Methods of Empirical Finance

Seminar (UE)

Christoph Huber
University of Innsbruck

Master in Banking and Finance
Winter term 2019/20 (this version: 2019-12-10)

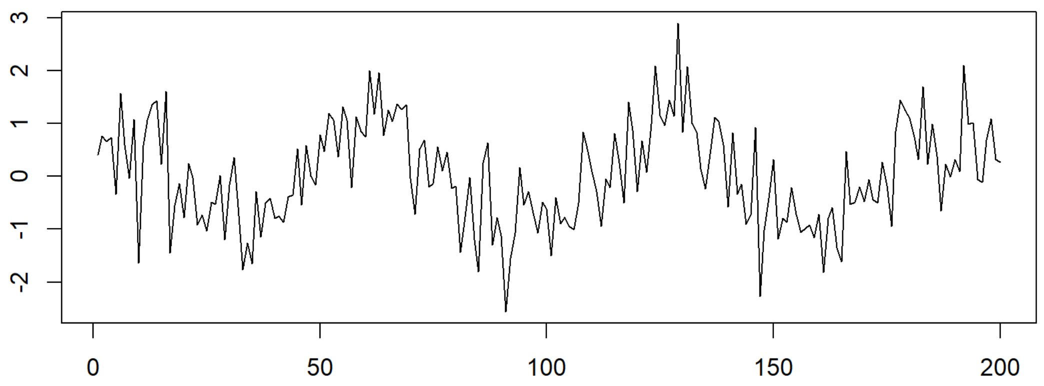
Time Series Analysis

Methods of Empirical Finance

Time Series Analysis

Time Series Processes

> A time series $\{y_t\}$ is an ordered set of (random) variables, where the (time) ordering is important.
>
> Data obtained from observations collected *sequentially* over time.



Notation: Observations y_t, at time $t = 1, \ldots, T$.

Goals: *Explanation* (understand or model the stochastic mechanism underlying the series) and *Prediction* (Forecast future values)

Time Series Analysis

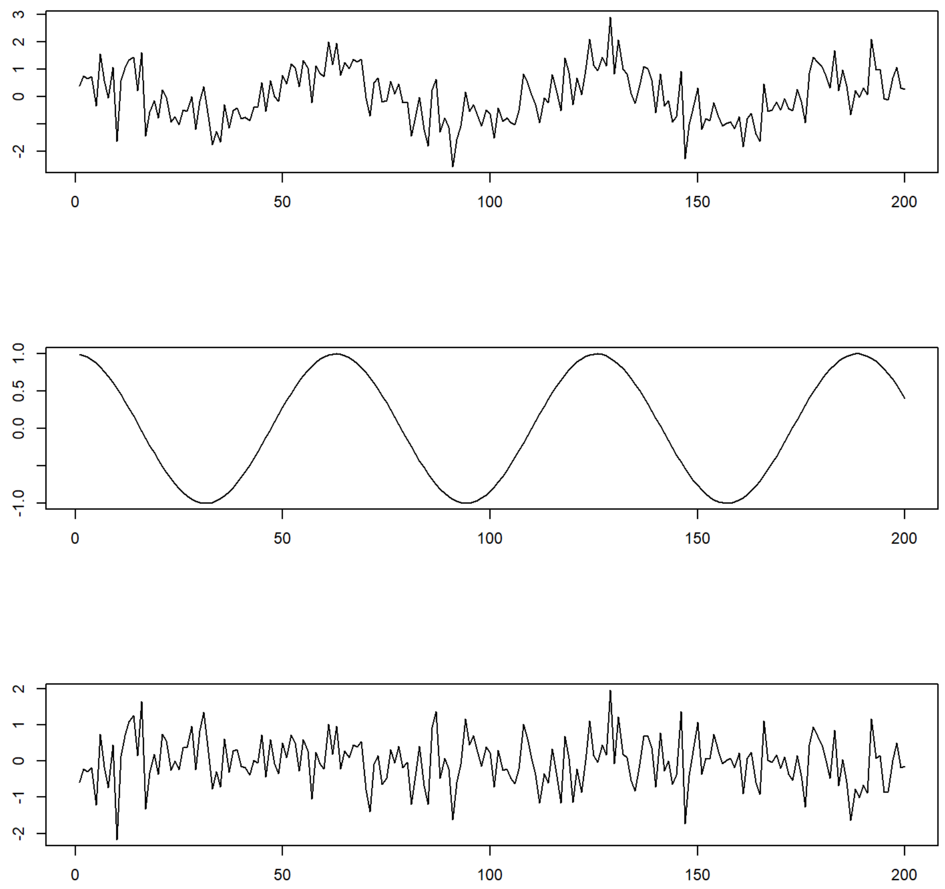
Univariate and Multivariate Time Series Analysis

- time series analysis accounts for the fact that data points taken over time may have an *internal structure* that could be utilized to draw conclusions about the series' movements and to make predictions about its future path
- the aim of time series analysis is to find an appropriate statistical model for the data and to use this model for prediction: in this way, *the variables are allowed to speak for themselves*, without the "confines" of economic theory

- **univariate time series models** are a class of specifications to model and predict variables using only information contained *in their own past values* (and possibly current and past values of an error term)
- **structural models**, on the other hand, are multivariate in nature, and attempt to explain changes in a variable by reference to the movements in the current or past values of other (explanatory) variables (e.g., vector auto-regressive models, VAR)

- $\rightarrow$ time series analyses can be applied to get knowledge about the underlying *data generating process*, to **estimate inherent patterns** like trends, seasonality, or cyclical components, and to test economic hypotheses

Time Series Analysis



Time Series Analysis

Time Series Operators

- the i^{th} **difference operator** is defined by

$$\Delta_i y_t = y_t - y_{t-i}$$

Examples:

- first difference operator: $\Delta y_t = y_t - y_{t-1}$
- second difference operator: $\Delta y_t = y_t - y_{t-2}$
- Seasonal differencing, e.g.: $\Delta y_t = y_t - y_{t-12}$

In R: `diff()`

Time Series Analysis

Time Series Operators

- the **lag operator** is defined as

$$Ly_t = y_{t-1}$$

- more generally: $L^k y_t = y_{t-k}$
- thus, $y_t + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} = (1 + \alpha_1 L + \alpha_2 L^2) y_t$

