MW82: Time Series Analysis

Dr. Giulia Sabbadini

DICE

WS 2023/24

Syllabus

- Time: Friday 12:30 - 14:00
- Place: Seminarraum 2 (Oeconomicum)
- probably 8 tutorials
- Contact: **sabbadini@dice.hhu.de**
- everybody on ILIAS?

Time Table

Date	Tutorial	Topic
Oct 13	1	Introduction
Oct 20	-	*No Tutorial*
Oct 27	2	AR, ACF
Nov 3	3	MA, PACF
Nov 10	4	ARIMA, Model Selection
Nov 17	-	*No Tutorial*
Nov 24	5	Forecasting / ARIMA
Dec 1	6	Non-stationarities
Dec 8	-	*No Tutorial*
Dec 15	7	Multiple Time Series (VAR I)
Dec 22	-	*No Tutorial*
Jan 12	8	Multiple Time Series (VAR II)

Introduction to Time Series

Introduction

- Three types of datasets studied in econometrics:
  - Cross-sectional: collected at one given time across multiple entities (countries, industries, firms)
  - Time series: any set of data that registers periodically; e.g. yearly GDP for 20 years, monthly firm sales
  - Panel: combination of both
- Time series: use natural one-way ordering of time, values can be expressed as deriving from past value (rather than future values)

Introduction to Time Series

- A set of data ordered by time forms a time series, $\{y_t\}_{t=1}^{T}$
- This implies special approaches to their analysis
  - time ordering enables estimation as a function of its past values (lags)
  - for example: variable is regressed on its own past values, called autoregressive (AR) process
  - autocorrelation: strenght of the relationship across time of the same random variable

Introduction to Time Series

- A set of data ordered by time forms a time series, $\{y_t\}_{t=1}^{T}$
- This implies special approaches to their analysis
  - time ordering enables estimation as a function of its past values (lags)
  - for example: variable is regressed on its own past values, called autoregressive (AR) process
  - autocorrelation: strenght of the relationship across time of the same random variable
- Example stochastic process, simple one lag/AR(1) process: $y_t = 0.5y_{t-1} + \varepsilon_t$
- with initial condition $y_0 = 0$, we can say:
  - Random variables of this process: $\varepsilon_1,\ \ 0.5\varepsilon_1 + \varepsilon_2,\ \ 0.25\varepsilon_1 + 0.5\varepsilon_2 + \varepsilon_3,\ \ \ldots$
- Goal: 'estimate' a time series, i.e. based on the data (concrete realization) we estimate the underlying process

Some definitions

- Time series that show no autocorrelation are called white noise
- White noise
  - uncorrelated random variables (i.i.d.) with zero mean and constant variance
  - for example $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$
  - can not be predicted, is truly random
  - residuals from correctly specified (true) model are white noise
  - Ideally: residuals from our model should be white noise; if not, implies we do not fully capture the non-random/predictable part of process

Example: White Noise

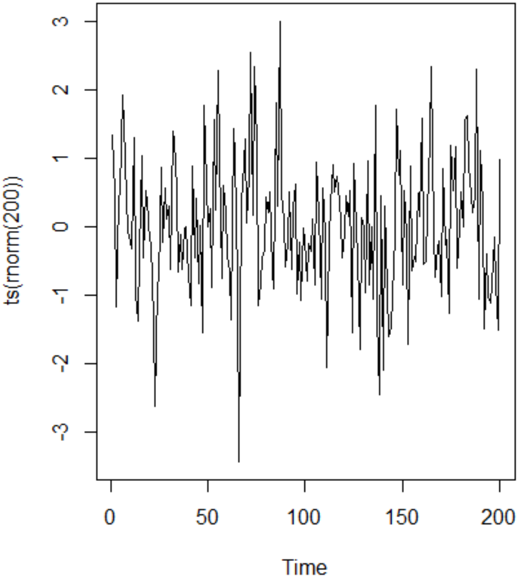

Figure: White Noise from standard Normal distribution

Describing a Time Series

- Trend
  - A trend exists when there is a long-term increase or decrease in the data. It does not have to be linear.
- Seasonal
  - A seasonal pattern occurs when a time series is affected by seasonal factors such as the time of the year or the day of the week. Seasonality is always of a fixed and known frequency.
- Cyclic
  - A cycle occurs when the data exhibit rises and falls that are not of a fixed frequency. These fluctuations are usually due to economic conditions, and are often related to the "business cycle".

