Forecasting: principles and practice

Rob J Hyndman

2 ARIMA models
1 Stationarity and differencing
2 Backshift notation
3 Autoregressive models
4 Moving Average models
5 Non-seasonal ARIMA models
6 Seasonal ARIMA models
7 Lab Session 3
Stationarity

**Definition**

If \( \{ y_t \} \) is a stationary time series, then for all \( s \), the distribution of \( (y_t, \ldots, y_{t+s}) \) does not depend on \( t \).
Stationarity

**Definition**
If \( \{ y_t \} \) is a stationary time series, then for all \( s \), the distribution of \( (y_t, \ldots, y_{t+s}) \) does not depend on \( t \).

A **stationary series** is:
- roughly horizontal
- constant variance
- no patterns predictable in the long-term
Stationarity

**Definition**

If \( \{y_t\} \) is a stationary time series, then for all \( s \), the distribution of \((y_t, \ldots, y_{t+s})\) does not depend on \( t \).

A stationary series is:
- roughly horizontal
- constant variance
- no patterns predictable in the long-term

Transformations (e.g., logs) can help to **stabilize the variance**.

Differences can help to **stabilize the mean**.
Stationary?

dj %>% autoplot() +
  ylab("Dow Jones Index") + xlab("Day")
Stationary?

dj %>% diff() %>% autoplot() + ylab("Change in Dow Jones Index") + xlab("Day")
Stationary?

```r
hsales %>% autoplot() +
  xlab("Year") + ylab("Total sales") +
  ggtitle("Sales of new one-family houses, USA")
```
Stationary?

```r
hsales %>% diff(lag=12) %>% autoplot() +
  xlab("Year") + ylab("Total sales") +
  ggtitle("Seasonal differences of sales of new one-family houses, USA")
```
Stationary?

```r
hsales %>% diff(lag=12) %>% diff(lag=1) %>% autoplot() + xlab("Year") + ylab("Total sales") + ggtitle("Seasonal differences of sales of new one-family houses, USA")
```
Electricity production

usmelec %>% autoplot()
Electricity production

```
usmelec %>% log() %>% autoplot()
```
Electricity production

\texttt{usmelec \texttt{\%\%} \texttt{log()} \texttt{\%\%} \texttt{diff(\texttt{lag=12}) \texttt{\%\%} autoplot()}}
Electricity production

\texttt{usmelec} \texttt{\%\%} \texttt{log()} \texttt{\%\%} \texttt{diff(lag=12)} \texttt{\%\%} \texttt{diff(lag=1)} \texttt{\%\%} \texttt{autoplot()}
1. Stationarity and differencing
2. Backshift notation
3. Autoregressive models
4. Moving Average models
5. Non-seasonal ARIMA models
6. Seasonal ARIMA models
7. Lab Session 3
Backshift notation

Backward shift operator

Shift back one period

\[ B y_t = y_{t-1} \]
Backshift notation

Backward shift operator

Shift back one period

\[ By_t = y_{t-1} \]

Shift back two periods:

\[ B(By_t) = B^2 y_t = y_{t-2} \]
Backward shift operator

Shift back one period

\[ B y_t = y_{t-1} \]

Shift back two periods:

\[ B(By_t) = B^2 y_t = y_{t-2} \]

Shift back 12 periods

\[ B^{12} y_t = y_{t-12} \]
First differences

\[ y'_t = y_t - y_{t-1} = y_t - By_t = (1 - B)y_t . \]
Backshift notation

- First differences
  \[ y'_t = y_t - y_{t-1} = y_t - By_t = (1 - B)y_t . \]

- Second-order differences (i.e., first differences of first differences):
  \[ y''_t = (1 - B)^2 y_t . \]
First differences

\[ y'_t = y_t - y_{t-1} = y_t - By_t = (1 - B)y_t. \]

Second-order differences (i.e., first differences of first differences):

\[ y''_t = (1 - B)^2y_t. \]

dth-order differences:

\[ (1 - B)^dy_t. \]
Backshift notation

- First differences
  \[ y'_t = y_t - y_{t-1} = y_t - By_t = (1 - B)y_t. \]

