Forecasting Book Club

Chapter 5: Box-Jenkins ARIMA Models

Dr Shixuan Wang

Department of Economics
University of Reading

Oct 6th, 2020

Shixuan ≈ Sichuan

My Presentation Style

Combination of Three Things

- What is in the book? (for practitioners)
- What is my reflection? (some statistical thoughts)
- R demo.

Outline

Outline for section 1

Stationarity

Definition (Stationarity)

Stationary time series are sales histories that have an underlying structure that does not change over time.

Assessing stationarity

We can gain an initial idea by looking at the graph of the series.

Stationarity

Definition (Weak Stationarity)

For a time series x_t to be **weakly stationary** (or covariance stationary):

1. Mean of X_t is constant

$$\mathbb{E}(x_t) = \mathbb{E}(x_{t-1}) = ... = \mathbb{E}(x_1) = \mu$$

2. Variance of X_t is constant

$$\mathbb{V}(x_t) = \mathbb{V}(x_{t-1}) = ... = \mathbb{V}(x_1) = \sigma^2$$

3. Autocovariance only depends on the lag k and not time t

$$Cov(x_1, x_{1+k}) = Cov(x_2, x_{2+k}) = ... = Cov(x_{n-k}, x_n) = \gamma_k$$

Definition (Strict Stationarity)

A time series x_t is called be **strictly stationary**:

- if for any integer k and $s \geq 1$, the multivariate joint distribution of $(x_1, ..., x_s)$ is the same as $(x_{1+k}, ..., x_{s+k})$.

Statistical Tests on Stationarity

Unit root tests

- Augmented Dickey-Fuller (ADF) Test.
- Phillips-Perron (PP) Test.

By rejecting the ADF or PP tests, you end up with stationary series.

Stationarity Test

- KPSS test (Kwaitowski, Phillips, Schmidt and Shin, 1992).

By not rejecting the KPSS test, you end up with stationary series.

R Demo

White noises

$$\varepsilon_t \sim N(0, \sigma^2)$$

Using R!

```
e <- rnorm(1000, mean = 0, sd = 1)
plot(e,type='l')
hist(e,20)
acf(e)
```

The deterministic trend process

$$x_t = \alpha + \beta t + \varepsilon_t$$

where t denotes the linear time trend, and ε_t is the white noise with variance σ^2.

R Demo (Cont'd)

Using R!

```r
# Set the parameters
TT    <- 100
alpha <- 0
beta  <- 0.2
# Simulate Deterministic Trend
e     <- rnorm(TT, mean=0, sd=1)
trend <- 1:TT
x     <- alpha+beta*trend+e
# Plot
plot(x,type='l',xlab="",ylab="")
grid()
acf(x)
```

R Demo (Cont'd)

Structural break (change-point) in the varaince

$$x_t = \begin{cases} \alpha + \varepsilon_t & t < t_c \\ \alpha + \sqrt{2}\varepsilon_t & t \geq t_c \end{cases}$$

where t_c is time of change, and ε_t is the white noise with variance σ^2.

R Demo (Cont'd)

Using R!

```r
NN <- 5
TT <- 100
X  <- matrix(NA,TT,NN)

for (i in 1:NN){
  e.part1 <- rnorm(TT/2, mean=0, sd=1)
  e.part2 <- rnorm(TT/2, mean=0, sd=2)
  X[,i] <-  c(e.part1 , e.part2)
}

matplot(X,type='l',xlab="",ylab="")
grid()
```

Outline for section 2

Autoregressive Models

AR(1)

$$Sales_t = a + bSales_{t-1} + \varepsilon_t$$

where a, b are parameters and ε_t is the noise.

AR(2)

$$Sales_t = a + b_1 Sales_{t-1} + b_2 Sales_{t-2} + \varepsilon_t$$

where a, b_1, b_2 are parameters and ε_t is the noise.

ACF and PACF

ACF of AR

AR models have geometrically decaying ACF.

PACF of AR

AR(p) models have PACF truncated at p.

