Applied Time Series Analysis
SS 2014 – Week 02

Marcel Dettling

Institute for Data Analysis and Process Design

Zurich University of Applied Sciences

marcel.dettling@zhaw.ch

http://stat.ethz.ch/~dettling

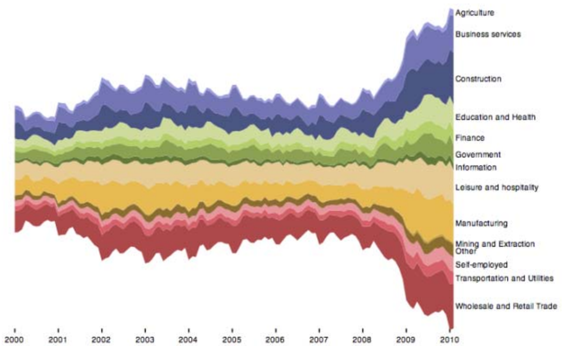
ETH Zürich, February 24, 2014

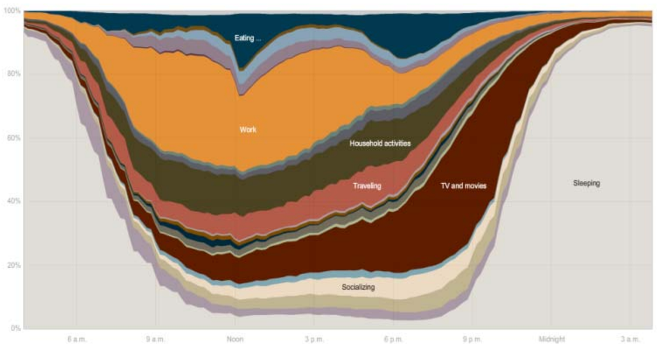
Applied Time Series Analysis
SS 2014 – Week 02

Descriptive Analysis

As always, when working with data, it is important to first gain an overview. In time series analysis, the following is required:

- *Understanding the context of the data and the data source*
- *Making suitable plots, looking for structure and outliers*
- *Thinking about transformations, e.g. to reduce skewness*
- *Judging stationarity and achieve it by decomposition*
- *For stationary series, the analysis of autocorrelations*

