ARMA definition

$ARMA(p, q)$ processes are random processes depending on $p + q$ real-valued parameters.

We consider an $ARMA(p, q)$ process given by

$$X_t - \phi_1 X_{t-1} - ... - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + ... + \theta_q Z_{t-q}$$

where$\{Z_t\} \sim WN(0, \sigma^2)$. $\{X_t\}$ is then defined by the two polynomials $\phi(z) = 1 - \phi_1 z - ... - \phi_p z^p$ and $\theta(z) = 1 + \theta_1 z + ... + \theta_q z^q$.

We assume that $\phi(z)$ and $\theta(z)$ do not have any common root.

Causal ARMA

The process $\{X_t\}$ is said to be causal if $\phi(z) \neq 0$ for all $\mid z \mid \leq 1$. If so, the process can be written as

$$X_t = \sum_{j=0}^{\infty} \psi_j Z_{t-j}$$

where $\sum_{j=0} \mid \psi_j \mid < \infty$ and where

$$\psi_j = \sum_{k=1}^{p} \phi_k \psi_{j-k} + \theta_j$$

and where $\psi_0 = 0, \theta_0 = 1, \theta_j = 0$, for $j > q$.

Invertible ARMA

The process $\{X_t\}$ is said to be invertible if $\theta(z) \neq 0$ for all $\mid z \mid \leq 1$. If so, the process can be written as

$$Z_t = \sum_{j=0}^{\infty} \pi_j X_{t-j}$$

where $\sum_{j=0} \mid \pi_j \mid < \infty$ and where

$$\pi_j = -\sum_{k=1}^{q} \theta_k \pi_{j-k} - \phi_j$$

and where $\phi_0 = -1, \phi_j = 0$, for $j > p$ and $\pi_j = 0$, for $j < 0$.

ARMA(1,1) example

The time series $\{X_t\}$ is said to be an $ARMA(1,1)$ if it is stationary and verifies the equation

$$X_t - \phi X_{t-1} = Z_t + \theta Z_{t-1}$$

where $\{Z_t\} \sim WN(0, \sigma^2)$ and $\phi + \theta \neq 0$.

ARMA(1,1) example

The time series $\{X_t\}$ is said to be an $ARMA(1,1)$ if it is stationary and verifies the equation

$$X_t - \phi X_{t-1} = Z_t + \theta Z_{t-1}$$

where $\{Z_t\} \sim WN(0, \sigma^2)$ and $\phi + \theta \neq 0$.

Using the backward shift operator, the equation of $ARMA(1,1)$ can be rewritten as

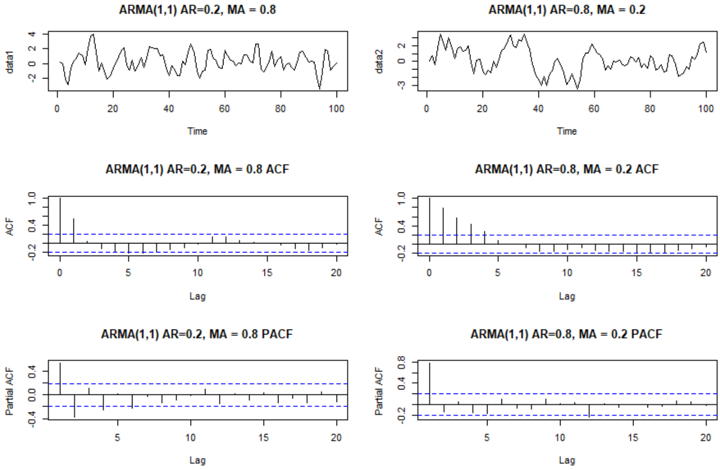
$$\phi(B) X_t = \theta(B) Z_t$$

where $\phi(B) = 1 - \phi(B)$ and $\theta(B) = 1 + \theta(B)$ are two linear filters.