Generating an AR(1) Process

Using R!

```r
ar1 <- arima.sim(list(order=c(1,0,0),ar=0.5),n=100000)
plot(ar1,type='l')
acf(ar1)
pacf(ar1)
```

Generating an AR(2) Process

Using R!

```r
ar2 <- arima.sim(list(order=c(2,0,0),ar=c(-0.2,0.35)),n=100000)
plot(ar2,type='l')
acf(ar2)
pacf(ar2)
```

Outline for section 3

Moving Average Models

MA(1)

$$Sales_t = a + b\varepsilon_{t-1} + \varepsilon_t$$

where a, b are parameters and ε_t is the noise.

MA(2)

$$Sales_t = a + b_1\varepsilon_{t-1} + b_2\varepsilon_{t-2} + \varepsilon_t$$

where a, b_1, b_2 are parameters and ε_t is the noise.

ACF and PACF

ACF of MA

MA(p) models have ACF truncated at p.

PACF of MA

MA models have geometrically decaying PACF.

Generating an MA(1) Process

Using R!

```
ma1 <- arima.sim(list(order=c(0,0,1),ma=0.5),n=100000)
plot(ma1,type='l')
acf(ma1)
pacf(ma1)
```

Generating an MA(2) Process

Using R!

```
ma2 <- arima.sim(list(order=c(0,0,2),ma=c(0.3,0.1)),n=100000)
plot(ma2,type='l')
acf(ma2)
pacf(ma2)
```

Outline for section 4

Autoregressive Moving Average Models

ARMA(1,1)

$$Sales_t = a + b_1 Sales_{t-1} + b_2 \varepsilon_{t-1} + \varepsilon_t$$

where a, b_1, b_2 are parameters and ε_t is the noise.

ACF and PACF

ACF of ARMA

ARMA models have geometrically decaying ACF.

PACF of ARMA

ARMA models have geometrically decaying PACF.

Generating an ARMA(1,1) Process

Using R!

```
# Set the parameters
phi <- 0.9
theta <- 0.3

# Simulate many trajectories of ARMA(1,1) and calculate ACF
my.model <- list(order=c(1,0,1),ar=phi,ma=theta)
arma11 <- arima.sim(my.model,n=100000)
plot(arma11,type='l')
acf(arma11)
pacf(arma11)
```

Outline for section 5

Data

- Name: Growth Rate of Gross Domestic Product
- Seasonally adjusted: yes
- Frequency: quarterly
- Period: 1955Q2 – 2018Q3
- Unit: £million
- Source: UK Office for National Statistics
- URL: `https://www.ons.gov.uk/economy/grossdomesticproductgdp/timeseries/abmi/pn2`

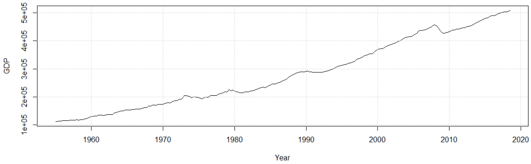
Histogram plot

Using R!

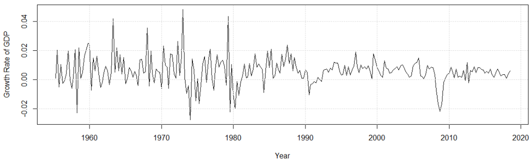
```
Raw.Data <- read_excel("UK GDP Quarterly.xls")
Raw.Date  <- as.yearqtr(Raw.Data$TIME)
Date <- Raw.Date[-1]
GDP <- Raw.Data$Value
LGDP <- log(GDP)
DLGDP <- diff(LGDP)

plot(Raw.Date, GDP, type='l')
plot(Date, DLGDP, type='l',ylab='Growth Rate of GDP', xlab='Time')
```

Histogram plot (Cont'd)

Original Data of UK GDP



Growth Rate of UK GDP (i.e. $\Delta \log(GDP)$)