In R: `lag()`

Time Series Analysis

Mean and Variance of Time Series

- time series may have both **stochastic** and **deterministic** components, e.g., a series with a deterministic trend and a stochastic white noise component:

$$y_t = \alpha + \beta t + \varepsilon_t \qquad \text{where } \varepsilon_t \sim i.i.d(0, \sigma^2)$$

- most time series models of financial markets will have a stochastic component, and so the **unconditional expectation** and variance of the t^{th} observation on the time series can be calculated; for example in the model above

$$\mu_{y_t} = E[y_t] = E[\alpha + \beta t + \varepsilon_t] = \alpha + \beta t$$

$$\lambda_0 = Var[y_t] = E[(y_t - E[y_t])^2] = E[(\alpha + \beta t + \varepsilon_t - \alpha + \beta t)^2] = E[\varepsilon_t^2] = \sigma^2$$

Time Series Analysis

Autocovariance and Autocorrelation of Time Series

- the s^{th}**-order autocovariance** of some time series $\{y_t\}$, i.e., the unconditional covariance of y_t and y_{t-s} (that is, the unconditional covariance *at lag s* is

$$\lambda_s = Cov[y_t, y_{t-s}] = E[(y_t - E[y_t])(y_{t-s} - E[y_{t-s}])]$$

- for instance, for the series above, the s^{th}-order autocovariance is

$$\begin{aligned} \lambda_s = Cov[y_t, y_{t-s}] &= E[(\alpha + \beta t + \varepsilon_t - \alpha + \beta t)(\alpha + \beta t + \varepsilon_{t-s} - \alpha + \beta t)] \\ &= E[\varepsilon_t \varepsilon_{t-s}] = 0 \end{aligned}$$

- the **autocorrelation function** is then defined by

$$\rho_s = Corr[y_t, y_{t-s}] = \frac{\lambda_s}{\lambda_0}$$

Time Series Analysis

Autocovariance and Autocorrelation of Time Series

- the **partial autocorrelation function** between y_t and y_{t-s} is is the correlation between y_t minus the part explained by intervening lags and y_{t-s} (where E^* is the minimum mean-squared error prediction of y_t by $y_{t-1}, \ldots, y_{t-s+1}$)

$$\rho_s^* = Corr[y_t - E^*[y_t | y_{t-1}, \ldots, y_{t-s+1}], y_{t-s}]$$

- thus, the partial autocorrelation gives the partial correlation of a time series eith its own lagged values, controlling for intervening effects of values at shorter lags

Time Series Analysis

Returns in Financial Analysis and Modeling

- Starting point is often **prices**, however, most often it is preferable to work with *returns*, i.e., prices have to be converted to returns

Arithmetic and logarithmic returns

- let p_t denote the price at time t; the simple "arithmetic" return R_t and the *continuously compounded* logarithmic return r_t are then defined by

$$R_t = \frac{p_t - p_{t-1}}{p_{t-1}} \qquad r_t = ln\left(\frac{p_t}{p_{t-1}}\right)$$

- arithmetic returns are *not symmetric*: e.g. an increase by 50% and a decrease by 50% lead to a total change of -25%
- log returns (continuously compounded returns) are *time-additive* but not additive across portfolios

Stationarity

Stationarity

Weak Stationarity

> A time series process $\{y_t\}$ is said to be covariance stationary or weakly stationary of the following requirements are satisfied:
>
> 1. $E[y_t] = \mu$
> 2. $Var[y_t] = \lambda_0^2 < \infty$
> 3. $Cov[y_t, y_{t-s}] = \lambda_{t,t-s} \forall t, t-s$

That is, a time series process is *weakly stationary* if . . .

- the mean is constant (independent of t),
- the variance is a finite, positive constant, independent of t, and
- the auto-covariance is a finite function of the lag s, but not of the absolute location of either observation on the time scale

Stationarity

Strong Stationarity

> A time series process $\{y_t\}$ is said to be strictly stationary or strongly stationary if the joint probability distribution of any set of k observations in the sequence $\{y_t, y_{t+1}, \ldots, y_{t+k}\}$ is the same regardless of the origin t in the time scale.

That is, a time series process is said to be *strongly stationary* if the joint probability distribution does not change when shifted in time, i.e., if for any $t_1, t_2, \ldots, t_T \in \mathbb{Z}$, any $k \in \mathbb{Z}$, and $T = 1, 2, \ldots, T, \ldots$

$$F_{y_{t_1}, y_{t_2}, \ldots, y_{t_T}}(y_1, \ldots, y_T) = F_{y_{t_{1+k}}, y_{t_{2+k}}, \ldots, y_{t_{T+k}}}(y_1, \ldots, y_T)$$

I.e., a series `({y_t})` is strictly stationary if the distribution of its values remains the same is time progresses, implying that the probability that `(y)` falls within a particular interval is the same now as at any time in the past or future.

Stationarity

Financial Time Series

In applications of time series analysis to financial markets, *return data* and *price data* have to be treated differently as we have to distinguish between **stationary** and **non-stationary time series**:

- **returns** are mainly *stationary* (mean-reverting, little autocorrelation)
- **prices** are mainly *non-stationary* (time-trending, autocorrelation)

Note that some concepts – in terms of financial economics – such as, for instance, the volatility of time series or the correlation between two different processes, which apply to stationary processes, do not apply to non-stationary processes.

