VAR(p) model can also be written as

$$
\begin{aligned}
\mathbf{Y}_t &= \mathbf{c} + \mathbf{\Phi}_1 \mathbf{Y}_{t-1} + \mathbf{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \mathbf{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t \\
\mathbf{Y}_t - \mathbf{\Phi}_1 \mathbf{Y}_{t-1} - \mathbf{\Phi}_2 \mathbf{Y}_{t-2} - \cdots - \mathbf{\Phi}_p \mathbf{Y}_{t-p} &= \mathbf{c} + \mathbf{e}_t \\
\underbrace{\left(I_m - \mathbf{\Phi}_1 B - \mathbf{\Phi}_2 B^2 - \cdots - \mathbf{\Phi}_p B^p\right)}_{\text{auto-regressive operator, } \mathbf{\Phi}(B)} \mathbf{Y}_t &= \mathbf{c} + \mathbf{e}_t \\
\mathbf{\Phi}(B) \mathbf{Y}_t &= \mathbf{c} + \mathbf{e}_t \\
\mathbf{Y}_t &= [\mathbf{\Phi}(B)]^{-1} \mathbf{c} + [\mathbf{\Phi}(B)]^{-1} \mathbf{e}_t
\end{aligned}
$$

VAR - Weak Stationarity

Weak Stationarity of VAR(p)

The VAR(p) model is covariance stationary if

$$\det[I_m - \mathbf{\Phi}_1 B - \mathbf{\Phi}_2 B^2 - \cdots - \mathbf{\Phi}_p B^p)] = 0$$

has roots outside $|B| < 1$ for complex B.

- Similar idea as in the univariate case "Claim"
- Time series that have time-invariant means, variances, and covariances

VAR - Weak Stationarity

- The mean of a weakly stationary VAR(p) model is

$$
\begin{aligned}
\mathrm{E}(\mathbf{Y}_t) = \boldsymbol{\mu}_Y &= \mathbf{c} + \boldsymbol{\Phi}_1 \mathrm{E}(\mathbf{Y}_{t-1}) + \boldsymbol{\Phi}_2 \mathrm{E}(\mathbf{Y}_{t-2}) + \cdots + \boldsymbol{\Phi}_p \mathrm{E}(\mathbf{Y}_{t-p}) + \mathrm{E}(\mathbf{e}_t) \\
\boldsymbol{\mu}_Y &= \mathbf{c} + \sum_{k=1}^{p} \boldsymbol{\Phi}_k \boldsymbol{\mu}_Y + \mathbf{0}_m \\
\boldsymbol{\mu}_Y &= \left(I_m - \sum_{k=1}^{p} \boldsymbol{\Phi}_k \right)^{-1} \mathbf{c}
\end{aligned}
$$

- Using $\mathbf{c} = \left(I_m - \sum_{k=1}^{p} \boldsymbol{\Phi}_k\right) \boldsymbol{\mu}_Y$, we can represent VAR($p$) model as

$$
\begin{aligned}
\mathbf{Y}_t &= \left(I_m - \sum_{k=1}^{p} \boldsymbol{\Phi}_k \right) \boldsymbol{\mu}_Y + \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \boldsymbol{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t \\
(\mathbf{Y}_t - \boldsymbol{\mu}_Y) &= \boldsymbol{\Phi}_1 (\mathbf{Y}_{t-1} - \boldsymbol{\mu}_Y) + \boldsymbol{\Phi}_2 (\mathbf{Y}_{t-2} - \boldsymbol{\mu}_Y) + \cdots + \boldsymbol{\Phi}_p (\mathbf{Y}_{t-p} - \boldsymbol{\mu}_Y) + \mathbf{e}_t \\
\tilde{\mathbf{Y}}_t &= \boldsymbol{\Phi}_1 \tilde{\mathbf{Y}}_{t-1} + \boldsymbol{\Phi}_2 \tilde{\mathbf{Y}}_{t-2} + \cdots + \boldsymbol{\Phi}_p \tilde{\mathbf{Y}}_{t-p} + \mathbf{e}_t
\end{aligned}
$$