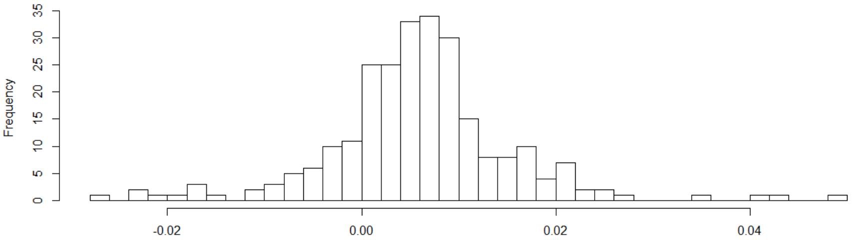


Histogram plot

Using R!

```
hist(DLGDP,50, main="", xlab="")
```

Histogram of Growth Rate

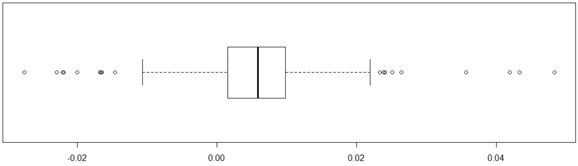


Box plot

Using R!

```
boxplot(DLGDP, horizontal=TRUE)
```

Box plot of Growth Rate



Model Selection by ACF and PACF

This is a judgmental call.

ACF

- AR models have geometrically decaying ACF.
- MA models have truncating ACF.
- ARMA models have geometrically decaying ACF.

PACF

- AR models have truncating PACF.
- MA models have geometrically decaying PACF.
- ARMA models have geometrically decaying PACF.

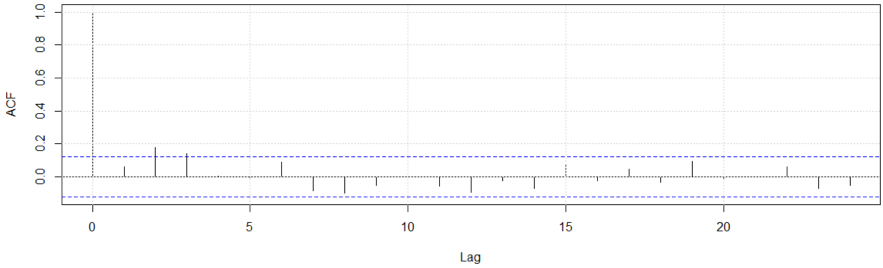
I personally find selecting model by ACF and PACF is difficult to use in practice.

ACF Plot

Using R!

```
acf(DLGDP, 24, main="")
```

ACF Plot of Growth Rate

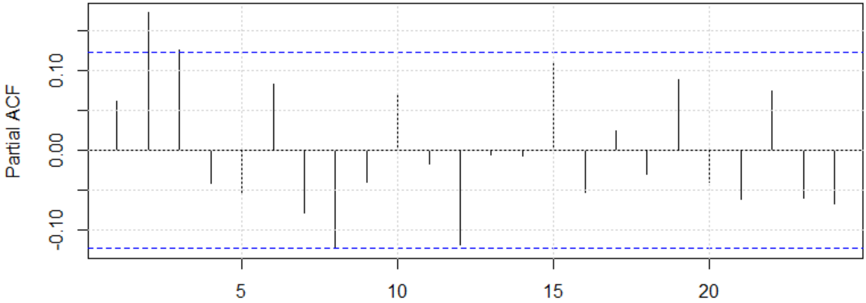


PACF Plot

Using R!

```
pacf(DLGDP, 24, main="")
```

ACF Plot of Growth Rate



Information Criteria for Model Selection

Two most popular information criteria

The two most popular criteria are Akaike's (1974) information criterion (AIC), and Schwarz's (1978) Bayesian information criterion (BIC or SBIC).

$$AIC = -2\log\mathcal{L}(\hat{\theta}) + 2k$$
$$BIC = -2\log\mathcal{L}(\hat{\theta}) + k\log(N)$$

where $\log\mathcal{L}(\hat{\theta})$ denotes the value of the maximized log-likelihood objective function for a model, k is the number of parameters ($k = p + q + 2$ in R for ARMA because it counts the intercept and the variance of the residuals as two additional parameters), and N is the sample size (number of observations).

Information Criteria for Model Selection (Cont'd)

Which IC should be preferred if they suggest different model orders?

- *BIC* embodies a stiffer penalty term than *AIC*.
- *AIC* is not consistent, and will typically pick "bigger" models.
- *BIC* is strongly consistent but (inefficient), and will pick "smaller" models.