ARMA(1,1) example - R code

```
# ARMA(1,1) simulations
set.seed(1986)
n <- 100
sd <- 1
data1 <- arima.sim(model = list(ar = c(0.2), ma = c(0.8),
         order = c(1, 0, 1)), n = n, sd = sd)
data2 <- arima.sim(model = list(ma = c(0.2), ar = c(0.8),
         order = c(1, 0, 1)), n = n, sd = sd)

par(mfrow = c(3, 2))
plot(data1, main = "ARMA(1,1) AR=0.2, MA = 0.8")
plot(data2, main = "ARMA(1,1) AR=0.8, MA = 0.2")
acf(data1, main = "ARMA(1,1) AR=0.2, MA = 0.8 ACF")
acf(data2, main = "ARMA(1,1) AR=0.8, MA = 0.2 ACF")
pacf(data1, main = "ARMA(1,1) AR=0.2, MA = 0.8 PACF")
pacf(data2, main = "ARMA(1,1) AR=0.8, MA = 0.2 PACF")
```

ARMA(1,1) example - plots



ARMA(1,1) causality and invertibility example

(after P. Brockwell and R. Davis, p.86)

We consider the following $ARMA(1,1)$ process

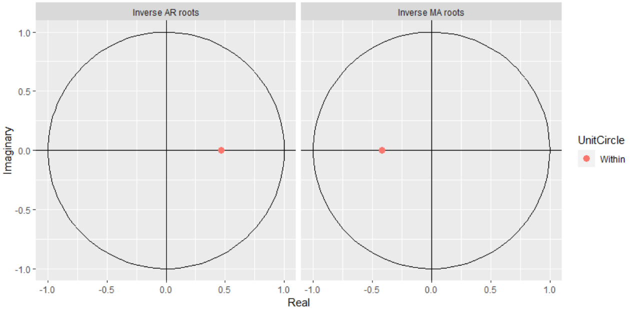
$$X_t - 0.5X_{t-1} = Z_t + 0.4Z_{t-1}, \quad \{Z_t\} \sim WN(0, \sigma^2)$$

The AR polynomial $\phi(z) = 1 - 0.5z$ has a zero at $z = 2$. It is located outside of the unit circle so we can conclude that the process is causal. The inverse is $1/z = 0.5$, which is located inside of the unit circle (same conclusion of causality).

The MA polynomial $\theta(z) = 1 + 0.4z$ has a zero at $z = -2.5$. It is located outside of the unit circle so we can conclude that the process is invertible. The inverse is $1/z = -0.4$, which is located inside of the unit circle (same conclusion of invertibility).

ARMA(1,1) visualizing causality and invertibility

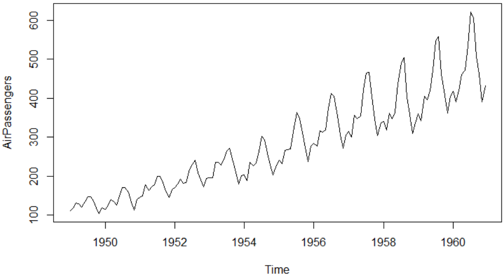
```
# inverse of unit root
library(forecast)
set.seed(1986)
n <- 1000
sd <- 1
data3 <- arima.sim(model = list(ar = c(0.5), ma = c(0.4), order = c(1, 0, 1)),
                   n = n, sd = sd)
fit <- Arima(data3, order=c(1,0,1))
autoplot(fit)
```



Practical example

We will fit an $ARIMA(p, d, q)$ model to the 'AirPassengers' data.

```
# practical example: Airpassengers data
plot(AirPassengers)
class(AirPassengers)
# [1] "ts"
AirPassengers
#       Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
# 1949 112 118 132 129 121 135 148 148 136 119 104 118
# 1950 115 126 141 135 125 149 170 170 158 133 114 140
# 1951 145 150 178 163 172 178 199 199 184 162 146 166
# 1952 171 180 193 181 183 218 230 242 209 191 172 194
```



Testing for stationarity

Clearly, the data appear non stationary as the mean is not constant over time. We will apply differencing of order one and test for stationarity using the Augmented Dickey-Fuller test.