  - This is a **mean-corrected** series

VAR - Nonstationary Series

- Spurious regression problem can arise with standard OLS when the time series are nonstationary
- Need different methodology for nonstationary time series
  - Tsay (2010) ch.8
  - Greene (2012) ch.21.3

- **Step-by-step approach**:
  1. If variables $\mathbf{Y}_t$ are I(0), we do not difference data, and estimate VAR in levels
  2. If variables $\mathbf{Y}_t$ are I(1), we first test them for **cointegration**
     2.1 If they are cointegrated, we estimate **vector error correction (VEC)** model
     2.2 If they are not cointegrated, we difference the data, and estimate a VAR model on first differences $\Delta\mathbf{Y}_t$

Cointegration

- An m-dimensional stochastic process $\{\mathbf{Y}_t\}$ is integrated of order d, $\mathrm{I}(d)$, if:
    1. $\mathbf{Y}_t$ is nonstationary
    2. the d-difference process $\Delta^d \mathbf{Y}_t = (1 - B)^d \mathbf{Y}_t$ is stationary
- *Linear combination* of component series of $\{\mathbf{Y}_t\}$:
    - If two series $(Y_{1,t}, Y_{2,t})$ are integrated to different orders, then linear combinations of them will be integrated to the **higher** of the two orders
    - Every component series of $\mathbf{Y}_t$ may be $\mathrm{I}(1)$, but the process may not be jointly integrated
    - Linear combinations of the component series (without any differencing) may be stationary
- If so, the multivariate time series $\mathbf{Y}_t$ is **cointegrated**

Cointegration

Cointegration (Engle and Granger, 1987, *Econometrica*)

- Consider $\{\mathbf{Y}_t\}$ where $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t}, \ldots, Y_{m,t})'$ is an m-vector of component time series
- $\{\mathbf{Y}_t\}$ is said to be **cointegrated** of order d, b, denoted by $\mathrm{CI}(d, b)$, if
  - each component is integrated of order d, i.e., $\mathrm{I}(d)$
  - there exists an m-vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_m)'$ such that

$$\boldsymbol{\beta}'\mathbf{Y}_t = \beta_1 Y_{1,t} + \beta_2 Y_{2,t} + \cdots + \beta_m Y_{m,t}$$

    is $I(d-b)$

- **Cointegration vector** β:
  - is not unique (the number of cointegrating vectors is called the *cointegrating rank* of $\mathbf{Y}_t$)
  - can be scaled arbitrarily, so assume a normalization $\boldsymbol{\beta} = (1, \beta_2, \ldots, \beta_m)'$
- Most nonstationary time series are I(1) and so CI(1, 1) is the most common case of cointegration

Cointegration

Intuition:

- The implication of cointegration is that the series $(Y_{1,t}, Y_{2,t}, \dots, Y_{m,t})'$ are **drifting together** (long-run relationship) at roughly the same rate
  - The drunk and her dog: the distance between the two is fairly predictable
- Economic variables are often related to each other through a long-run equilibrium relationship
- If this is the case, then differencing the individual series (ignoring the long-run relationship between the series) would be *counterproductive*
  - There may be more unit-root nonstationary components than the number of unit roots in a cointegrated system
  - Hence differencing individual components to achieve stationarity would result in **overdifferencing**

Cointegration

- Some examples of relationships predicted by economic theory
  - consumption and income (GDP)
  - real wages and labor productivity: $\frac{w}{p} = (1-\alpha)\frac{Y}{H}$
  - purchasing power parity: prices of the same good in two countries and the exchange rate: $p_t^* = e_t p_t$
  - money demand: stock of money, price level, real GDP, and nominal interest rate $\frac{M_t}{p_t} = L(Y_t, r_t)$
  - prices of same stock traded on two stock exchanges
- Driving force behind cointegration: variables share a *common stochastic trend*
  - Real wages and labor productivity both grow because of technological progress that affects both of them
- Testing for cointegration among component time series is then essentially checking whether a particular theory is consistent with data