- Second-order differences (i.e., first differences of first differences):
  \[ y''_t = (1 - B)^2 y_t. \]

- \(d\)th-order differences:
  \[ (1 - B)^d y_t. \]

- Seasonal difference followed by first difference:
  \[ (1 - B)(1 - B^m)y_t. \]
1. Stationarity and differencing
2. Backshift notation
3. Autoregressive models
4. Moving Average models
5. Non-seasonal ARIMA models
6. Seasonal ARIMA models
7. Lab Session 3
Autoregressive models

Autoregressive (AR) models:

\[ y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t \]

\[(1 - \phi_1 B - \cdots - \phi_p B^p)y_t = c + \varepsilon_t \]

where \( \varepsilon_t \) is white noise. This is a multiple regression with lagged values of \( y_t \) as predictors.
Stationarity conditions

We normally restrict autoregressive models to stationary data, and then some constraints on the values of the parameters are required.

**General condition for stationarity**

Complex roots of $1 - \phi_1 z - \phi_2 z^2 - \cdots - \phi_p z^p$ lie outside the unit circle on the complex plane.
Stationarity conditions

We normally restrict autoregressive models to stationary data, and then some constraints on the values of the parameters are required.

**General condition for stationarity**

Complex roots of $1 - \phi_1 z - \phi_2 z^2 - \cdots - \phi_p z^p$ lie outside the unit circle on the complex plane.

- For $p = 1$: $-1 < \phi_1 < 1$.
- For $p = 2$:
  
  $-1 < \phi_2 < 1$ \quad \phi_2 + \phi_1 < 1 \quad \phi_2 - \phi_1 < 1$.
- More complicated conditions hold for $p \geq 3$. 
Outline

1. Stationarity and differencing
2. Backshift notation
3. Autoregressive models
4. Moving Average models
5. Non-seasonal ARIMA models
6. Seasonal ARIMA models
7. Lab Session 3
Moving Average (MA) models:

\[ y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \]

\[ y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q) \varepsilon_t \]

where \( \varepsilon_t \) is white noise. This is a multiple regression with past errors as predictors.
Invertibility

Invertible models have property that distant past has negligible effect on forecasts. Requires constraints on MA parameters.

**General condition for invertibility**

Complex roots of $1 + \theta_1 z + \theta_2 z^2 + \cdots + \theta_q z^q$ lie outside the unit circle on the complex plane.
Invertibility

- Invertible models have property that distant past has negligible effect on forecasts. Requires constraints on MA parameters.

**General condition for invertibility**

Complex roots of $1 + \theta_1 z + \theta_2 z^2 + \cdots + \theta_q z^q$ lie outside the unit circle on the complex plane.

- For $q = 1$: $-1 < \theta_1 < 1$.
- For $q = 2$:
  - $-1 < \theta_2 < 1$
  - $\theta_2 + \theta_1 > -1$
  - $\theta_1 - \theta_2 < 1$.
- More complicated conditions hold for $q \geq 3$. 
1. Stationarity and differencing
2. Backshift notation
3. Autoregressive models
4. Moving Average models
5. Non-seasonal ARIMA models
6. Seasonal ARIMA models
7. Lab Session 3
ARIMA models

**Autoregressive Moving Average models:**

\[ y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} \]
\[ + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \]

\[ \phi_p(B)y_t = \theta_q(B)\varepsilon_t \]

Predictors include both lagged values of \( y_t \) and lagged errors.

\( \phi_p(B) \) is a \( p \)th order polynomial in \( B \)

\( \theta_q(B) \) is a \( q \)th order polynomial in \( B \)

ARIMA models combine ARMA model with differencing.