Stationarity

Example: Download stock price data

```r
library(tidyverse)
library(tidyquant)

apple <- tq_get("AAPL")    # download Apple Inc. (AAPL) prices
```

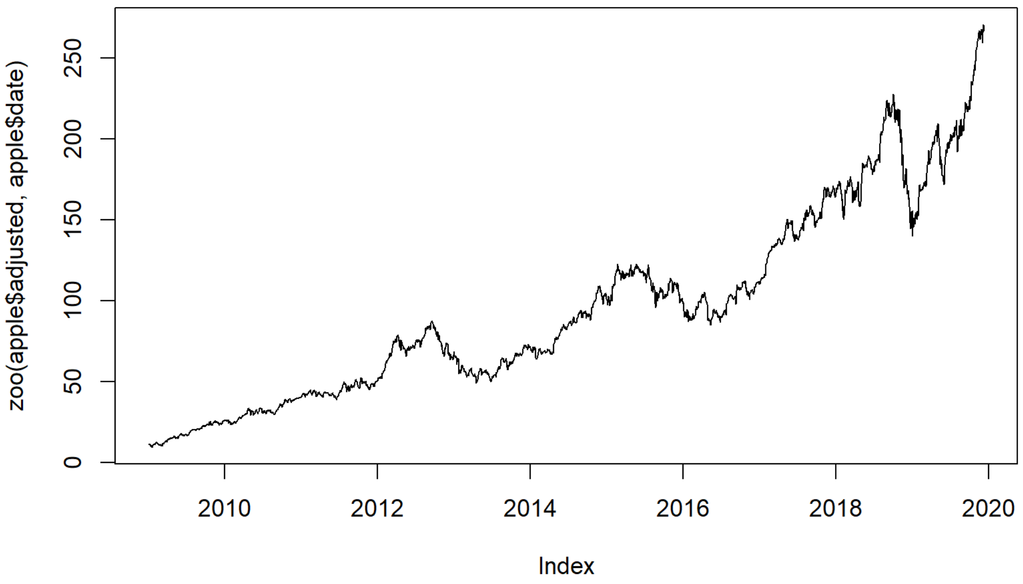
Stata:

`getsymbols` command (install using `ssc install getsymbols`)

Stationarity

Example: Apple Inc. stock price — a non-stationary time series

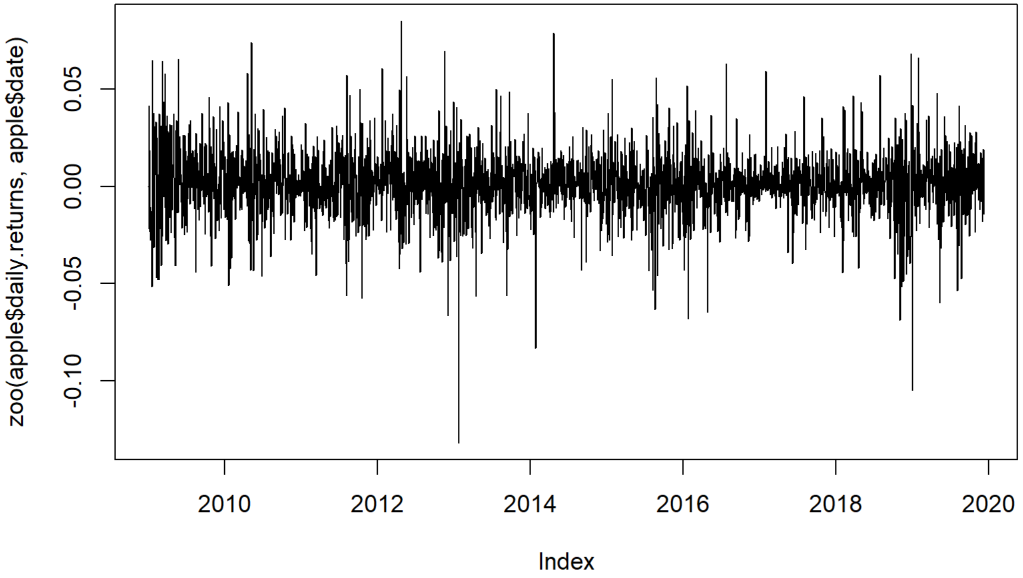
```
plot(zoo(apple$adjusted, apple$date))
```



Stationarity

Example: Apple Inc. stock return — a stationary time series

```
apple <- apple %>% tq_mutate(select = adjusted, mutate_fun = periodReturn,
                             period = "daily", type = "log")
plot(zoo(apple$daily.returns, apple$date))
```



Stationarity

Mean Reversion

- stationary series exhibit a well-know property referred to as **mean-reversion**: due to its finite variance, a stationary process can never drift too far from its mean
- the speed of mean reversion is determined by the *autocovariance*: mean reversion is quick when autocovariances are small and slow when autocovariances are large

- in finance, mean reversion refers to the assumption that security prices tend to move towards the average price over time: assets seem to be attractive to buy when the current price is below the average price and vice versa
- in this sense, mean reversion is opposed by the empirically observed tendency for rising asset prices to rise further and falling prices to keep falling (**momentum**)

Stationarity

Stationarity vs. Non-Stationarity

- any series with a trend in the mean will not be stationary: when prices appear to be trending, this is normally due to a stochastic rather than a deterministic trend
- while prices (or log prices) in most markets are non-stationary, the first difference in prices – or, more usually, the **first difference in log prices** (as these are approx. equal to returns) – are modeled as a stationary process

Stationarity

Integrated Processes

> Let $\{y_t\}$ be a non-stationary process such that $\Delta^d\{y_t\}$ is stationary but $\Delta^{d-1}\{y_t\}$ is non-stationary. Then $\{y_t\}$ is called integrated of order d, denoted $y_t \sim I(d)$.

Example:

- consider the time series process $y_t = \mu + y_{t-1} + u_t$ where $u_t \sim WN(0, \sigma^2)$ (referred to as *random walk with drift*); taking first difference of y_t yields $\Delta y_t = y_t - y_{t-1} = \mu + u_t$, i.e., the white noise error term which is stationary
- i.e., a random walk is non-stationary but becomes stationary when taking the first difference; thus, a random walk is an *integrated process of order 1*, denoted $I(1)$

Stationarity

Tests for Stationarity: unit root tests

A number of different statistical tests are available to test whether a time series is *stationary* or *non-stationary*: *(augmented) Dickey-Fuller (GLS) test, Phillips-Perron test, KPPS test*, etc.