Key Point

Smaller IC $\Rightarrow$ Better Model

Model Selection by AIC

Using R!

```
# Set the all possible lags of p and q. In my case, p and q can be 0,1,2,3,4,5.
pLag <- 0:5
qLag <- 0:5
np   <- length(pLag)
nq   <- length(qLag)

# Make a container for storing all AIC
IC   <- matrix(NA, np, nq)

# Caluclate AIC for all possible models and store then in the container
for (i in 1:np){
  for (j in 1:nq){
    p <- pLag[i]
    q <- qLag[j]

    ifit    <- arima(DLGDP, order=c(p,0,q))
    IC[i,j] <- AIC(ifit)
  }
}

# Find the minimum AIC and the corresponding best lags of p and q
idx <- which(IC == min(IC), arr.ind = TRUE)
best.p=idx[1]-1
best.q=idx[2]-1
```

Model Selection by AIC (Cont'd)

Include intercept in ARMA?

- In my estimation, an intercept is included in the ARMA model.
- This is because the average UK GDP growth rate is non-zero.
- Alternatively, you can firstly de-mean the data and then estimate the ARMA model without the intercept.

AIC of all possible ARMA(p,q)

p\q	**0**	**1**	**2**	**3**	**4**	**5**
0	-1633.30	-1632.02	-1637.86	-1639.88	-1638.21	-1636.51
1	-1632.26	-1635.38	-1639.20	-1638.13	-1637.81	-1635.55
2	-1637.93	-1638.15	-1637.29	-1637.50	-1635.87	-1633.85
3	-1639.92	-1638.13	-1645.31	**-1645.38**	-1639.02	-1634.76
4	-1638.36	-1637.76	-1644.45	-1638.81	-1641.42	-1643.35
5	-1637.08	-1636.44	-1643.73	-1642.05	-1641.61	-1642.57

The minimum AIC (i.e. the most negative) among the all possible 36 models is -1645.38, which corresponds to ARMA(3,3).

Model Diagnostics

What is the ideal situation?

- All coefficients should be statistically significant.
- In theory, the residuals from the ARMA model should be close to the white noise, which is i.i.d. without any autocorrelation.
- If there is any remaining (significant) autocorrelation in the residuals, this may suggest the need for additional AR or MA terms.

Check 1: Are the coefficients significant?

Using R!

```r
fit.best <- arima(DLGDP, order=c(best.p,0,best.q))
coeftest(fit.best)
```

```
            Estimate  Std. Error z value  Pr(>|z|)
ar1        0.48326145  0.21383995  2.2599 0.0238261 *
ar2       -0.67298959  0.07982496 -8.4308 < 2.2e-16 ***
ar3        0.52058855  0.15144350  3.4375 0.0005871 ***
ma1       -0.43545678  0.22258086 -1.9564 0.0504183 .
ma2        0.88082538  0.04981684 17.6813 < 2.2e-16 ***
ma3       -0.48757189  0.20010116 -2.4366 0.0148250 *
intercept  0.00596099  0.00082367  7.2371 4.585e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Check 2: Overfitting - Should we use a more complex model?

Using R!

```
overfit <- arima(DLGDP, order=c(3,0,4))
coeftest(overfit)
```

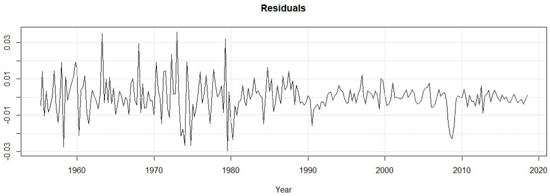
```
           Estimate   Std. Error z value  Pr(>|z|)
ar1       -0.32595272 0.29441426 -1.1071  0.268241
ar2       -0.60991385 0.23056321 -2.6453  0.008161 **
ar3        0.46213848 0.28443567  1.6248  0.104215
ma1        0.37149540 0.29533189  1.2579  0.208431
ma2        0.81960371 0.26765540  3.0622  0.002197 **
ma3       -0.25902203 0.31952943 -0.8106  0.417575
ma4        0.14889397 0.07791786  1.9109  0.056016 .
intercept  0.00598080 0.00081996  7.2940 3.009e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Check 3: Are the residuals white noise? - Visualization on the Residuals.

The time series plot of the residuals should be inspected for any obvious departures from white noise.