Cointegration

- Consider a bivariate time series $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$, where $Y_{1,t}$ and $Y_{2,t}$ are both I(1)
- Assume

$$Y_{i,t} = c_i + \mu_{i,t} + e_{i,t} \qquad i = 1, 2$$

  - c_i is a constant
  - $\mu_{i,t}$ is the *stochastic trend* component
    - $\mu_{i,t} = \phi_i t$ (linear trend)
    - $\mu_{i,t} = \phi_i W_t$ where $W_t = W_{t-1} + \eta_t$ (random walk)
  - $e_{i,t}$ is some weakly stationary I(0) process
- $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$ is C(1,1) cointegrated if there exists vector $\boldsymbol{\beta} = (\beta_1, \beta_2)'$ such that

$$\beta_1 Y_{1,t} + \beta_2 Y_{2,t}$$

  is I(0), i.e., weakly stationary
  - We can set $\beta_1 = 1$, and hence $\boldsymbol{\beta} = (1, \beta)'$

Cointegration

- In particular,

$$(\beta_1 Y_{1,t} + \beta_2 Y_{2,t}) = (\beta_1 c_1 + c_2 \delta_2) + (\beta_1 \mu_{1,t} + \beta_2 \mu_{2,t}) + (\beta_1 e_{1,t} + \beta_2 e_{2,t})$$

  is weakly stationary if

$$\beta_1 \mu_{1,t} + \beta_2 \mu_{2,t} = 0 \implies \mu_{1,t} = -\frac{\beta_2}{\beta_1} \mu_{2,t}$$

- This implies that $Y_{1,t}$ and $Y_{2,t}$ must share the **common stochastic trend**, and cointegration vector $\boldsymbol{\beta} = (\beta_1, \beta_2)'$ removes this stochastic trend from the linear combination of $Y_{1,t}$ and $Y_{2,t}$

Cointegration Test - Engle-Granger Methodology

- **Step-by-step approach** (Engle and Granger, 1987, *Econometrica*):
  1. Given $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$, test whether variables $\mathbf{Y}_t$ are I(1)
  2. Estimate one of the models

$$Y_{1,t} = \beta Y_{2,t} + e_t$$
$$Y_{1,t} = \delta_0 + \beta Y_{2,t} + e_t$$
$$Y_{1,t} = \delta_0 + \delta_1 t + \beta Y_{2,t} + e_t$$

  3. Test residuals e_t for the presence of unit root
     - If we can not reject H_0 of unit root in residuals, we can not reject the H_0 that $Y_{1,t}$ and $Y_{2,t}$ are not cointegrated
     - If we reject H_0 of unit root in residuals, we can reject the H_0 that $Y_{1,t}$ and $Y_{2,t}$ are not cointegrated
- Methodology has several significant drawbacks
  - Exchanging $Y_{1,t}$ and $Y_{2,t}$ in the OLS may lead to contradictory results
  - No way to test for cointegrating rank
  - Greene (2011) ch.21.3 and ch.21.3.3

Cointegration Test - Johansen's Methodology

- **Step-by-step approach** (Johansen, 1991, *Econometrica*):
  1. Given $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$, test whether variables $\mathbf{Y}_t$ are I(1)
  2. Specify and estimate a VAR(p) model for $\mathbf{Y}_t$ (in levels, not in differences)

$$\mathbf{Y}_t = \mathbf{\Phi}_1 \mathbf{Y}_{t-1} + \mathbf{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \mathbf{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t$$

  3. Estimate a *vector error correction* model (using the VAR order from step 2)
  4. Cointegration test: determine the *number of cointegrating vectors* using trace and max eigenvalue tests

- We will next jump to step 3, assumming for now that the series $\mathbf{Y}_t$ is cointegrated

Vector Error Correction Model

- Consider a bivariate time series $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$
- Assume:
  - $Y_{1,t}$ and $Y_{2,t}$ are both I(1)
  - $\mathbf{Y}_t$ is CI(1,1) cointegrated with cointergation vector $\boldsymbol{\beta} = (1, \beta_2)'$ so that $Y_{1,t} + \beta_2 Y_{2,t}$ is I(0) - **this needs to be checked in step 4**
- Consider a bivariate VAR(1) model

$$\mathbf{Y}_t = \mathbf{\Phi}\mathbf{Y}_{t-1} + \mathbf{e}_t$$

or, alternatively,

$$\begin{aligned} Y_{1,t} &= \psi_{1,1} Y_{1,t-1} + \psi_{1,2} Y_{2,t-1} + e_{1,t} \\ Y_{2,t} &= \psi_{2,1} Y_{1,t-1} + \psi_{2,2} Y_{2,t-1} + e_{2,t} \end{aligned}$$