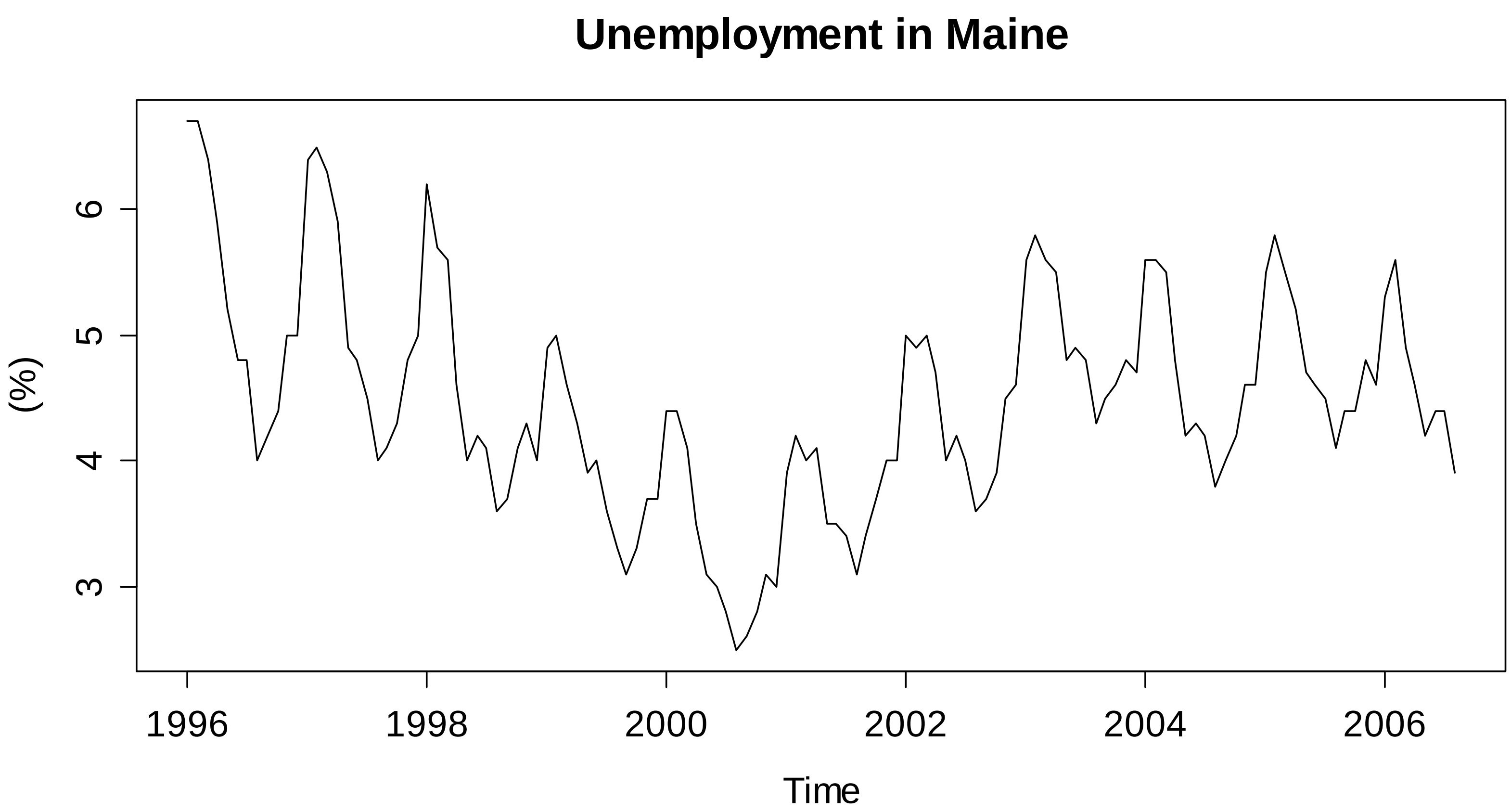




Applied Time Series Analysis
SS 2014 – Week 02

Visualization: Time Series Plot

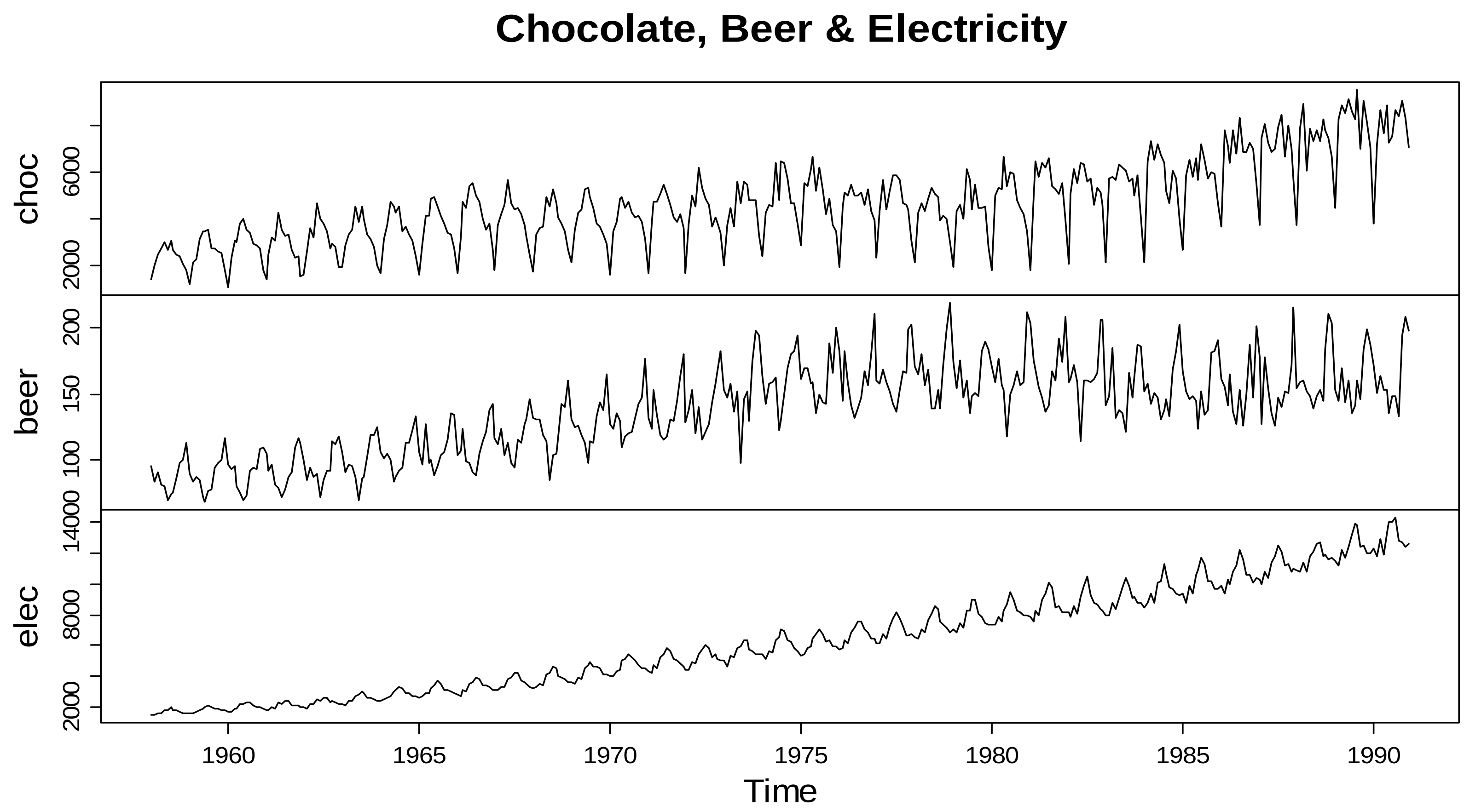
```
> plot(tsd, ylab="(%)", main="Unemployment in Maine")
```



Applied Time Series Analysis
SS 2014 – Week 02

Multiple Time Series Plots

```
> plot(tsd, main="Chocolate, Beer & Electricity")
```



Only One or Multiple Frames?

- Due to different scale/units it is often impossible to directly plot multiple time series in one single frame. Also, multiple frames are convenient for visualizing the series.

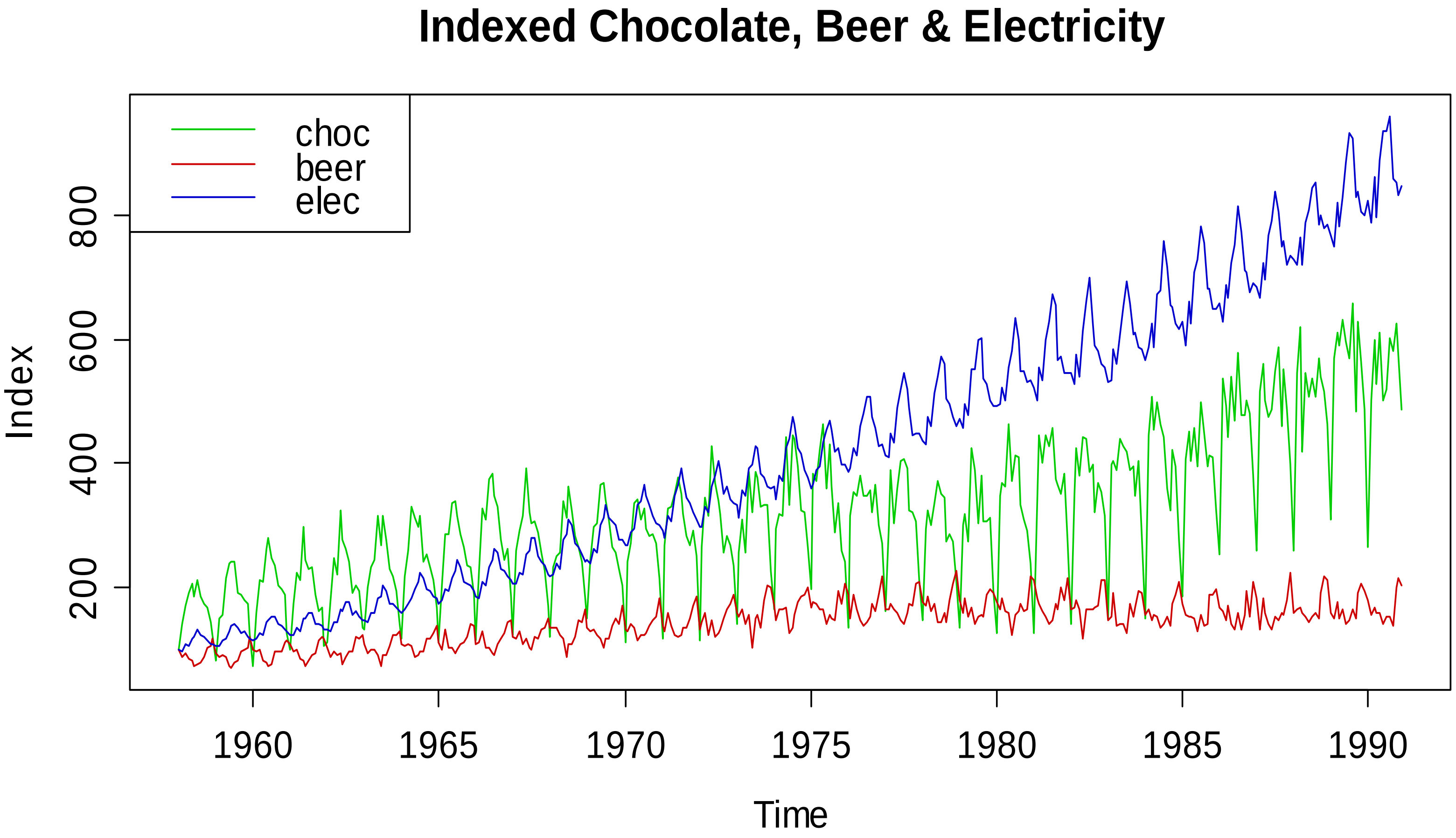
- If the relative development of multiple series is of interest, then we can (manually) index the series and (manually) plot them into one single frame.