Using R!

```r
my.resid <- resid(fit.best)
plot(Year1, my.resid, type='l', xlab="Year", ylab="", main="Residuals")
```



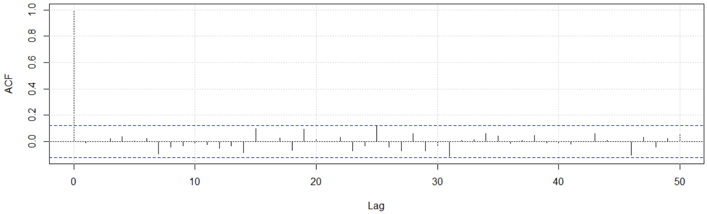
But only relying on the time series plot is not enough to ensure that the residuals are uncorrelated.

Check 3: Are the residuals white noise? - ACF Plot of Residuals

We could also inspect the sample autocorrelations of the residuals, $\hat{\rho}_{\varepsilon}(h)$, for any patterns or large values.

Using R!

```
acf(my.resid,50, main="")
```



The ACF plot of the residuals shows no significant autocorrelation because all sample autocorrelations of the residuals (the bars) are within the critical level (the blue dash line).

Check 3: Are the residuals white noise? - Ljung-Box Test

The hypothesis of Ljung-Box Test

H_0: The data are independently distributed.
H_1: The data are not independently distributed; they exhibit serial correlation.

Ljung-Box Q-Statistics

We can perform a general test that takes into consideration the magnitudes of $\hat{\rho}_\varepsilon(h)$ as a group. The intuition is that it may be the case that, individually, each $\hat{\rho}_\varepsilon(h)$ is small in magnitude, but, collectively, the values are large. The Ljung-Box Q-statistics is given by

$$Q = N(N+2)\sum_{h=1}^{H}\frac{\hat{\rho}_\varepsilon^2(h)}{N-h}$$

where H is chosen somewhat arbitrarily (typical values $H = 20$), N is the sample size, and $\hat{\rho}_\varepsilon(h)$ are sample autocorrelations of the residuals.

- Under the null hypothesis, $Q \sim \chi^2_{H-p-q}$
- Thus, we should reject the null hypothesis at level α (such as 5%) if the value of Q-statistics exceeds the $(1-\alpha)$-quantile of the χ^2_{H-p-q} distribution.
- Alternatively, we can simply use the p-value, which is the probability of the null hypothesis. Practically, if p-value is less than the significance level α (such as 5%), then we should reject the null hypothesis.

Check 3: Are the residuals white noise? - Ljung-Box Test

Using R!

```
Box.test(my.resid,type="Ljung-Box",lag=10, fitdf=best.p+best.q)
Box.test(my.resid,type="Ljung-Box",lag=20, fitdf=best.p+best.q)
Box.test(my.resid,type="Ljung-Box",lag=30, fitdf=best.p+best.q)
```

```
> Box.test(my.resid,type="Ljung-Box",lag=10, fitdf=best.p+best.q)

        Box-Ljung test

data:  my.resid
X-squared = 3.9331, df = 4, p-value = 0.4151

> Box.test(my.resid,type="Ljung-Box",lag=20, fitdf=best.p+best.q)

        Box-Ljung test

data:  my.resid
X-squared = 13.606, df = 14, p-value = 0.4795

> Box.test(my.resid,type="Ljung-Box",lag=30, fitdf=best.p+best.q)

        Box-Ljung test

data:  my.resid
X-squared = 24.268, df = 24, p-value = 0.4464
```

Check 3: Are the residuals white noise? - Ljung-Box Test (Cont'd)

How to interpret the results of Ljung-Box test?

- We can simply use the p-values.
- As can be seen, the p-values at $H = 10, 20, 30$ are 0.4151, 0.4795, and 0.4464, respectively. They are all above the significance level 5%.
- Hence, we cannot reject the null hypothesis that the data are independently distributed.
- We can conclude that there is no autocorrelation in the residuals of our ARMA(3,3) model.

Check 4: Are the residuals normally distributed?

We can use

- QQ plot.
- Jarque-Bera test.

But nowadays, we actually do not necessarily need the normality assumption. Because we can assume non-normal distribution for the error term.