Vector Error Correction Model

- Substract $Y_{i,t-1}$, $i = 1, 2$, from both sides of equation i

$$\begin{aligned} \Delta Y_{1,t} &= -(1-\psi_{1,1})\left[Y_{1,t-1} - \frac{\psi_{1,2}}{1-\psi_{1,1}} Y_{2,t-1}\right] + e_{1,t} \\ \Delta Y_{2,t} &= \psi_{2,1}\left[Y_{1,t-1} - \frac{1-\psi_{2,2}}{\psi_{2,1}} Y_{2,t-1}\right] + e_{2,t} \end{aligned}$$

- The LHS variables $\Delta Y_{1,t}$ and $\Delta Y_{2,t}$ are both I(0)
- The RHS is I(0) only if:
  - $(1-\psi_{1,1}) = 0$ and $\psi_{2,1} = 0$; or
  - $-\frac{\psi_{1,2}}{1-\psi_{1,1}} = -\frac{\psi_{1,2}}{1-\psi_{1,1}} = \beta_2$
    - This yields I(0) due to the fact that $\mathbf{Y}_t$ is CI(1,1) cointegrated
    - Hence the series $\mathbf{Y}_t$ should be cointegrated to start with!

Vector Error Correction Model

Transitory VEC (Johansen, 1991, *Econometrica*)

- Given $Y_{1,t}$ and $Y_{2,t}$ are both I(1), and $\mathbf{Y}_t$ is CI(1,1) cointegrated with cointergation vector $\beta = (1, \beta_2)'$ so that $Y_{1,t} + \beta_2 Y_{2,t}$ is I(0), then a simple vector error correction model is

$$\begin{aligned} \Delta Y_{1,t} &= \alpha_1(Y_{1,t-1} + \beta_2 Y_{2,t-1}) + e_{1,t} \\ \Delta Y_{2,t} &= \alpha_2(Y_{1,t-1} + \beta_2 Y_{2,t-1}) + e_{2,t} \end{aligned}$$

- In matrix form,

$$\underset{2\times 1}{\Delta \mathbf{Y}_t} = \underset{2\times 2}{\mathbf{\Pi}}\, \underset{2\times 1}{\mathbf{Y}_{t-1}} + \underset{2\times 1}{\mathbf{e}_t}$$

  where

$$\Delta \mathbf{Y}_t = \begin{bmatrix} \Delta Y_{1,t} \\ \Delta Y_{2,t} \end{bmatrix} \quad \mathbf{\Pi} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \begin{bmatrix} 1 & \beta_2 \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_1\beta_2 \\ \alpha_2 & \alpha_2\beta_2 \end{bmatrix} \quad \mathbf{e}_t = \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix}$$

  - $\mathbf{\Pi Y}_{t-1}$ is the **error correction term**
  - α_1 and α_2 are the *adjustment parameters*

Vector Error Correction Model

Notes:

- VEC model allows to study the short-run dynamics in the relationship between component series
- If the two variables $(Y_{1,t}, Y_{2,t})$ are related through a long-run equilibrium relationship (i.e., common stochastic trend), then the error correction term $(Y_{1,t-1} + \beta_2 Y_{2,t-1})$ controls for the differences between $Y_{1,t-1}$ and $\beta_2 Y_{2,t-1}$ so that the realizations of $Y_{1,t}$ and $Y_{2,t}$ do not deviate from the equilibrium
- The model is reasonable if the two variables $(Y_{1,t}, Y_{2,t})$ are cointegrated
  - If not, then $(Y_{1,t-1} + \beta_2 Y_{2,t-1})$, and hence the RHS, cannot be I(0)
- If $\alpha_i = 0, \alpha_j \neq 0$, then $Y_{i,t}$ is a pure random walk and all the adjustment occurs in $Y_{j,t}$