- H_0: the (time series) process has a unit root, i.e., is *non-stationary*
- H_1: depending on test, usually *stationarity* or *trend-stationarity* of the time series

Stationarity

Tests for Stationarity: unit root tests

```
library(tseries)
adf.test(apple$adjusted)
```

```
##
##      Augmented Dickey-Fuller Test
##
## data:  apple$adjusted
## Dickey-Fuller = -1.2799, Lag order = 14, p-value = 0.8832
## alternative hypothesis: stationary
```

```
pp.test(apple$adjusted)
```

```
##
##      Phillips-Perron Unit Root Test
##
## data:  apple$adjusted
## Dickey-Fuller Z(alpha) = -4.2365, Truncation lag parameter = 9,
## p-value = 0.8734
## alternative hypothesis: stationary
```

Stationarity

Tests for Stationarity: unit root tests

```
adf.test(apple$daily.returns)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  apple$daily.returns
## Dickey-Fuller = -13.693, Lag order = 14, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(diff(log(apple$adjusted)))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(log(apple$adjusted))
## Dickey-Fuller = -13.669, Lag order = 14, p-value = 0.01
## alternative hypothesis: stationary
```

Stationarity

Detrending Financial Time Series

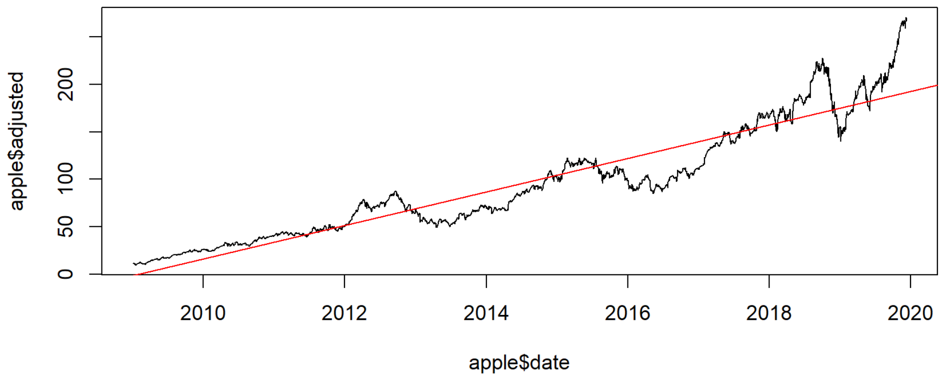
- since a trending time series is typically non-stationary, we would like to remove trends in order to make it a stationary time series (referred to as detrending)
- importantly, trends might either be deterministic or stochastic in nature

- to remove a deterministic trend… … compute deviations from a fitted line
- to remove a stochastic trend… … integrate the process, i.e., take differences

Stationarity

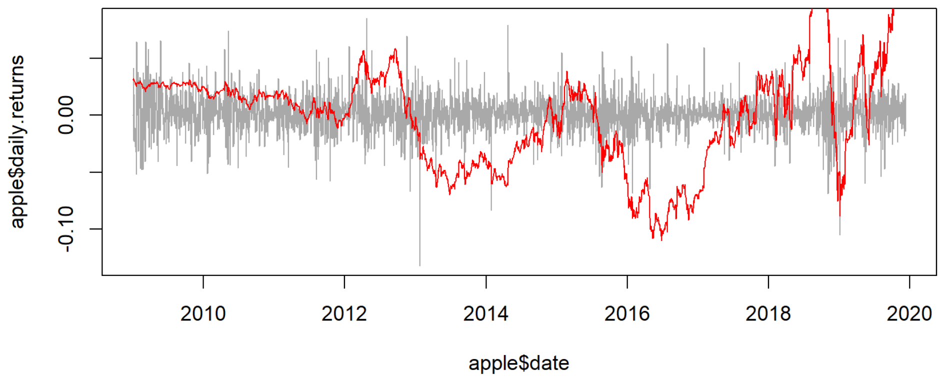
Detrending Financial Time Series

```
plot(apple$date, apple$adjusted, type="l")
abline(lm(adjusted ~ date, data = apple), col=2)
```




```
#
#
```

```
apple$trend <- predict(lm(adjusted ~ date, data = apple)
apple$deviation <- apple$adjusted - apple$trend
plot(apple$date, apple$daily.returns, type = "l", col =
lines(apple$date, apple$deviation/400, col = 2)
```



Autoregressive and Moving Average Models

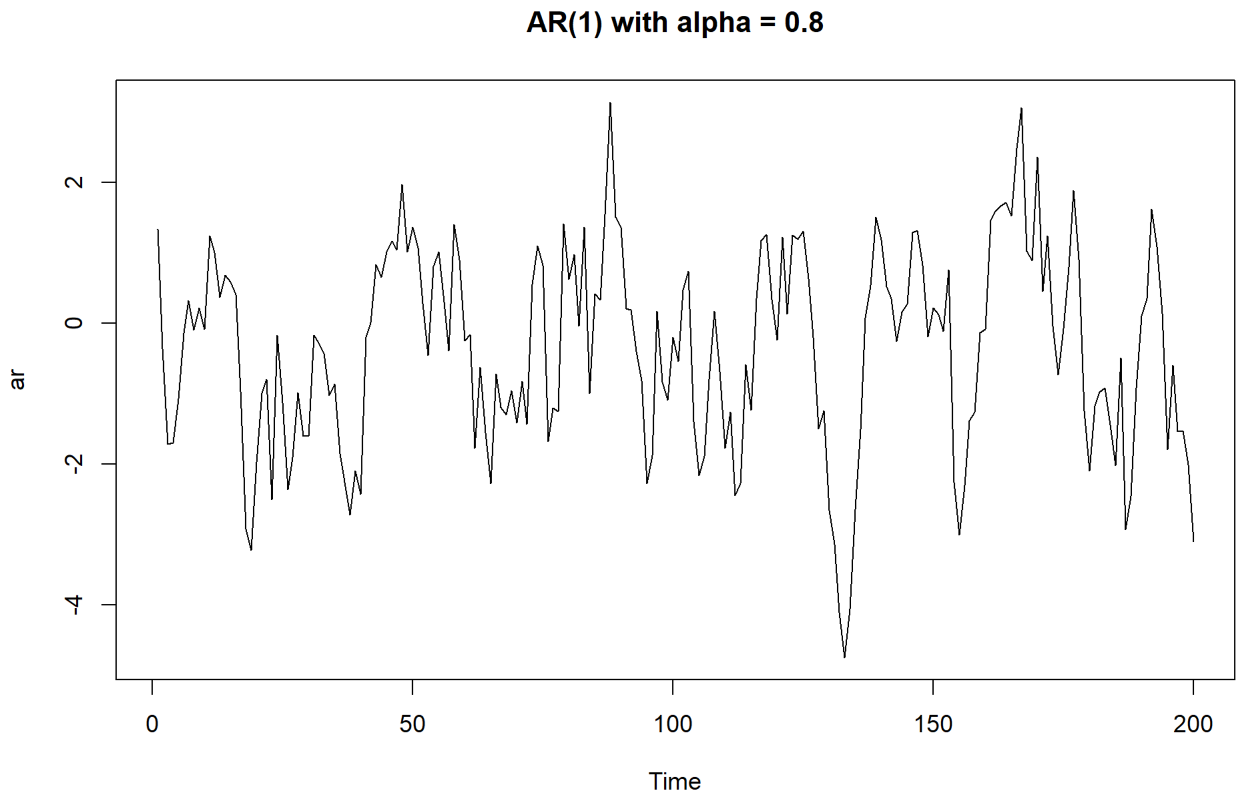
Autoregressive Processes, AR(p)