\((1 - B)^d y_t\) follows an ARMA model.
**ARIMA models**

**Autoregressive Moving Average models:**

\[ y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \]
\[ \phi_p(B)y_t = \theta_q(B)\varepsilon_t \]

- Predictors include both lagged values of \( y_t \) and lagged errors.
- \( \phi_p(B) \) is a \( p \)th order polynomial in \( B \)
- \( \theta_q(B) \) is a \( q \)th order polynomial in \( B \)
ARIMA models

Autoregressive Moving Average models:

\[ y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} \]
\[ + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \]

\[ \phi_p(B)y_t = \theta_q(B)\varepsilon_t \]

- Predictors include both lagged values of \( y_t \) and lagged errors.
- \( \phi_p(B) \) is a \( p \)th order polynomial in \( B \)
- \( \theta_q(B) \) is a \( q \)th order polynomial in \( B \)

Autoregressive Integrated Moving Average models

- Combine ARMA model with **differencing**.
- \( (1 - B)^d y_t \) follows an ARMA model.
ARIMA models

Autoregressive Integrated Moving Average models

ARIMA($p,d,q$) model

AR: $p = \text{order of the autoregressive part}$

I: $d = \text{degree of first differencing involved}$

MA: $q = \text{order of the moving average part}$.

- White noise model: ARIMA(0,0,0)
- Random walk: ARIMA(0,1,0) with no constant
- Random walk with drift: ARIMA(0,1,0) with const.
- AR($p$): ARIMA($p$,0,0)
- MA($q$): ARIMA(0,0,$q$)
Backshift notation for ARIMA

- **ARIMA**(p, 0, q) model:

\[ y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \]

\[ y_t = c + \phi_1 B y_t + \cdots + \phi_p B^p y_t + \varepsilon_t + \theta_1 B \varepsilon_t + \cdots + \theta_q B^q \varepsilon_t \]

or \( (1 - \phi_1 B - \cdots - \phi_p B^p) y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q) \varepsilon_t \)
Backshift notation for ARIMA

- **ARIMA**\((p, 0, q)\) model:

\[
y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t
\]

\[
y_t = c + \phi_1 B y_t + \cdots + \phi_p B^p y_t + \varepsilon_t + \theta_1 B \varepsilon_t + \cdots + \theta_q B^q \varepsilon_t
\]

or \((1 - \phi_1 B - \cdots - \phi_p B^p)y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q)\varepsilon_t\)

- **ARIMA**\((1,1,1)\) model:

\[
(1 - \phi_1 B) (1 - B) y_t = c + (1 + \theta_1 B) \varepsilon_t
\]

↑

 AR(1) First difference

↑

MA(1)
Backshift notation for ARIMA

- ARIMA($p, 0, q$) model:
  \[ y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \]
  \[ y_t = c + \phi_1 By_t + \cdots + \phi_p B^p y_t + \varepsilon_t + \theta_1 B \varepsilon_t + \cdots + \theta_q B^q \varepsilon_t \]
  or \[ (1 - \phi_1 B - \cdots - \phi_p B^p) y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q) \varepsilon_t \]

- ARIMA(1,1,1) model:
  \[ (1 - \phi_1 B) (1 - B) y_t = c + (1 + \theta_1 B) \varepsilon_t \]

  \[\uparrow \quad \uparrow \quad \uparrow\]

  AR(1)  First  MA(1)

  difference

Written out:

\[ y_t = c + y_{t-1} + \phi_1 y_{t-1} - \phi_1 y_{t-2} + \theta_1 \varepsilon_{t-1} + \varepsilon_t \]
<table>
<thead>
<tr>
<th><strong>R model</strong></th>
</tr>
</thead>
</table>

### Intercept form

\[
(1 - \phi_1 B - \cdots - \phi_p B^p)(1 - B)^d y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q) \varepsilon_t
\]

### Mean form

\[
(1 - \phi_1 B - \cdots - \phi_p B^p)(1 - B)^d (y_t - \mu t^d/d!) =
\]

\[
(1 + \theta_1 B + \cdots + \theta_q B^q) \varepsilon_t
\]