Vector Error Correction Model

Let us obtain VEC form for m-variate VAR(p) model

- Consider $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t}, \ldots, Y_{m,t})'$ where component time series are I(1) and $\mathbf{Y}_t$ follows VAR(p)

$$\mathbf{Y}_t = \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \boldsymbol{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t$$

- Add and substract $\boldsymbol{\Phi}_p \mathbf{Y}_{t-p+1}$ on LHS

$$\begin{aligned}
\mathbf{Y}_t &= \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \boldsymbol{\Phi}_{p-1} \mathbf{Y}_{t-p+1} + \boldsymbol{\Phi}_p \mathbf{Y}_{t-p+1} - \boldsymbol{\Phi}_p \mathbf{Y}_{t-p+1} + \boldsymbol{\Phi}_p \mathbf{Y}_{t-p} + \mathbf{e}_t \\
&= \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+1} - \boldsymbol{\Phi}_p \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t
\end{aligned}$$

- Add and substract $(\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+2}$ on LHS

$$\begin{aligned}
\mathbf{Y}_t &= \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + \boldsymbol{\Phi}_{p-2} \mathbf{Y}_{t-p+2} + (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+2} \\
&\quad - (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+2} + (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+1} - \boldsymbol{\Phi}_p \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t \\
&= \boldsymbol{\Phi}_1 \mathbf{Y}_{t-1} + \boldsymbol{\Phi}_2 \mathbf{Y}_{t-2} + \cdots + (\boldsymbol{\Phi}_{p-2} + \boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+2} \\
&\quad - (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \Delta \mathbf{Y}_{t-p+2} - \boldsymbol{\Phi}_p \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t
\end{aligned}$$

- Add and substract $(\boldsymbol{\Phi}_{p-i+1} + \cdots + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-p+i}$ for $i = 1, \ldots, p-1$ on LHS

$$\begin{aligned}
\mathbf{Y}_t &= (\boldsymbol{\Phi}_1 + \cdots + \boldsymbol{\Phi}_p) \mathbf{Y}_{t-1} - (\boldsymbol{\Phi}_2 + \cdots + \boldsymbol{\Phi}_p) \Delta \mathbf{Y}_{t-1} - \cdots \\
&\quad - (\boldsymbol{\Phi}_{p-1} + \boldsymbol{\Phi}_p) \Delta \mathbf{Y}_{t-p+2} - \boldsymbol{\Phi}_p \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t
\end{aligned}$$

Vector Error Correction Model

- Finally, substract $\mathbf{Y}_{t-1}$ from both sides

$$\begin{aligned}\Delta \mathbf{Y}_t &= (\mathbf{\Phi}_1 + \cdots + \mathbf{\Phi}_p - I_m)\mathbf{Y}_{t-1} - (\mathbf{\Phi}_2 + \cdots + \mathbf{\Phi}_p)\Delta \mathbf{Y}_{t-1} - \cdots \\ &\quad -(\mathbf{\Phi}_{p-1} + \mathbf{\Phi}_p)\Delta \mathbf{Y}_{t-p+2} - \mathbf{\Phi}_p \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t\end{aligned}$$

- More compactly,

$$\Delta \mathbf{Y}_t = \mathbf{\Pi} \mathbf{Y}_{t-1} + \mathbf{\Gamma}_1 \Delta \mathbf{Y}_{t-1} + \cdots + \mathbf{\Gamma}_{p-1} \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t$$

  - The model can be augmented with a deterministic term $\mathbf{c} = \mathbf{c}_0 + \mathbf{c}_1 t$ and/or other I(0) exogenous variables ($\mathbf{X}_t$)
  - If $\mathbf{\Pi} = \mathbf{0}$, VEC model becomes a reduced form VAR(p) model estimated on differenced data
  - If $\mathbf{\Pi}$ contains non-zero elements, then estimating a VAR on differenced data leads to **omitted variable bias** (hence the name)

Granger Representation Theorem

For any set of I(1) variables ($Y_{1,t}, Y_{2,t}, \ldots, Y_{m,t}$), error correction representation exists if and only if they are cointegrated

Cointegration Test - Johansen's Methodology

Cointegration Test (Step 4)