- let u_t be a white noise process with $E[u_t] = 0$ and $Var[u_t] = \sigma^2$, then an **autoregressive model of order** p, denoted as $AR(p)$, is defined as

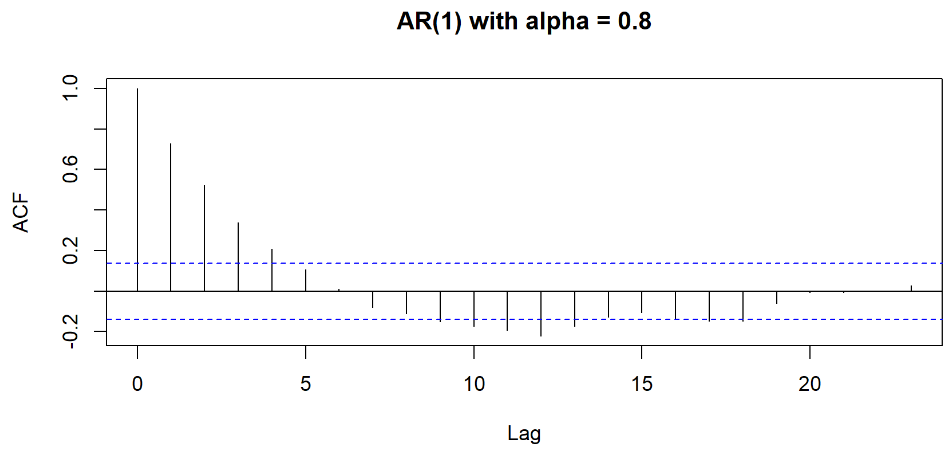
$$y_t = \alpha_0 + \alpha_1 y_{t-1} + \ldots + \alpha_p y_{t-p} + u_t$$

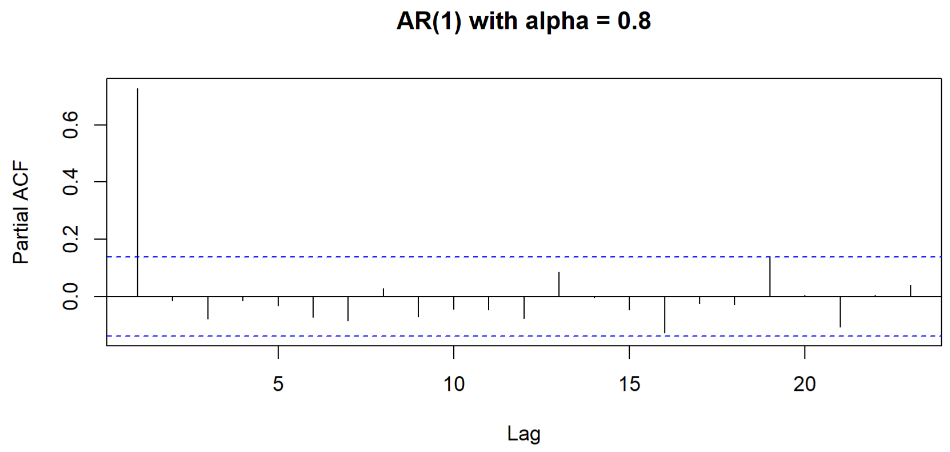
- i.e., an autoregressive model is a process where the current value of y depends only on the value of y in previous periods plus an error term

Autoregressive Processes, AR(p)



Autoregressive Processes, AR(p)





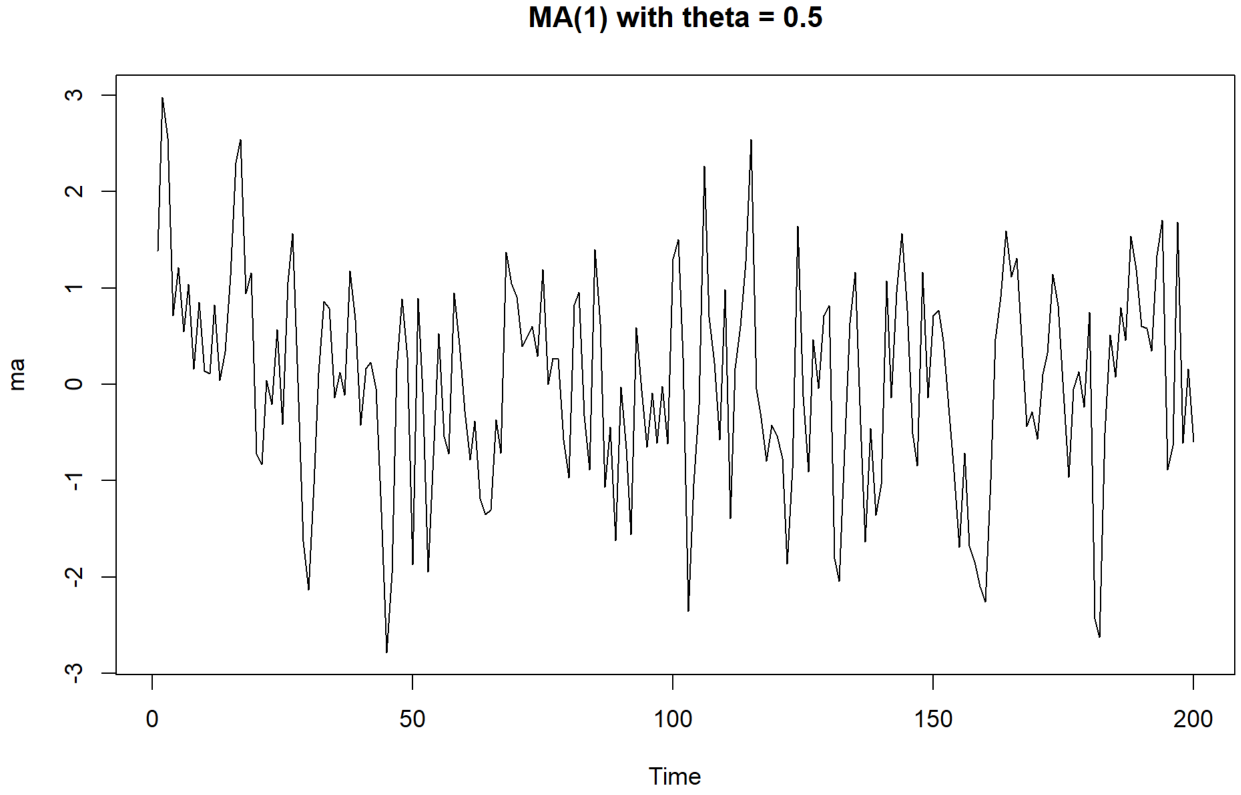
Moving Average Processes, MA(q)