- \( \mu \) is the mean of \((1 - B)^d y_t\).
- \( c = \mu(1 - \phi_1 - \cdots - \phi_p) \).
- R uses mean form.
- Including \( c \) equivalent to \( y_t \) having \( d \)th order polynomial trend.
US personal consumption

```r
autoplot(uschange[, "Consumption"])
  + xlab("Year") + ylab("Quarterly percentage change")
  + ggtitle("US consumption")
```
US personal consumption

```r
(fit <- auto.arima(uschange[, "Consumption"]))
```

```r
## Series: uschange[, "Consumption"]
## ARIMA(2,0,2) with non-zero mean
##
## Coefficients:
##  ar1  ar2  ma1   ma2  mean
## 1.391 -0.581 -1.180 0.558 0.746
## s.e. 0.255 0.208 0.238 0.140 0.084
##
## sigma^2 estimated as 0.351: log likelihood=-165.1
## AIC=342.3 AICc=342.8 BIC=361.7
```
US personal consumption

```r
(fit <- auto.arima(uschange[, "Consumption"]))
```

## Series: uschange[, "Consumption"]
## ARIMA(2,0,2) with non-zero mean
##
## Coefficients:
## ar1   ar2  ma1  ma2  mean
## 1.391 -0.581 -1.180 0.558 0.746
## s.e.  0.255 0.208 0.238 0.140 0.084
##
## sigma^2 estimated as 0.351: log likelihood=-165.1
## AIC=342.3  AICc=342.8  BIC=361.7

**ARIMA(2,0,2) model:**

\[ y_t = c + 1.391y_{t-1} - 0.581y_{t-2} - 1.180\varepsilon_{t-1} + 0.558\varepsilon_{t-2} + \varepsilon_t, \]

where \( c = 0.746 \times (1 - 1.391 + 0.581) = 0.142 \) and \( \varepsilon_t \sim N(0, 0.351). \)
US personal consumption

fit %>% forecast(h=10) %>% autoplot(include=80)

Forecasts from ARIMA(2,0,2) with non-zero mean
Akaike’s Information Criterion (AIC):

\[
\text{AIC} = -2 \log(L) + 2(p + q + k + 1),
\]

where \( L \) is the likelihood of the data,
\( k = 1 \) if \( c \neq 0 \) and \( k = 0 \) if \( c = 0 \).
Akaike’s Information Criterion (AIC):

\[ \text{AIC} = -2 \log(L) + 2(p + q + k + 1), \]

where \( L \) is the likelihood of the data,
\( k = 1 \) if \( c \neq 0 \) and \( k = 0 \) if \( c = 0 \).

Corrected AIC:

\[ \text{AICc} = \text{AIC} + \frac{2(p + q + k + 1)(p + q + k + 2)}{T - p - q - k - 2}. \]
Akaike’s Information Criterion (AIC):

\[ AIC = -2 \log(L) + 2(p + q + k + 1), \]

where \( L \) is the likelihood of the data,
\( k = 1 \) if \( c \neq 0 \) and \( k = 0 \) if \( c = 0 \).

Corrected AIC:

\[ AICc = AIC + \frac{2(p + q + k + 1)(p + q + k + 2)}{T - p - q - k - 2}. \]

Good models are obtained by minimizing the AICc.
How does auto.arima() work?

A non-seasonal ARIMA process

\[ \phi(B)(1 - B)^d y_t = c + \theta(B) \varepsilon_t \]

Need to select appropriate orders: \( p, q, d \)

Hyndman and Khandakar (JSS, 2008) algorithm:

- Select no. differences \( d \) and \( D \) via KPSS test and seasonal strength measure.
- Select \( p, q \) by minimising AICc.
- Use stepwise search to traverse model space.
How does auto.arima() work?

**Step 1:** Select values of $d$ and $D$.

**Step 2:** Select current model (with smallest AICc) from:

- ARIMA(2, $d$, 2)
- ARIMA(0, $d$, 0)
- ARIMA(1, $d$, 0)
- ARIMA(0, $d$, 1)
How does aut.arima() work?