- This clearly shows the magnitudes for trend and seasonality. However, the original units are lost.

- For details on how indexing is done, see the scriptum.

Applied Time Series Analysis
SS 2014 – Week 02

Multiple Time Series Plots



Transformations

For strictly stationary time series, we have: $X_t \sim F$

We did not specify the distribution F and there is no restriction to it. However, many popular time series models are based on:

1) *Gaussian distribution*

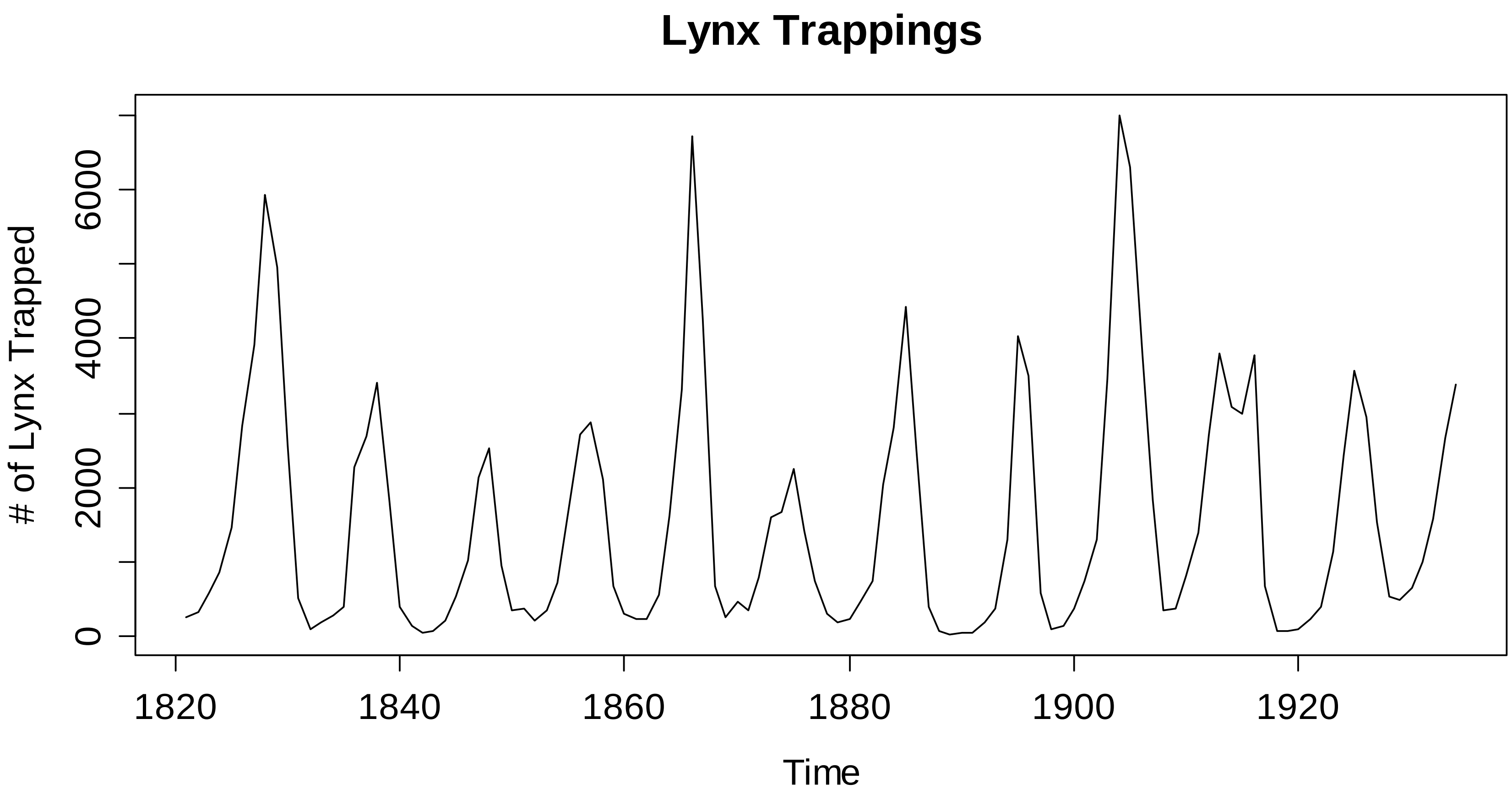
2) *linear relations between the variables*

If the data show different behaviour, we can often improve the situation by transforming $x_1,...,x_n$ to $g(x_1),...,g(x_n)$. The most popular and practically relevant transformation is:

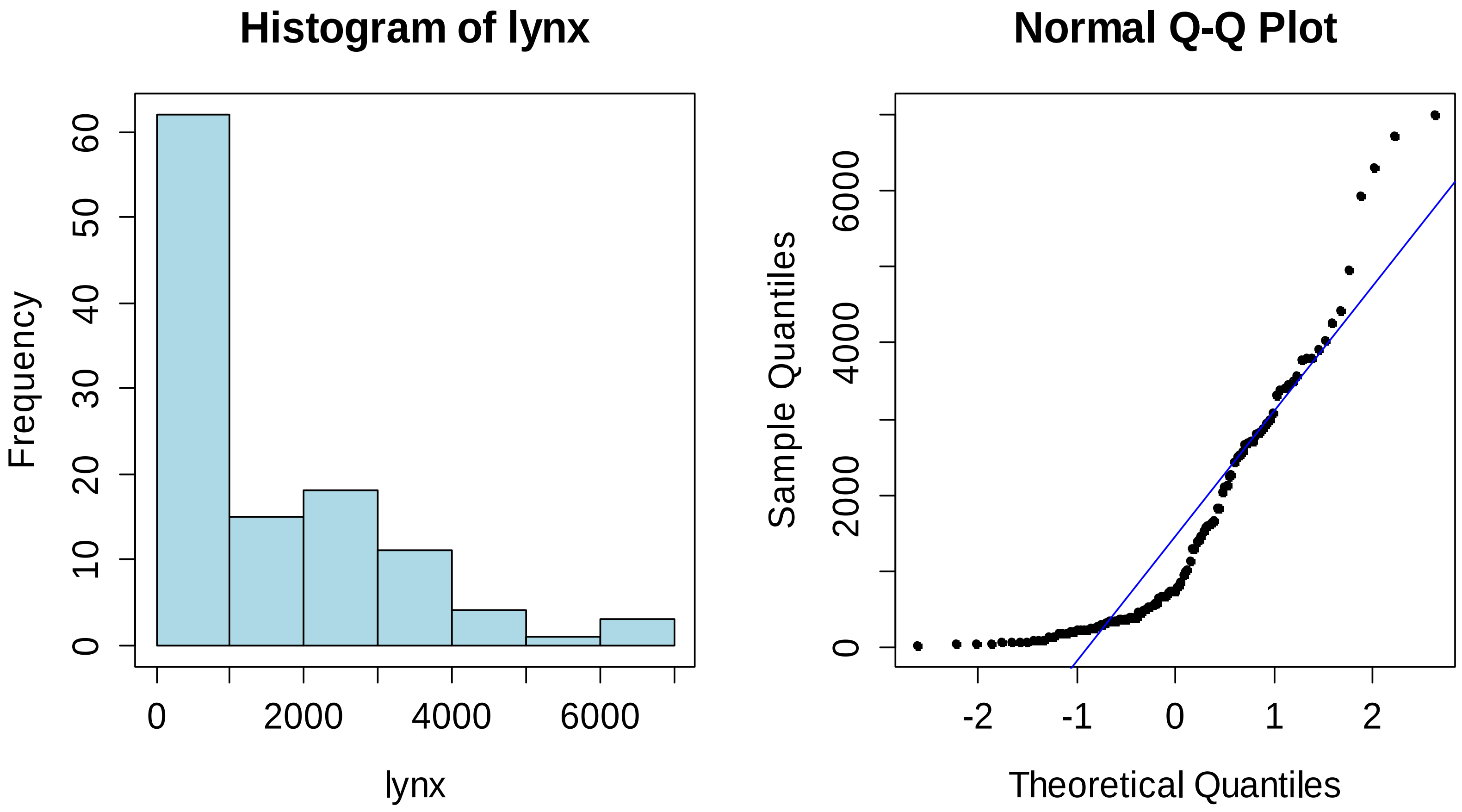
$$g(\cdot) = \log(\cdot)$$

Applied Time Series Analysis
SS 2014 – Week 02

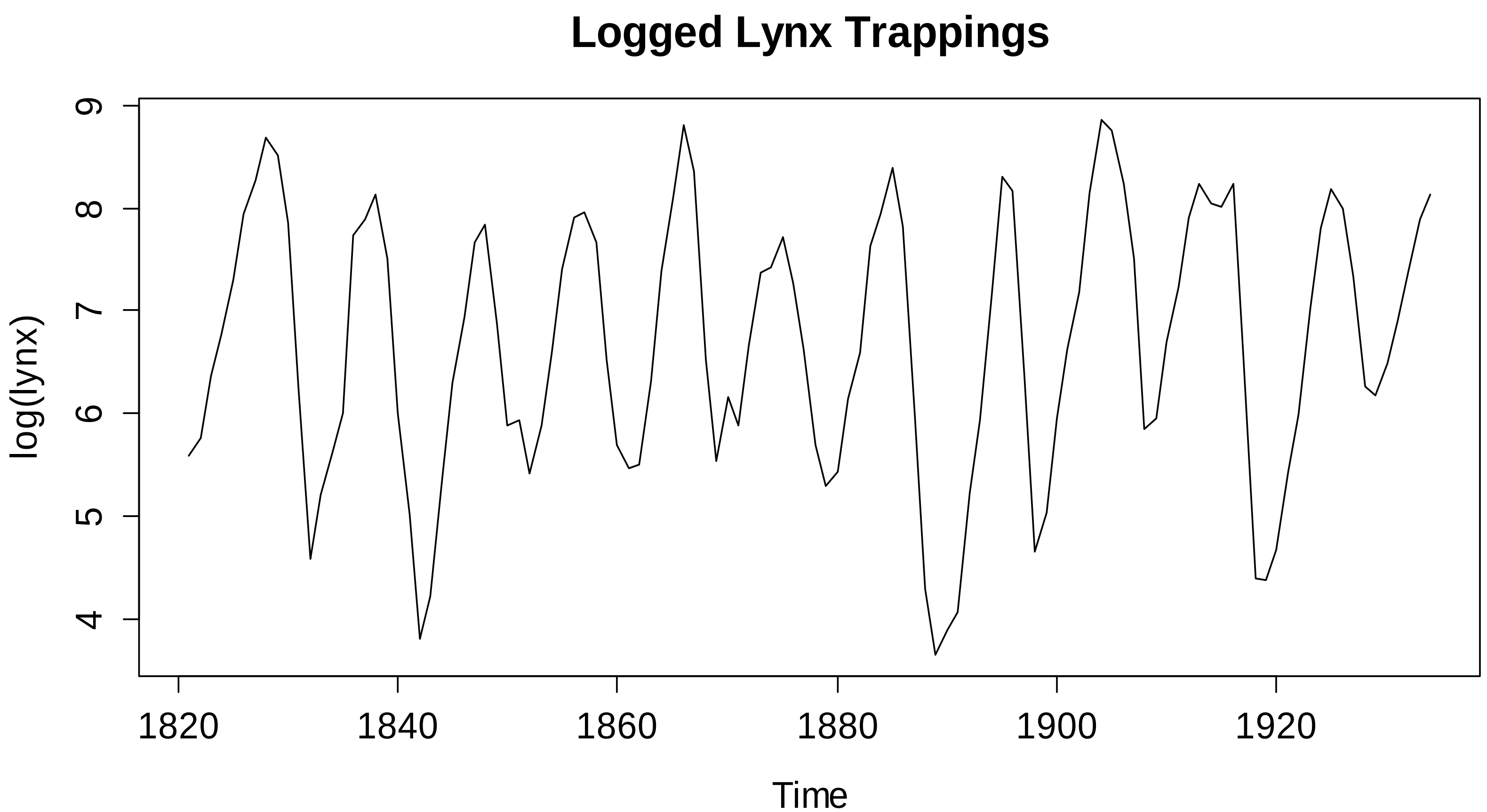
Transformations: Lynx Data



Transformations: Lynx Data



Transformations: Lynx Data



Decomposition

Stationarity is key for statistical learning, but real data often have trend/seasonality, and are non-stationary. We can (often) deal with that using the simple additive decomposition model:

$$X_t = m_t + s_t + R_t$$

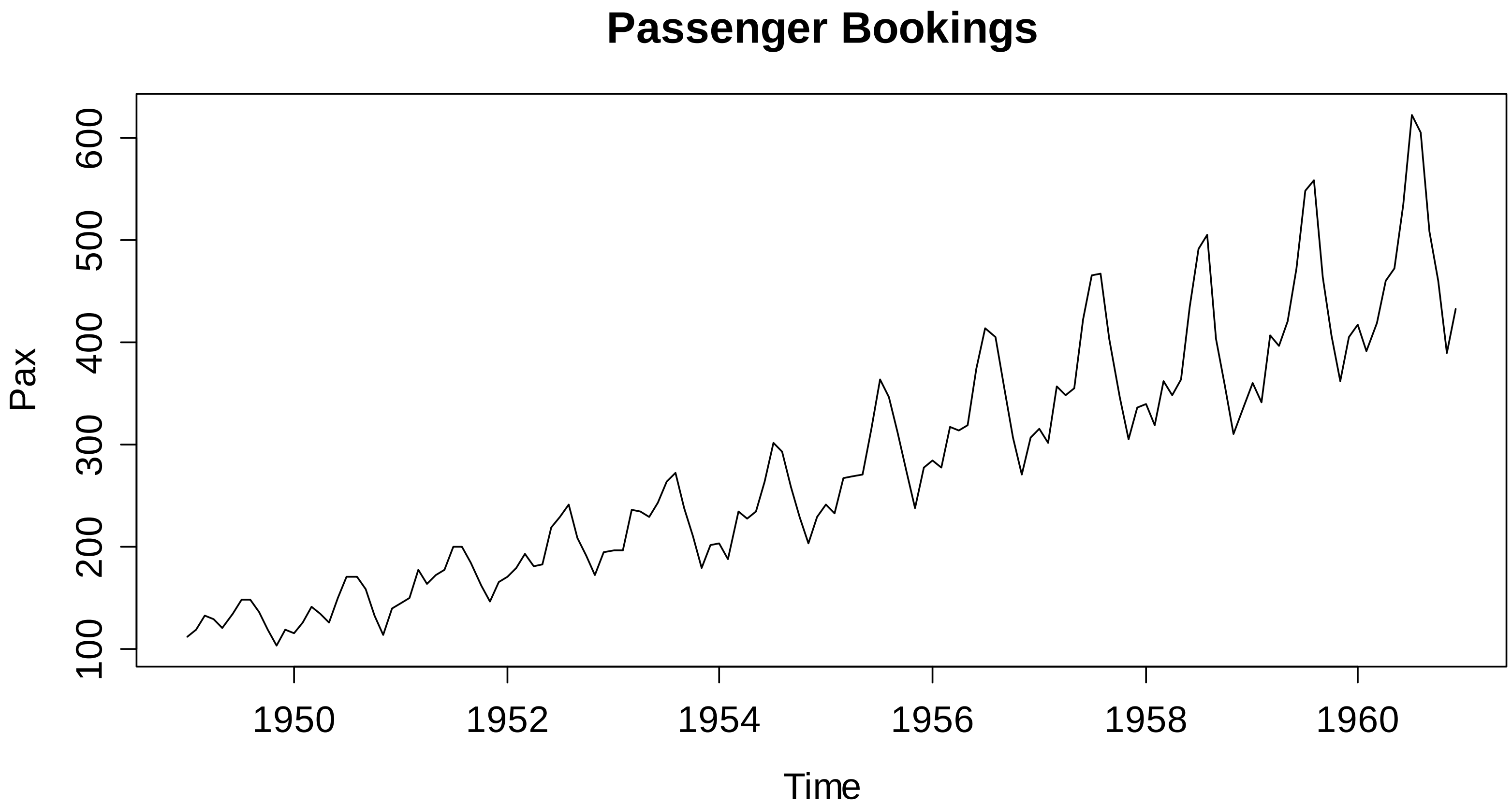
= trend + seasonal effect + stationary remainder

The goal is to find a remainder term R_t, as a sequence of correlated random variables with mean zero, i.e. a stationary ts.

We can employ: 1) *taking differences (=differencing)*
2) *smoothing approaches (= filtering)*
3) *parametric models (= curve fitting)*