- let u_t be a white noise process with $E[u_t] = 0$ and $Var[u_t] = \sigma^2$, then a q^{th} **order moving average processes**, denoted as $MA(q)$, is defined by

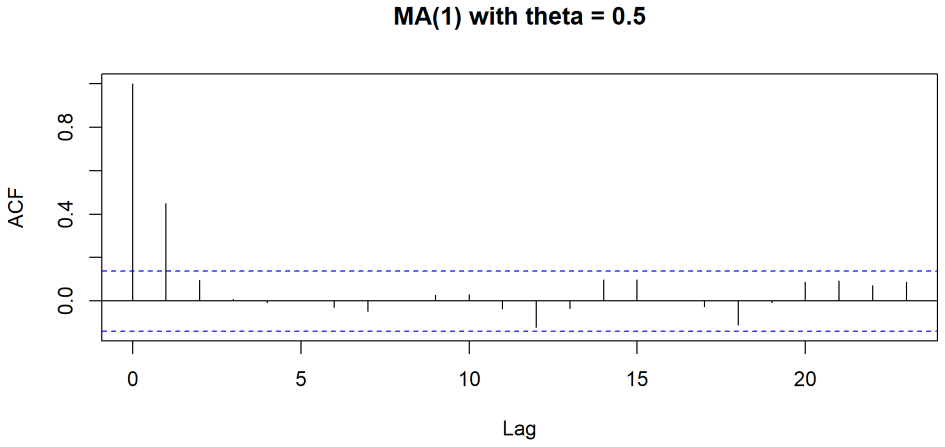
$$y_t = \mu + u_t + \theta_1 u_{t-1} + \theta_2 u_{t-1} + \ldots + \theta_q u_{t-q}$$

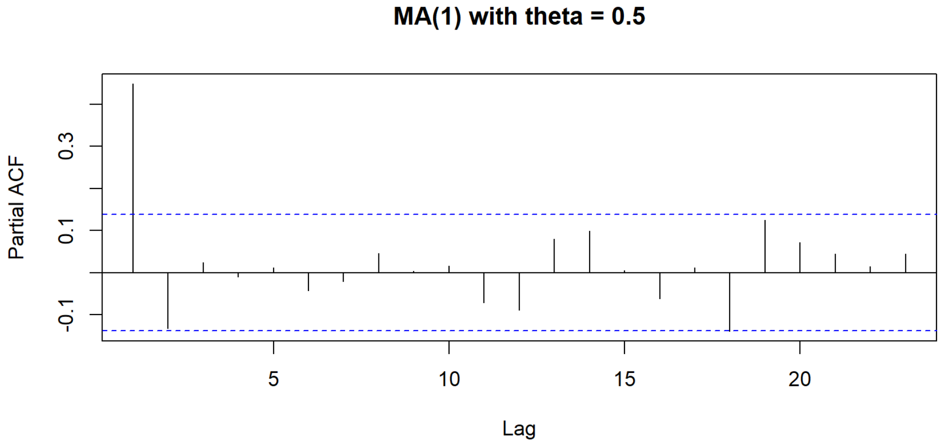
- i.e., a moving average model is simply a linear combination of white noise processes, such that y_t depends on the current and previous values of a white noise disturbance term

Moving Average Processes, MA(q)



Moving Average Processes, MA(q)