**Step 1:** Select values of $d$ and $D$.

**Step 2:** Select current model (with smallest AICc) from:
- ARIMA(2, $d$, 2)
- ARIMA(0, $d$, 0)
- ARIMA(1, $d$, 0)
- ARIMA(0, $d$, 1)

**Step 3:** Consider variations of current model:
- vary one of $p$, $q$, from current model by $\pm 1$;
- $p$, $q$ both vary from current model by $\pm 1$;
- Include/exclude $c$ from current model.

Model with lowest AICc becomes current model.

Repeat Step 3 until no lower AICc can be found.
Choosing an ARIMA model

`autoplot(internet)`
Choosing an ARIMA model

```r
(fit <- auto.arima(internet))
```

```
## Series: internet
## ARIMA(1,1,1)
##
## Coefficients:
##          ar1    ma1
## 0.650   0.526
## s.e.   0.084  0.090

## sigma^2 estimated as 10:  log likelihood=-254.2
## AIC=514.3   AICc=514.5   BIC=522.1
```
Choosing an ARIMA model

```r
(fit <- auto.arima(internet, stepwise=FALSE, approximation=FALSE))
```

```
## Series: internet
## ARIMA(3,1,0)
##
## Coefficients:
##          ar1     ar2     ar3
##      1.151   -0.661   0.341
## s.e.  0.095   0.135   0.094
##
## sigma^2 estimated as 9.66:  log likelihood=-252
## AIC=512   AICc=512.4   BIC=522.4
```
Choosing an ARIMA model

checkresiduals(fit, plot=TRUE)

Residuals from ARIMA(3,1,0)

Ljung-Box test
data: Residuals from ARIMA(3,1,0)
Q* = 4.5, df = 7, p-value = 0.7
Model df: 3. Total lags used: 10
Choosing an ARIMA model

```r
fit %>% forecast() %>% autoplot()
```

Forecasts from ARIMA(3,1,0)
### Seasonal ARIMA models

<table>
<thead>
<tr>
<th>ARIMA</th>
<th>$(p, d, q)$</th>
<th>$(P, D, Q)_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>↑</td>
<td>↑</td>
</tr>
<tr>
<td>Non-seasonal part of the model</td>
<td>Seasonal part of the model</td>
<td></td>
</tr>
</tbody>
</table>

where $m =$ number of observations per year.
Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)_4 model (without constant)
Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)₄ model (without constant)

\[(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^4)\varepsilon_t.\]
Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)_4 model (without constant)

\[(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^4)\varepsilon_t.\]
Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)_4 model (without constant)

\[(1-\phi_1B)(1-\Phi_1B^4)(1-B)(1-B^4)y_t = (1+\theta_1B)(1+\Theta_1B^4)\varepsilon_t.\]

All the factors can be multiplied out and the general model written as follows:

\[y_t = (1 + \phi_1)y_{t-1} - \phi_1y_{t-2} + (1 + \Phi_1)y_{t-4}\]
\[- (1 + \phi_1 + \Phi_1 + \phi_1\Phi_1)y_{t-5} + (\phi_1 + \phi_1\Phi_1)y_{t-6}\]
\[- \Phi_1y_{t-8} + (\Phi_1 + \phi_1\Phi_1)y_{t-9} - \phi_1\Phi_1y_{t-10}\]
\[+ \varepsilon_t + \theta_1\varepsilon_{t-1} + \Theta_1\varepsilon_{t-4} + \theta_1\Theta_1\varepsilon_{t-5}.\]
European quarterly retail trade

```r
autoplot(euretail) +
  xlab("Year") + ylab("Retail index")
```
European quarterly retail trade

(fit <- \texttt{auto.arima}(euretail))

## Series: euretail
## ARIMA(1,1,2)(0,1,1)[4]
##
## Coefficients:
## \begin{tabular}{ccccc}
## ar1 & ma1 & ma2 & sma1 \\
## 0.736 & -0.466 & 0.216 & -0.843 \\
## s.e. & 0.224 & 0.199 & 0.210 & 0.188 \\
## \end{tabular}
##
## \texttt{sigma^2} estimated as 0.159: \texttt{log likelihood}=-29.62
## \texttt{AIC}=69.24 \quad AICc=70.38 \quad BIC=79.63
(fit <- auto.arima(euretail, stepwise=TRUE, approximation=FALSE))