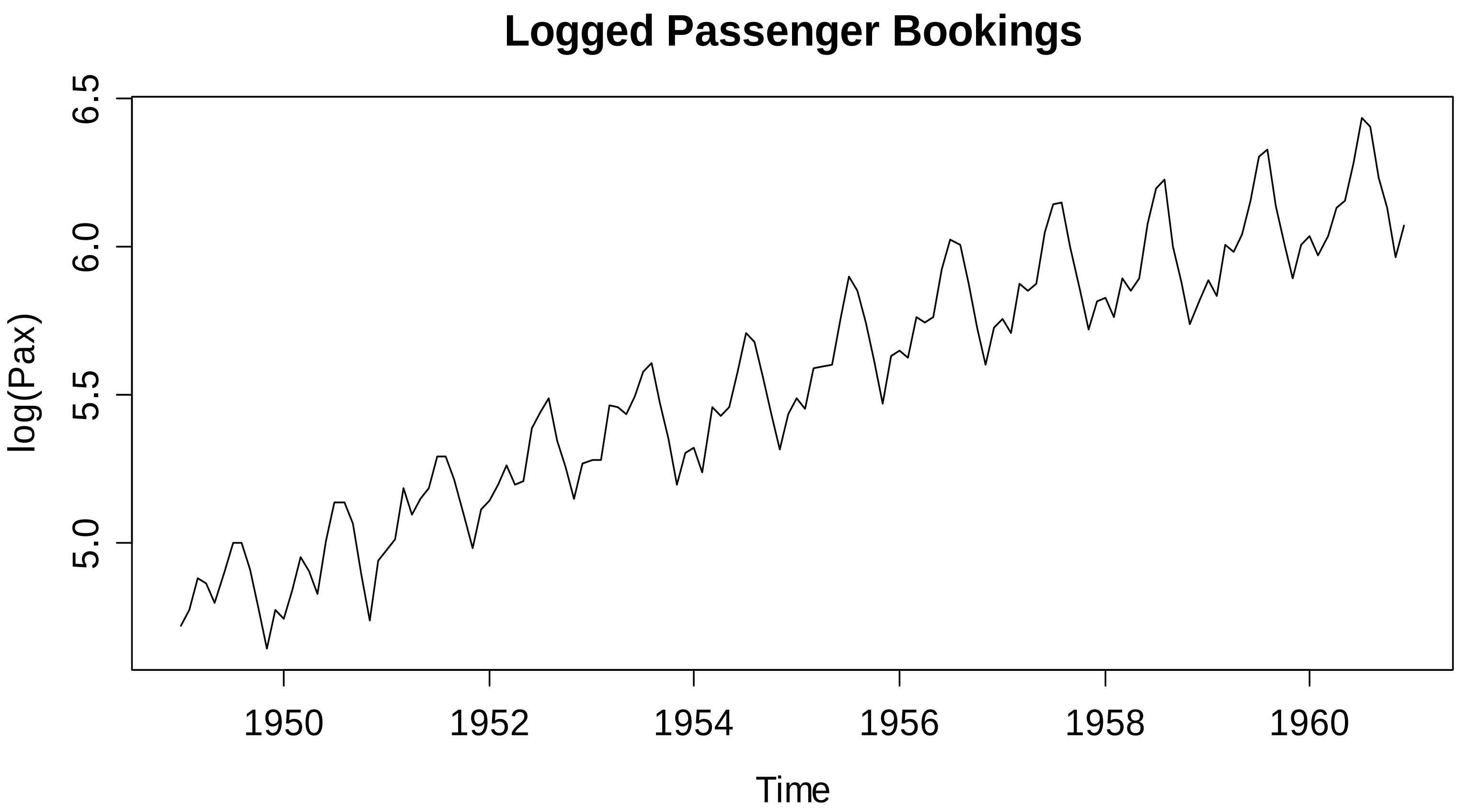
Multiplicative Decomposition

$X_t = m_t + s_t + R_t$ is not always a good model:



Multiplicative Decomposition

Better: $X_t = m_t \cdot s_t \cdot R_t$, respectively $\log(X_t) = m'_t + s'_t + R'_t$



Differencing: Removing a Trend

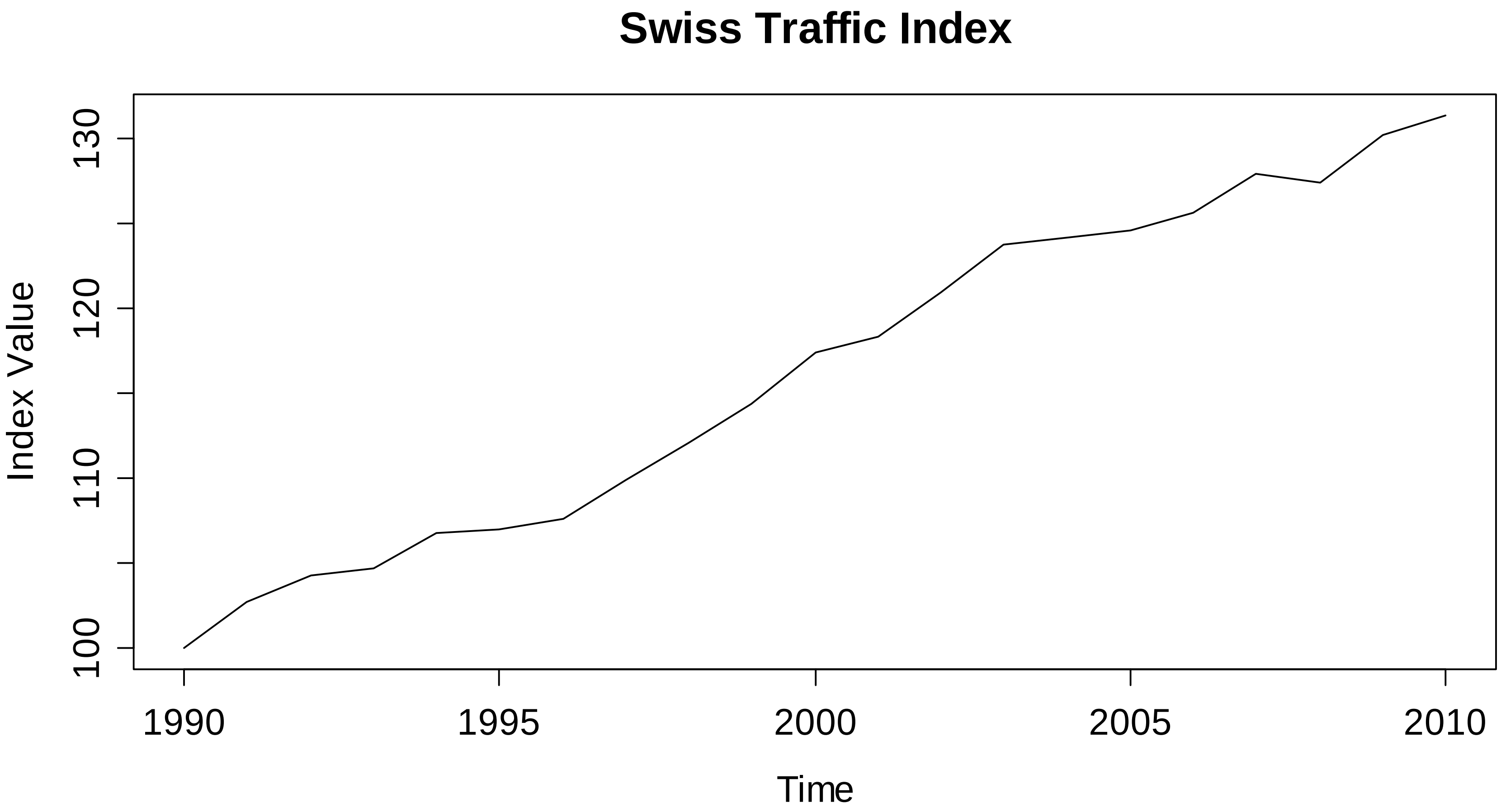
→ **see blackboard…**

Summary:

- Differencing means analyzing the observation-to-observation changes in the series, but no longer the original.
- This may (or may not) remove trend/seasonality, but does not yield estimates for m_t and s_t, and not even for R_t.
- Differencing changes the dependency in the series, i.e it artificially creates new correlations.