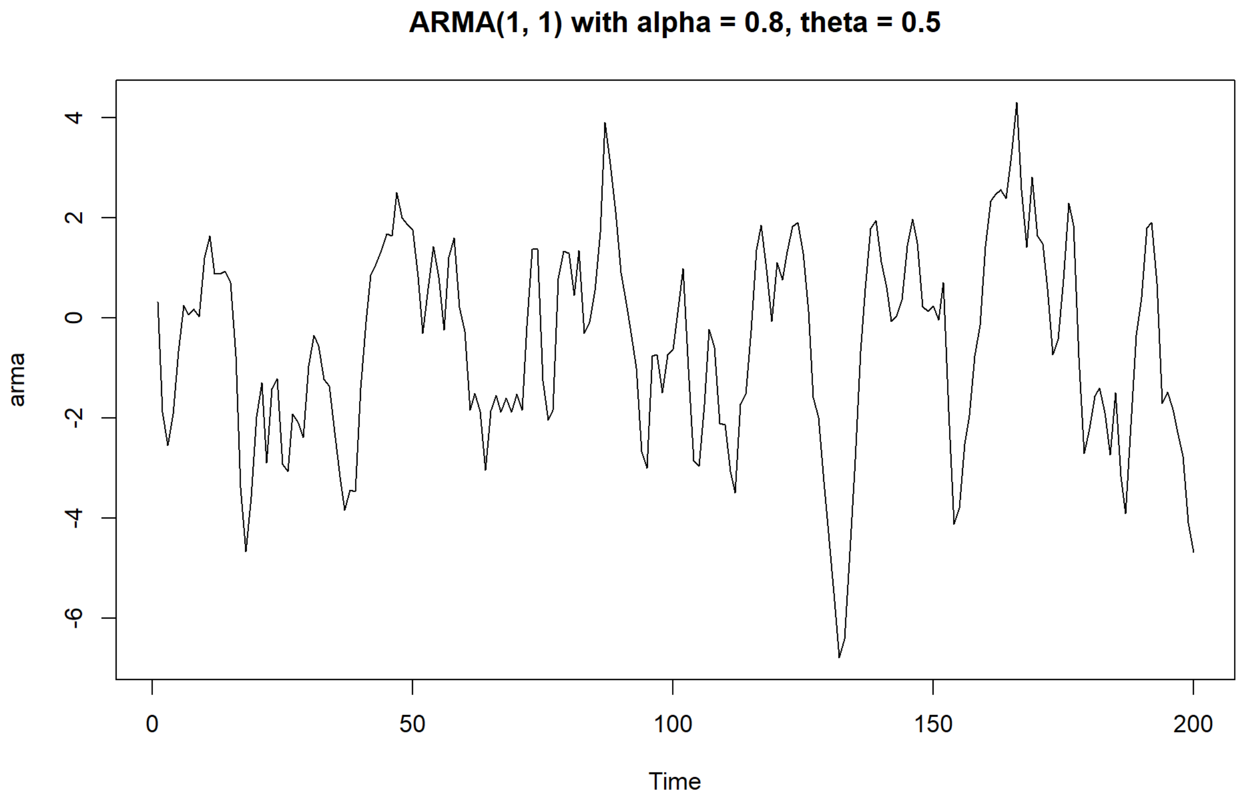
Autoregressive Moving Average Models (ARMA)

- autoregressive moving average models (ARMA) provide a parsimonious description of a weakly stationary stochastic process in terms of two polynomials, one for the autoregression and one for the moving average
- i.e., combining an AR(p) and a MA(q) model results in the **ARMA(p,q)** model

$$y_t = \alpha_0 + \alpha 1 y_{t-1} + \ldots + \alpha_p y_{t-q} + u_t + \theta_1 u_{t-1} + \ldots + \theta_q u_{t-q}$$

Note: if the data shows evidence of non-stationarity, the ARIMA generalization (autoregressive integrated moving average models) might be applied in order to eliminate non-stationarity by an initial differencing step

Autoregressive Moving Average Models (ARMA)



Model Identification

Model Identification

- the most crucial step in time series analysis is to identify and build a model based on the available data — i.e., to identify a process which best explains the observed movements over time

- the aim is to identify a process that leaves the residuals not being different from a white noise process (i.e., no exploitable information is contained in residuals)
- in order to statistically examine whether a process is significantly different from white noise, several tests are available (e.g., *portmanteau Q test*, *Bartlett's test*, *Box-Pierce test*, or *Ljung-Box test*)

Model Identification

Box-Jenkins Approach for Modelling Stochastic Processes

1. stationarity, model specification: check whether the series is stationary, determine the order (p, q) of the model, apply graphical methods and (partial) autocorrelation function

2. estimation: estimate the parameters using non-linear least squares or maximum likelihood (ordinary least squares will be biased and/or inefficient)

3. diagnosis, model quality: examine residuals and check the model quality (information criteria)

Aim: establish a model as parsimonious as possible because the variance of the estimates is inverse proportional to the number of degrees of freedom, i.e., parameters in the model to be estimated

Model Identification

ACF and PACF

process	**autocorrelation ρ**	**partial autocorrelation ρ^***
$MA(q)$	number of ρ's sig. different from zero determines order of MA process	decreasing
$AR(p)$	decreasing	number of ρ^*'s sig. different from zero determines order of AR process
$ARMA(p,q)$	decreasing	decreasing

Model Identification

Information Criteria

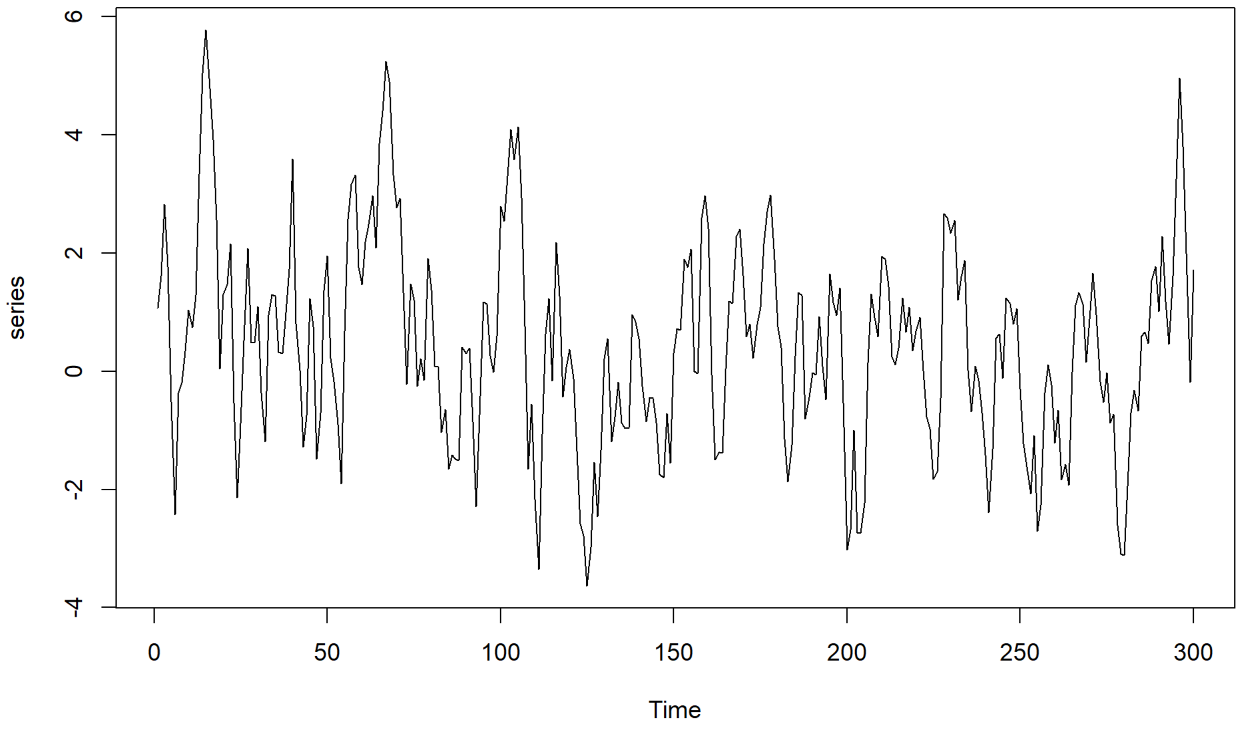
- generally, information criteria can be considered as goodness of fit measures: thus, information criteria are estimators of the *relative quality* of statistical models for a given set of data, providing a means for model selection
- information criteria estimate the relative information lost when a given model is used to represent the process generating the data (trade-off between simplicity and goodness-of-fit of the model)