## Series: euretail
## ARIMA(1,1,2)(0,1,1)[4]
##
## Coefficients:
##      ar1   ma1   ma2   sma1
##    0.736 -0.466  0.216  -0.843
## s.e. 0.224  0.199  0.210  0.188
##
## sigma^2 estimated as 0.159:  log likelihood=-29.62
## AIC=69.24   AICc=70.38   BIC=79.63
European quarterly retail trade

`checkresiduals(fit, test=FALSE)`

Residuals from ARIMA(1,1,2)(0,1,1)[4]

ACF

Count

Lag

Residuals

-1.0
-0.5
0.0
0.5
1.0
2000
2005
2010

-1.0
-0.5
0.0
0.5
1.0
4
8
12
16

0
5
10

-1.0
-0.5
0.0
0.5
1.0

residuals
count

-1.0
-0.5
0.0
0.5
1.0
2000
2005
2010

-1.0
-0.5
0.0
0.5
1.0
4
8
12
16

0
5
10

-1.0
-0.5
0.0
0.5
1.0

residuals
count

-1.0
-0.5
0.0
0.5
1.0
2000
2005
2010

-1.0
-0.5
0.0
0.5
1.0
4
8
12
16

0
5
10

-1.0
-0.5
0.0
0.5
1.0

residuals
count

-1.0
-0.5
0.0
0.5
1.0
2000
2005
2010

-1.0
-0.5
0.0
0.5
1.0
4
8
12
16

0
5
10

-1.0
-0.5
0.0
0.5
1.0

residuals
count

45
European quarterly retail trade

```r
forecast(fit, h=36) %>% autoplot()
```

**Forecasts from ARIMA(1,1,2)(0,1,1)[4]**
Cortecosteroid drug sales
(fit <- auto.arima(h02, lambda=0, max.order=9, stepwise=FALSE, approximation=FALSE))

## Series: h02
## ARIMA(4,1,1)(2,1,2)[12]
## Box Cox transformation: lambda= 0

## Coefficients:
##      ar1    ar2    ar3    ar4    ma1    sar1
## -0.042 0.210 0.202 -0.227 -0.742  0.621
##     s.e.  0.217  0.181  0.114  0.081  0.207  0.242
##      sar2    sma1    sma2
## -0.383 -1.202   0.496
##     s.e.  0.118  0.249  0.214
Cortecosteroid drug sales

checkresiduals(fit)

Residuals from ARIMA(4,1,1)(2,1,2)[12]

Lag

ACF

count

residuals

Ljung-Box test
data: Residuals from ARIMA(4,1,1)(2,1,2)[12]
Q* = 16, df = 15, p-value = 0.4
Model df: 9. Total lags used: 24
Understanding ARIMA models

### Long-term forecasts

<table>
<thead>
<tr>
<th>Type</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>zero</td>
<td>$c = 0, d + D = 0$</td>
</tr>
<tr>
<td>non-zero constant</td>
<td>$c = 0, d + D = 1$</td>
</tr>
<tr>
<td>linear</td>
<td>$c = 0, d + D = 2$</td>
</tr>
<tr>
<td>quadratic</td>
<td>$c = 0, d + D = 3$</td>
</tr>
</tbody>
</table>

### Forecast variance and $d + D$

- The higher the value of $d + D$, the more rapidly the prediction intervals increase in size.
- For $d + D = 0$, the long-term forecast standard deviation will go to the standard deviation of the historical data.
Prediction intervals increase in size with forecast horizon.

Calculations assume residuals are uncorrelated and normally distributed.

Prediction intervals tend to be too narrow.

- the uncertainty in the parameter estimates has not been accounted for.
- the ARIMA model assumes historical patterns will not change during the forecast period.
- the ARIMA model assumes uncorrelated future errors
Outline

1. Stationarity and differencing
2. Backshift notation
3. Autoregressive models
4. Moving Average models
5. Non-seasonal ARIMA models
6. Seasonal ARIMA models
7. Lab Session 3
Lab Session 3