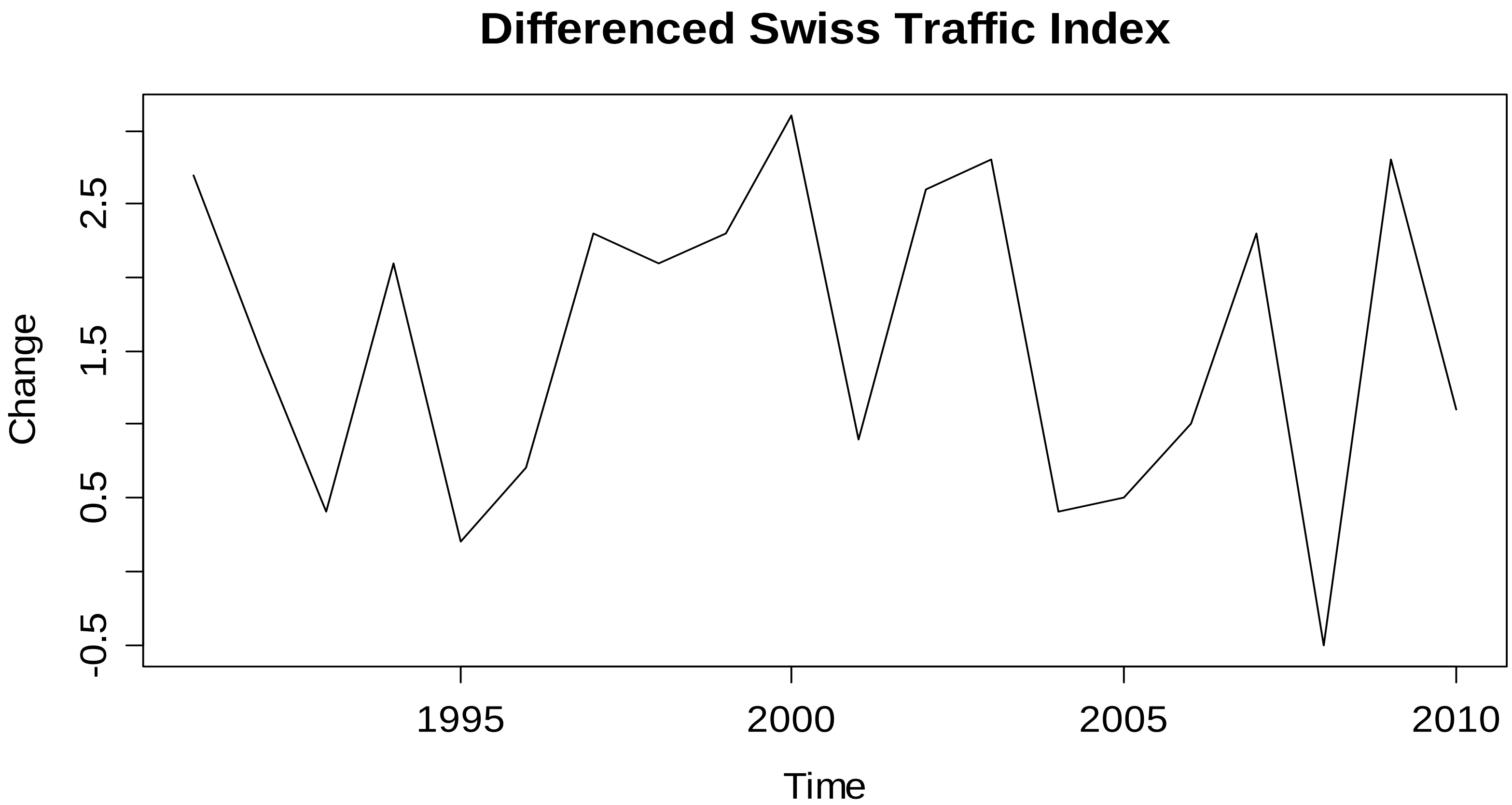
Applied Time Series Analysis
SS 2014 – Week 02

Differencing: Example



Differencing: Example

```
> plot(diff(SwissTraffic), main=…)
```



Differencing: Further Remarks

- If log-transformed series are difference (i.e. the SMI series), we are considering (an approximation to) the relative changes:

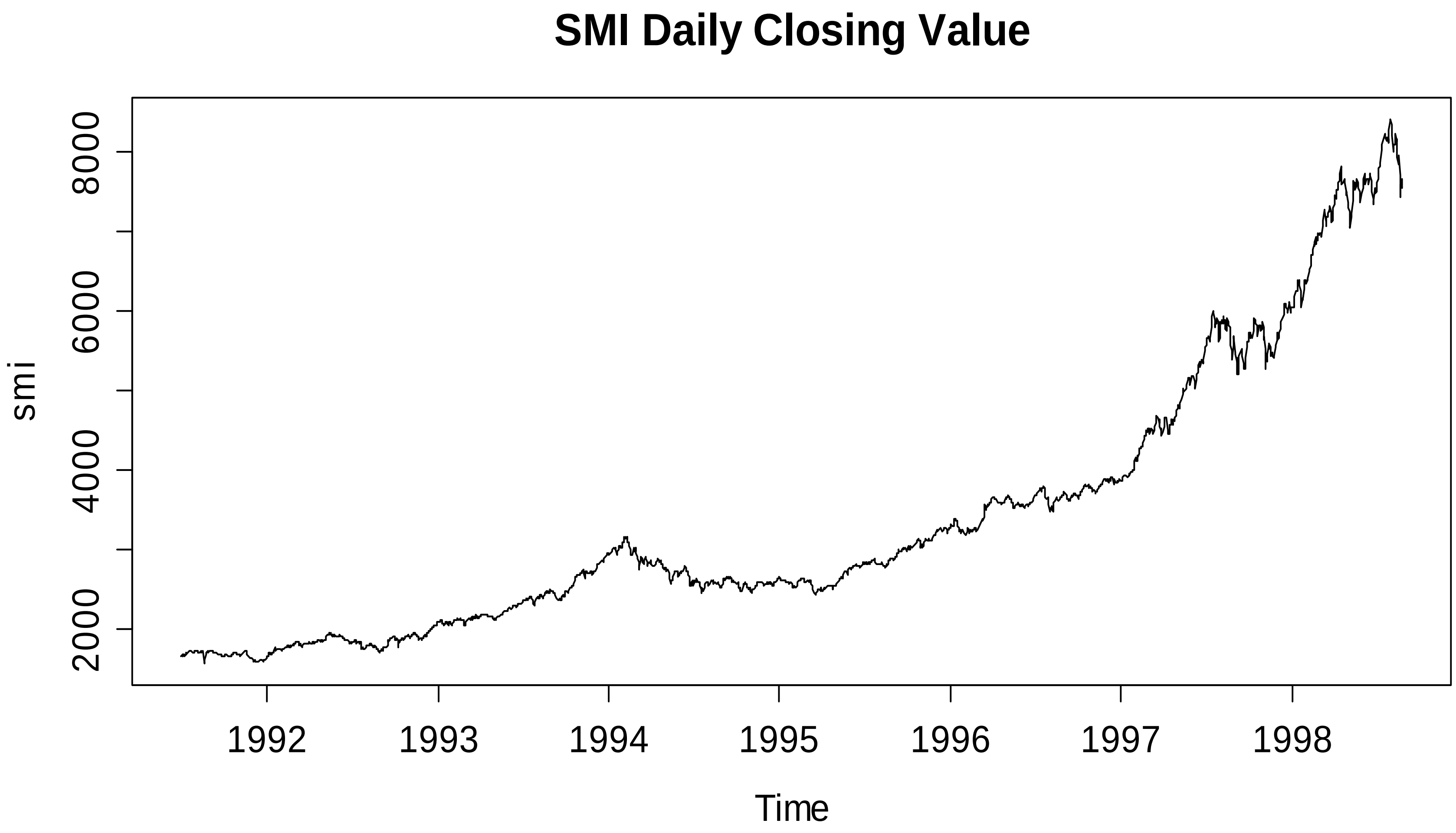
$$Y_t = \log(X_t) - \log(X_{t-1}) = \log\left(\frac{X_t}{X_{t-1}}\right) = \log\left(\frac{X_t - X_{t-1}}{X_{t-1}} + 1\right) \approx \frac{X_t - X_{t-1}}{X_{t-1}}$$