Outline for section 6

Nonstationary Time Series

Nonastationary in mean: difference

you simply model the differences between sales in successive periods - or, if necessary, the differences of the differences.

Nonastationary in variance: Box-Cox power transformation

We need to transform the series in the hope that the resulting new series will be stable in their variability over time.

$$x_t = \frac{x_t^{\lambda} - 1}{\lambda}$$

where $\lambda > 0$.

Relation of ARMA(p,q) with Other Processes

ARMA(p,q) has the most general form:

$$\Phi(L)x_t = \Theta(L)\varepsilon_t$$

where $\Phi(L) = 1 - \phi_1 L - \phi_2 L^2 - ... - \phi_p L^p$ and $\Theta(L) = 1 + \theta_1 L + \theta_2 L^2 + ... + \theta_q L^q$

Other Processes are special cases of ARMA(p,q):

- If $\Phi(L) = 1 - \phi_1 L - \phi_2 L^2 - ... - \phi_p L^p$ and $\Theta(L) = 1 \implies$ AR(p)
- If $\Phi(L) = 1$ and $\Theta(L) = 1 + \theta_1 L + \theta_2 L^2 + ... + \theta_q L^q \implies$ MA(q)
- If $\Phi(L) = 1 - \phi_1 L$ and $\Theta(L) = 1 + \theta_1 L \implies$ ARMA(1,1)
- If $\Phi(L) = 1 - \phi_1 L$ and $\Theta(L) = 1 \implies$ AR(1)
- If $\Phi(L) = 1$ and $\Theta(L) = 1 + \theta_1 L \implies$ MA(1)

Autoregressive Integrated Moving Average Process of Order (p,d,q), ARIMA(p,d,q)

Definition (ARIMA(p,d,q))

A process x_t is said to be ARIMA(p,d,q) if

$$\Delta^d x_t = (1-L)^d x_t$$

is ARMA(p,q). In general, we will write the model as

$$\Phi(L)\Delta^d x_t = \Theta(L)\varepsilon_t$$

If $\mathbb{E}(\Delta^d x_t)$ is not zero, we write the model as

$$\Phi(L)\Delta^d x_t = \alpha + \Theta(L)\varepsilon_t$$

Outline for section 7

Seasonal ARIMA Model: $ARIMA(p, d, q) \times (P, D, Q)_m$

$ARIMA(p, d, q) \times (P, D, Q)_m$

- P = the number of seasonal autoregressive terms,
- D = the number of seasonal differencing,
- Q = the number of seasonal moving average terms.
- m = the number of periods per year.

$ARIMA(0, 0, 0) \times (1, 0, 0)_{12}$

$$x_t = a + bx_{t-12} + \varepsilon_t$$

where a, b are parameters and ε_t is the noise.

$ARIMA(1, 1, 0) \times (0, 0, 1)_{12}$

$$\Delta x_t = a + b_1 \Delta x_{t-1} + b_2 \varepsilon_{t-12} + \varepsilon_t$$

where a, b_1, b_2 are parameters and ε_t is the noise.

Seasonal ARIMA Model: $ARIMA(p, d, q) \times (P, D, Q)_m$ (Cont'd)

$ARIMA(0, 1, 1) \times (0, 1, 1)_{12}$

$$y_t = \Delta x_t = x_t - x_{t-1}$$

$$z_t = \Delta^{12} y_t = y_t - y_{t-12}$$

$$z_t = a + b_1 \varepsilon_{t-1} + b_2 \varepsilon_{t-12} + \varepsilon_t$$

where a, b_1, b_2 are parameters and ε_t is the noise.

Generating an $ARIMA(1,1,1) \times (1,1,1)_4$ Process

Using R!

```
library(forecast)
model <- Arima(ts(rnorm(100),freq=4), order=c(1,1,1), seasonal=c(1,1,1),
fixed=c(phi=0.5, theta=-0.4, Phi=0.3, Theta=-0.2))
sarima <- simulate(model, nsim=1000)
plot(sarima,type='l')
acf(sarima)
pacf(sarima)
```

source: https://robjhyndman.com/hyndsight/simulating-from-a-specified-seasonal-arima-model/

E-mail: *shixuan.wang@reading.ac.uk*