Example: Components (1/2)

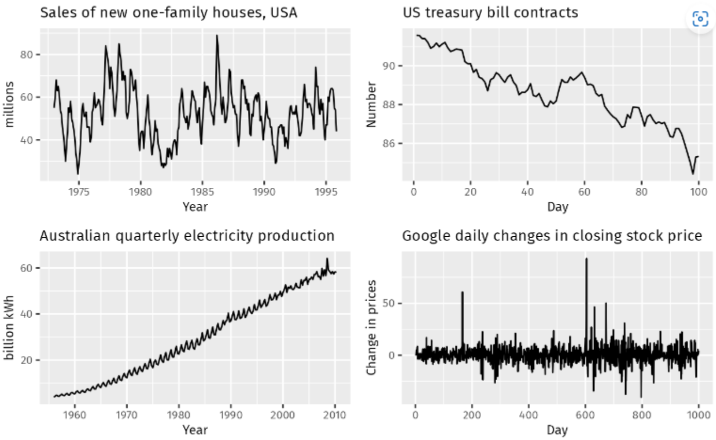

Figure: Time series components. Source: Forecasting: Principles and Practice

Example: Components (2/2)

- The monthly housing sales (top left) show strong seasonality within each year, as well as some strong cyclic behaviour with a period of about 6–10 years. There is no apparent trend in the data over this period.
- The US treasury bill contracts (top right) show results from the Chicago market for 100 consecutive trading days in 1981. Here there is no seasonality, but an obvious downward trend.
- The Australian quarterly electricity production (bottom left) shows a strong increasing trend, with strong seasonality. There is no evidence of any cyclic behaviour here.
- The daily change in the Google closing stock price (bottom right) has no trend, seasonality or cyclic behaviour. There are only random fluctuations.

Stationarity

- Desirable property of a time series
- important for properties of estimators, forecasting, ...
- A stationary process is identically distributed across time (same mean, same variance, ...)
- If it exhibits "similar" behavior, one can then proceed with the modeling efforts
- If a time series is stationary, then any shock that occurs in time t has a diminishing effect over time and finally disappears
- AR(1) process $y_t = \phi y_{t-1} + \varepsilon_t$ is stationary if $|\phi| < 1$. Intuition? ε_t is multiplied with value below one in all future periods, simply vanishes over time.

Stationarity

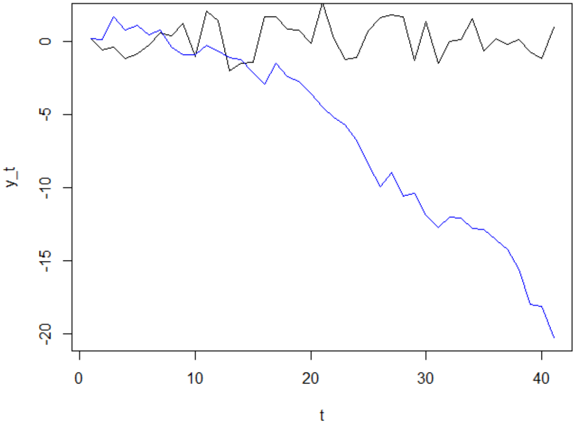

Figure: Stationary (black): $y_t = 0.25y_{t-1} + \varepsilon_t$ and non-stationary (blue) $y_t = 1.05y_{t-1} + \varepsilon_t$

Common threats to stationarity

- Trends
- Seasonal patterns
- Level changes
- Changes in Variance

Examples

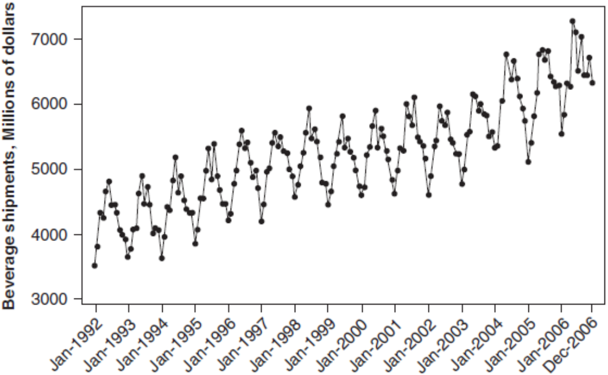

Figure: The US beverage manufacturer monthly product shipments, unadjusted. Source: Montgomery, Jennings, Kulahci (2015)

How to deal with non-stationary time series?

Transform the data:

- taking logs (works e.g. with exponential growth such as GDP)
- differencing: $\Delta y_t = y_t - y_{t-1}$, usually only first difference (observation and information loss)
- detrending: remove trend (linear, quadratic, ...)
- remove seasonal components

Starting tool: Decomposition into components

- series consists of three basic components: trend, seasonal component and irregular component
- $y_t = T_t + S_t + I_t$