- the smaller the estimator of an information criteria, the smaller the information loss and, thus, the better the particular model is relative to other models which are compared

- frequently applied information criteria are:
  - the Akaike information criteria (AIC),
  - the Bayesian information criteria (BIC), and
  - the Hannan-Quinn information criteria (HQIC)

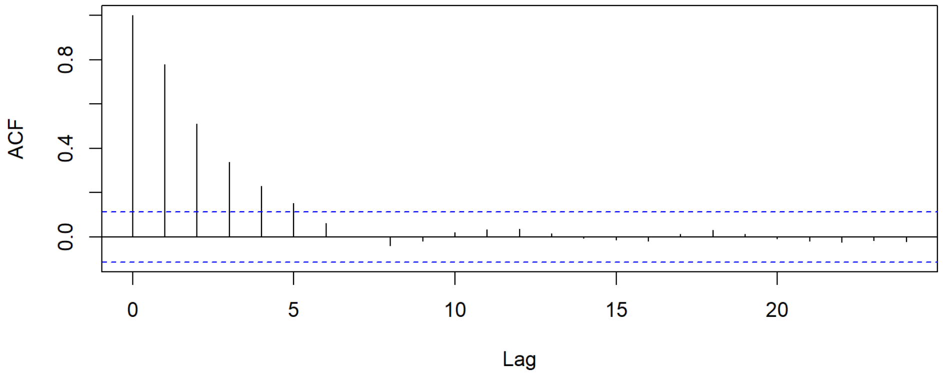
Model Identification

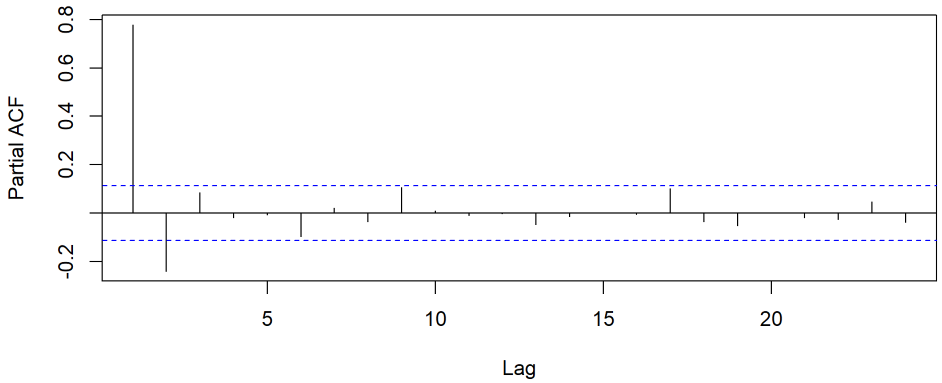
Example



Model Identification

Example





Model Identification

Example

- **test for stationarity:**
  - augmented Dickey-Fuller test: $z(t) = -5.4346, p = 0.01$
- **model selection:**
  - choose suitable models based on (partial) autocorrelation function
  - reasonable candidates seem to be $AR(1)$, $MA(1)$, and $ARMA(1,1)$

Model Identification

Example

- **model estimation:**

```r
arima(series, order = c(1, 0, 0))
```

```
##
## Call:
## arima(x = series, order = c(1, 0, 0))
##
## Coefficients:
##          ar1  intercept
##       0.7774     0.4637
## s.e.  0.0360     0.2761
##
## sigma^2 estimated as 1.16:  log likelihood = -448.36,  aic = 902.72
```

```r
arima(series, order = c(0, 0, 1))
```

```
##
## Call:
## arima(x = series, order = c(0, 0, 1))
##
## Coefficients:
##          ma1  intercept
##       0.7719     0.4497
```

Model Identification

```r
library(forecast)
```

```
## Registered S3 methods overwritten by 'forecast':
##   method             from
##   fitted.fracdiff    fracdiff
##   residuals.fracdiff fracdiff
```

```r
auto.arima(series, trace = TRUE)
```

```
##
##  Fitting models using approximations to speed things up...
##
##  ARIMA(2,1,2) with drift         : 900.2311
##  ARIMA(0,1,0) with drift         : 932.2426
##  ARIMA(1,1,0) with drift         : 931.5384
##  ARIMA(0,1,1) with drift         : 928.1396
##  ARIMA(0,1,0)                    : 930.2166
##  ARIMA(1,1,2) with drift         : 888.4378
##  ARIMA(0,1,2) with drift         : 913.231
##  ARIMA(1,1,1) with drift         : 925.2665
##  ARIMA(1,1,3) with drift         : 890.4646
##  ARIMA(0,1,3) with drift         : 906.2779
##  ARIMA(2,1,1) with drift         : 903.8821
##  ARIMA(2,1,3) with drift         : 895.117
##  ARIMA(1,1,2)                    : 887.228
##  ARIMA(0,1,2)                    : 911.1791
##  ARIMA(1,1,1)                    : 923.2353
##  ARIMA(2,1,2)                    : 898.3981
```