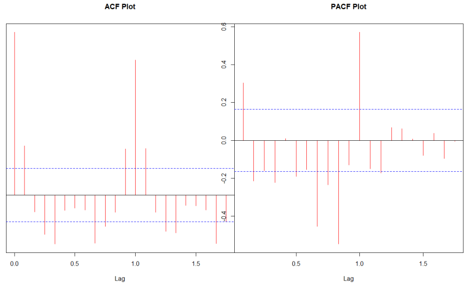
```
# Augmented Dickey-Fuller test for stationarity
library(tseries)
adf.test(diff(AirPassengers), alternative="stationary", k=0)

# Augmented Dickey-Fuller Test
#
# data:  diff(AirPassengers)
# Dickey-Fuller = -8.5472, Lag order = 0, p-value = 0.01
# alternative hypothesis: stationary
#
# Message d'avis :
# Dans adf.test(diff(AirPassengers), alternative = "stationary", k = 0) :
#   p-value smaller than printed p-value
```

Now we can use the Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF) to determine the best ARIMA model that could fit those data.

ACF and PACF of the stationary series

The ACF helps us determine the AR part of the ARIMA process, while the PACF helps us determine the MA part of the ARIMA process. Definitely the lags at which the ACF and PACF cut off remain unclear. We can use an automatic method to help us determine which model to fit.

```
# ACF and PACF of stationary series
par(mfrow = c(1,2))
acf(diff(AirPassengers), plot = TRUE, type = 'correlation',main='ACF␣Plot', col='red')
pacf(diff(AirPassengers), plot = TRUE, main="PACF␣Plot", col='red')
```



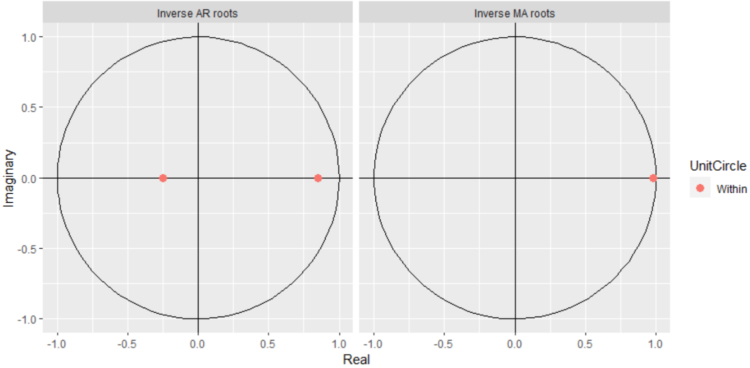
ARIMA model selection

We use the function *auto.arima()* to select the best model. The retained model will be an $ARIMA(2,1,1)$.

```
# ARIMA model estimation
library(forecast)
set.seed(1986)
summary(fit <- auto.arima(AirPassengers))
# Series: AirPassengers
# ARIMA(2,1,1)(0,1,0)[12]
#
# Coefficients:
#          ar1     ar2      ma1
#       0.5960  0.2143  -0.9819
# s.e.  0.0888  0.0880   0.0292
#
# sigma^2 estimated as 132.3:  log likelihood=-504.92
# AIC=1017.85   AICc=1018.17   BIC=1029.35
#
# Training set error measures:
#    ME      RMSE      MAE      MPE      MAPE      MASE        ACF1
# Training set 1.3423 10.84619 7.86754 0.420698 2.800458 0.245628 -0.00124847
```