- The matrix $\mathbf{\Pi}$ produces linear combinations of the variables $\mathbf{Y}_{t-1}$
- Since $\mathbf{\Pi}$ has as many linearly independent rows as there are cointegrating vectors β, it is possible to test for cointegration using the rank of matrix $\mathbf{\Pi}$
  1. If $\text{rank}(\mathbf{\Pi}) = 0$, then $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t}, \dots, Y_{m,t})'$ are I(1), but not cointegrated
     - Main equation reduces to

     $$\Delta \mathbf{Y}_t = \mathbf{\Gamma}_1 \Delta \mathbf{Y}_{t-1} + \cdots + \mathbf{\Gamma}_{p-1} \Delta \mathbf{Y}_{t-p+1} + \mathbf{e}_t$$

     so that $\Delta \mathbf{Y}_t$ follows a VAR$(p-1)$
  2. If $0 < \text{rank}(\mathbf{\Pi}) < m$, then $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t}, \dots, Y_{m,t})'$ are cointegrated with $r = \text{rank}(\mathbf{\Pi})$ linearly independent long-run relationships
  3. If $\text{rank}(\mathbf{\Pi}) = m$, then $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t}, \dots, Y_{m,t})'$ must be I(0) and there is no cointegration among them

Cointegration Test - Johansen's Methodology

Example (theory):

- Recall the bivariate case

$$\Delta \mathbf{Y}_t = \mathbf{\Pi} \mathbf{Y}_{t-1} + \mathbf{e}_t$$

where

$$\mathbf{Y}_t = \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix} \quad \mathbf{\Pi} = \begin{bmatrix} \alpha_1 & \alpha_1\beta_2 \\ \alpha_2 & \alpha_2\beta_2 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \begin{bmatrix} 1 & \beta_2 \end{bmatrix} \quad \mathbf{e}_t = \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix}$$

- Since the two rows of $\mathbf{\Pi}$ are linearly depended,

$$\text{rank}(\mathbf{\Pi}) = 1 \implies \begin{array}{c} \text{there exists a single} \\ \text{cointegrating vector} \\ \boldsymbol{\beta} = (1, \beta_2)' \end{array}$$

Cointegration Test - Johansen's Methodology

How do we verify this empirically?

- Let $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_m$ be the estimated eigenvalues of $\underset{m\times m}{\mathbf{\Pi}}$
- If $\text{rank}(\mathbf{\Pi}) = r$, then $\lambda_{r+1} > \cdots > \hat{\lambda}_m$ should be small (close to 0)
- Formal test statistics:
  1. **Trace statistic**: $H_0 : \text{rank}(\mathbf{\Pi}) = r$ vs. $H_1 : \text{rank}(\mathbf{\Pi}) > r$

$$LR_{\text{trace}}(r) = -T \sum_{i=r+1}^{m} \log(1 - \hat{\lambda}_i), \quad r = 0, 1, \ldots, m-1$$

  2. **Maximum eigenvalue statistic**: $H_0 : \text{rank}(\mathbf{\Pi}) = r$ vs. $H_1 : \text{rank}(\mathbf{\Pi}) = r + 1$

$$LR_{\max}(r) = -T \log(1 - \hat{\lambda}_{r+1}), \quad r = 0, 1, \ldots, m-1$$

- For each of the two tests we follow a sequential procedure (step 1: $r = 0$, step 2: $r = 1$, ...)
- Results of trace and max eigenvalue tests may be contradictory
  - If that happens, max eigenvalue test is usually prioritized

VEC - Implementation in R

urca package

- Cointegration tests and estimation tools for multivariate time series data
- `ca.jo()`: Johansen's cointegration test, `type={"trace","eigen"}`
- `cajorls()`: OLS regression of VECM
- Data on money supply for Denmark, 1974Q1 - 1987Q3, given by four-component series

$$\mathbf{Y}_t = \left(\log\left(\frac{M2_t}{P_t}\right), \log(I_t), i_t^b, i_t^d \right)'$$

  - $\log\left(\frac{M2_t}{P_t}\right)$ is log of money supply M2 deflated by price index
  - $\log(I_t)$ is log real income
  - i_t^b is bond rate
  - i_t^d is deposit rate
- Based on unit root tests all series appear to be I(1)