Decomposition

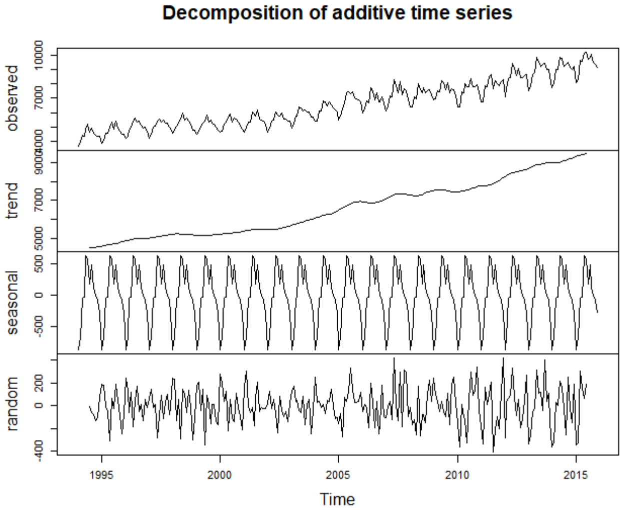

Figure: Beverage sales decomposed.

Introduction into R

Introduction

What is R?

- **Free**, cross platform, open-source software for data analysis
- Used by academics, companies, and governments
- Use command line interface to try to learn by imitation and trial-and-error!
- You can type commands and let R react

Example:

Meaning	R-Syntax	Result
$6+3$	`6 + 3`	9
$(2+\sqrt{6+3}) \cdot 3$	`(2 + sqrt(6 + 3))*3`	15

What is RStudio?

- **Free**, pretty and useful user interface for R.

Important:

- Variables are assigned with `<-` , i.e. `var1 <-5.5` (= also works)
- R is (usually) vectorized!

Types:

- Numeric/Integer: `2.456` , sequence of integers `1:10`
- String: `"Max"` or `'Max'`
- Logical: `TRUE` and `FALSE`
- Factor: Any variable with levels (e.g. high/low)
- List: `list("name"="Max","age"=20)`
- Functions: `func1 <-function(x){x+1}` , call with e.g. `func1(5)` which yields 6

Storing data:

- Vector: `c(1,2,3,4,-2)` or `c("düsseldorf","köln")`
- Matrix
  - `matrix(1:9, ncol=3, byrow=T)`
  - `rbind( c(1,2,3), c(4,5,6), c(7,8,9))`

```
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

- data.frame
  - columns (named) and row structure
  - `data1 <-read.csv("mydata.csv")`
- access data
  - unnamed (matrix, vector) with: `c(-4,5,-6)[2]` or `matrix1[2,2]`
  - named (data.frame, list) with $: `data1$varname`

Other stuff

Set working dictionary

```
setwd('path/to/my/working/directory')
```

Libraries/Packages:
R has a core to which you can add contributed packages. To install and load you can use the GUI or:

```
install.packages('xlsx')
library('xlsx')
```

Use functions of an installed package directly (without loading the entire library):

```
xlsx::read.xlsx()
```

Need help with a function, such as rnorm?

```
?rnorm
help(rnorm)
```

Linear Regressions

Estimating linear models (defaults to OLS):

```
#bivariate
lm(y ~ x)
#multivariate
lm(y ~ x1 + x2 + x3)
#without intercept
lm(y ~ 0+x)
#actual use
lmres <- lm(y ~ x)
summary(lmres) #results
coef(lmres)
```

Here, y and x are previously defined vectors. If you have a data.frame *mydata* with columns *sales* and *advertising*, use:

```
lm(sales ~ advertising, data = mydata)
```

Useful commands

Plot something: `plot()`
Read a CSV file:

```
df1 <- read.csv("monthly_gdp.csv")
```

Define a `ts` (time series) object.

```
gdpts <- ts(df1$monthly_gdp, start =
    2002, frequency = 12)
```

Some things to do to a `ts` object.

```
plot.ts(gdpts)
monthplot(gdpts)
diff(gdpts)
decompose(gdpts)
```

Exercise I

1. Generate 200 observations of X, a bivariate normal distribution with zero mean and covariance matrix[1] $\mathbf{K} = \begin{pmatrix} 1 & 0.2 \\ 0.2 & 1 \end{pmatrix}$
2. Generate a vector of standard normally distributed random errors ε of length 200
3. Assume the true parameters are $\beta_0 = 2, \beta_1 = 0.4$ and $\beta_2 = -0.8$. Now create the Y variable:
   $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$
4. Estimate a linear model with OLS. Interpret your results. What do you observe from the diagnostic plots?
5. Repeat the exercise but assume that the error is uniformly distributed between 0 and 1.

[1] Hint: Use the *mvrnorm* function from the MASS package

Exercise II

1. Load the beverage data set 'bev_data.csv' in R.
2. Rename the columns and create a date-month index variable.
3. Transform the series into a time series *ts* object and plot the series.
4. Do you observe seasonality? Check with a monthly plot.
5. First difference the series and plot again.
6. Decompose the series into its components and plot your results.
7. Estimate a linear trend model via OLS. What do you observe in your diagnostics plots?