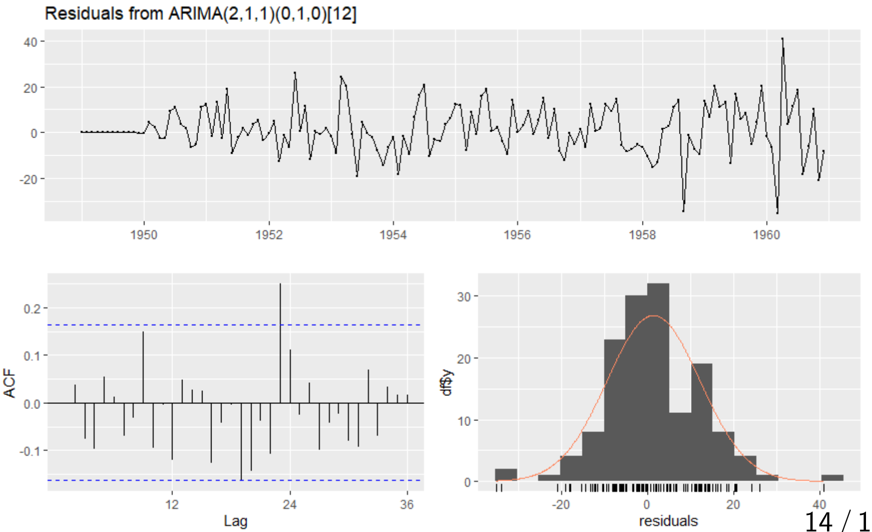
Checking characteristic roots for the ARIMA(2,1,1) model

```
autoplot(fit)
```



Checking the residuals of ARIMA(2,1,1)

```
checkresiduals(fit)
```

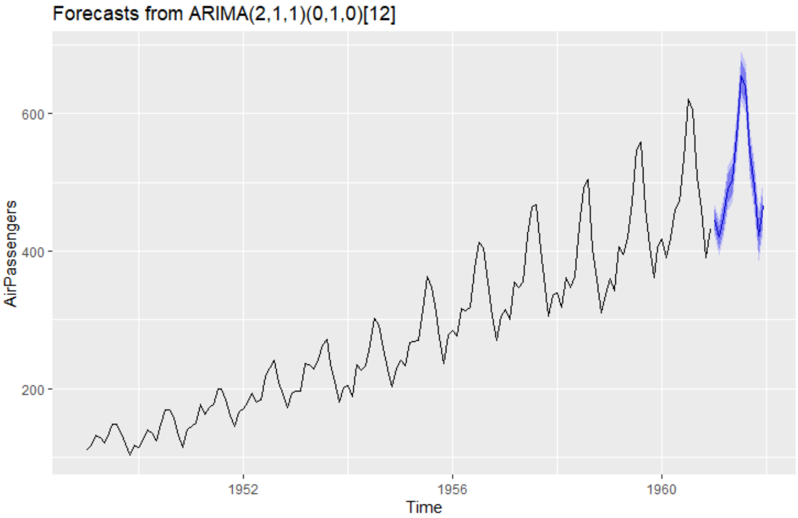


ARIMA(2,1,1) model forecasts

We can make forecasts for the next 12 months using the $ARIMA(2,1,1)$ model.

```
# ARIMA model forecasting
library(tidyverse)
fit %>% forecast(level = c(95), h=12) %>% autoplot()
forecast <- forecast(fit, level = c(95), h=12)
forecast
#          Point Forecast    Lo 95    Hi 95
# Jan 1961       445.6349 423.0851 468.1847
# Feb 1961       420.3950 393.9304 446.8596
# Mar 1961       449.1983 419.4892 478.9074
# Apr 1961       491.8399 460.0092 523.6707
# May 1961       503.3945 469.9953 536.7937
# Jun 1961       566.8624 532.3007 601.4242
# Jul 1961       654.2602 618.8122 689.7081
# Aug 1961       638.5975 602.4630 674.7320
# Sep 1961       540.8837 504.2081 577.5594
# Oct 1961       494.1266 457.0177 531.2356
# Nov 1961       423.3327 385.8715 460.7939
# Dec 1961       465.5076 427.7556 503.2596
```

Visualizing the forecasts of ARIMA(2,1,1)



Further reading and code

Brockwell, P.J., Davis, R.A, (2002). Introduction to Time Series and Forecasting, Second Edition (2nd ed.). Springer.
ISBN 0-387-95351-5

The R Project for Statistical Computing:
https://www.r-project.org/