VEC - Example

```r
# EXAMPLE - MONEY SUPPLY FOR DENMARK (1974Q1 - 1987Q3)
library(urca)
data(denmark)
head(denmark)
y<-zoo(denmark[, c("LRM","LRY","IBO","IDE")], order.by=as.yearqtr(denmark[,1], format="%Y:0%q"))
head(y)
plot(y)
plot(diff(y))

# UNIT ROOT TEST
library(tseries)
adf.test(y[,1], alternative="stationary", k=0)   #k=0: Dickey-Fuller test
adf.test(y[,1], alternative="stationary")
adf.test(diff(y[,1], differences=1, lag=1), alternative="stationary", k=0)
adf.test(diff(y[,1], differences=1, lag=1), alternative="stationary")

adf.test(y[,2], alternative="stationary", k=0)
adf.test(y[,2], alternative="stationary")
adf.test(diff(y[,2], differences=1, lag=1), alternative="stationary", k=0)
adf.test(diff(y[,2], differences=1, lag=1), alternative="stationary")

adf.test(y[,3], alternative="stationary", k=0)
adf.test(y[,3], alternative="stationary")
adf.test(diff(y[,3], differences=1, lag=1), alternative="stationary", k=0)
adf.test(diff(y[,3], differences=1, lag=1), alternative="stationary")
```

VEC - Example

```
adf.test(y[,4], alternative="stationary", k=0)
adf.test(y[,4], alternative="stationary")
adf.test(diff(y[,4], differences=1, lag=1), alternative="stationary", k=0)
adf.test(diff(y[,4], differences=1, lag=1), alternative="stationary")

# VAR MODEL ORDER SELECTION
library(vars)
ord.choice.y<-VARselect(y,
                      lag.max=5,
                      type="const")
print(ord.choice.y)

# COINTEGRATION TEST: TRACE STATISTIC
y.co.ts<-ca.jo(y,
               type="trace",
               ecdet="const",      #includes constant term
               K=2,                #lag order of the series in the VAR
               spec="transitory", #specified Gamma parameter vector as in the class
               season=4)           #includes seasonal dummies
summary(y.co.ts)

# COINTEGRATION TEST: MAXIMUM EIGENVALUE STATISTIC
y.co.me<-ca.jo(y,
               type="eigen",
               ecdet="const",
               K=2,
               spec="transitory",
               season=4)
summary(y.co.me)
```

VEC - Example

```r
# VEC MODEL ESTIMATION GIVEN COINTEGRATION RANK FROM COINTEGRATION TEST
y.VEC<-cajorls(y.co.me, r=1)
print(y.VEC)
summary(y.VEC$rlm)

# PREDICTION
y.VAR<-vec2var(y.co.me, r=1)
y.VAR.fcst<-predict(y.VAR, n.ahead=8)
plot(y.VAR.fcst)
```

VEC - Example

Notes:

- Cointegration test
  - Trace test statistic suggests that we cannot reject $H_0 : \text{rank}(\mathbf{\Pi}) = 0$ $\implies$ $\mathbf{Y}_t$ are not cointegrated
  - Maximum eigenvalue test however suggests that we can first reject $H_0 : \text{rank}(\mathbf{\Pi}) = 0$ and afterwards cannot reject $H_0 : \text{rank}(\mathbf{\Pi}) = 1$ $\implies$ $\mathbf{Y}_t$ are cointegrated with one cointegrating relationship
- Estimation
  - `ect`: parameter estimate of the error correction term, $\hat{\mathbf{\Pi}} = \begin{bmatrix} \hat{\alpha}_1 \\ \hat{\alpha}_2 \end{bmatrix} \begin{bmatrix} 1 & \hat{\beta}_2 \end{bmatrix}$
  - `sd`: seasonal dummies
  - `$beta`: cointegrating vector
- Forecast
  - To construct forecasts, we need to first transform the estimated VEC model in differences into a VAR in levels (`vec2var()`)

Extensions

- Nonparametric time series analysis
- Bayesian time series analysis
- Copula modeling of dependence in time series analysis
- State-space models (Kallman filer)