- The backshift operator "go back 1 step" allows for convenient notation with all differencing operations:

Backshift operator: $B(X_t) = X_{t-1}$

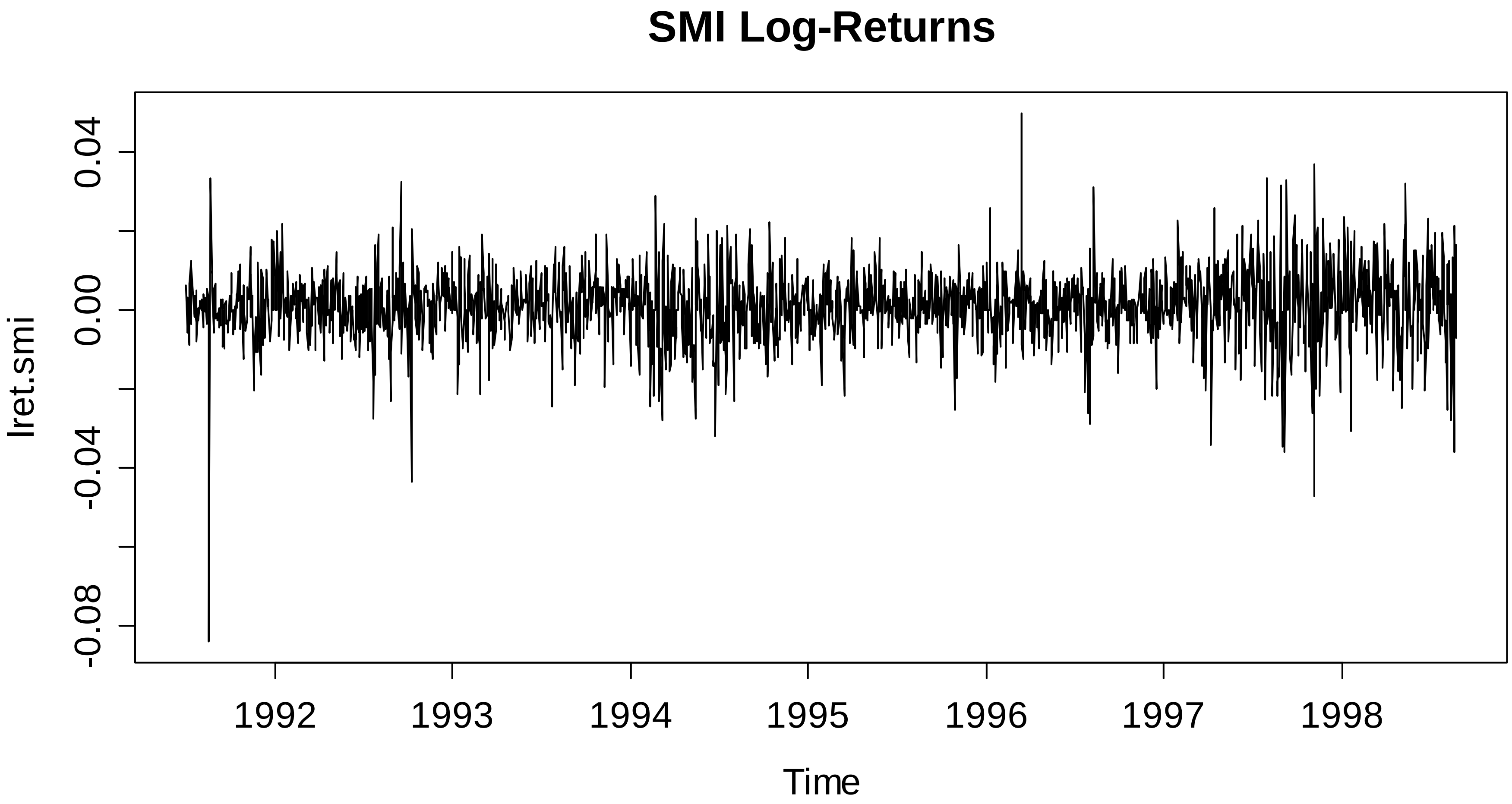
Differencing: $Y_t = (1-B)X_t = X_t - X_{t-1}$

Differencing Series with Transformation



Applied Time Series Analysis
SS 2014 – Week 02

Differencing Series with Transformation



Higher-Order Differencing

The “normal” differencing from above managed to remove any linear trend from the data. In case of polynomial trend, that is no longer true. But we can take higher-order differences:

$$\begin{aligned} X_t &= \alpha + \beta_1 t + \beta_2 t^2 + R_t,\ R_t \ \textit{stationary} \\ Y_t &= (1-B)^2 X_t \\ &= (X_t - X_{t-1}) - (X_{t-1} - X_{t-2}) \\ &= R_t - 2R_{t-1} + R_{t-2} + 2\beta_2 \end{aligned}$$

A quadratic trend can be removed by taking second-order differences. However, what we obtain is not an estimate of the remainder term R_t, but something that is much more complicated.

Removing Seasonal Effects

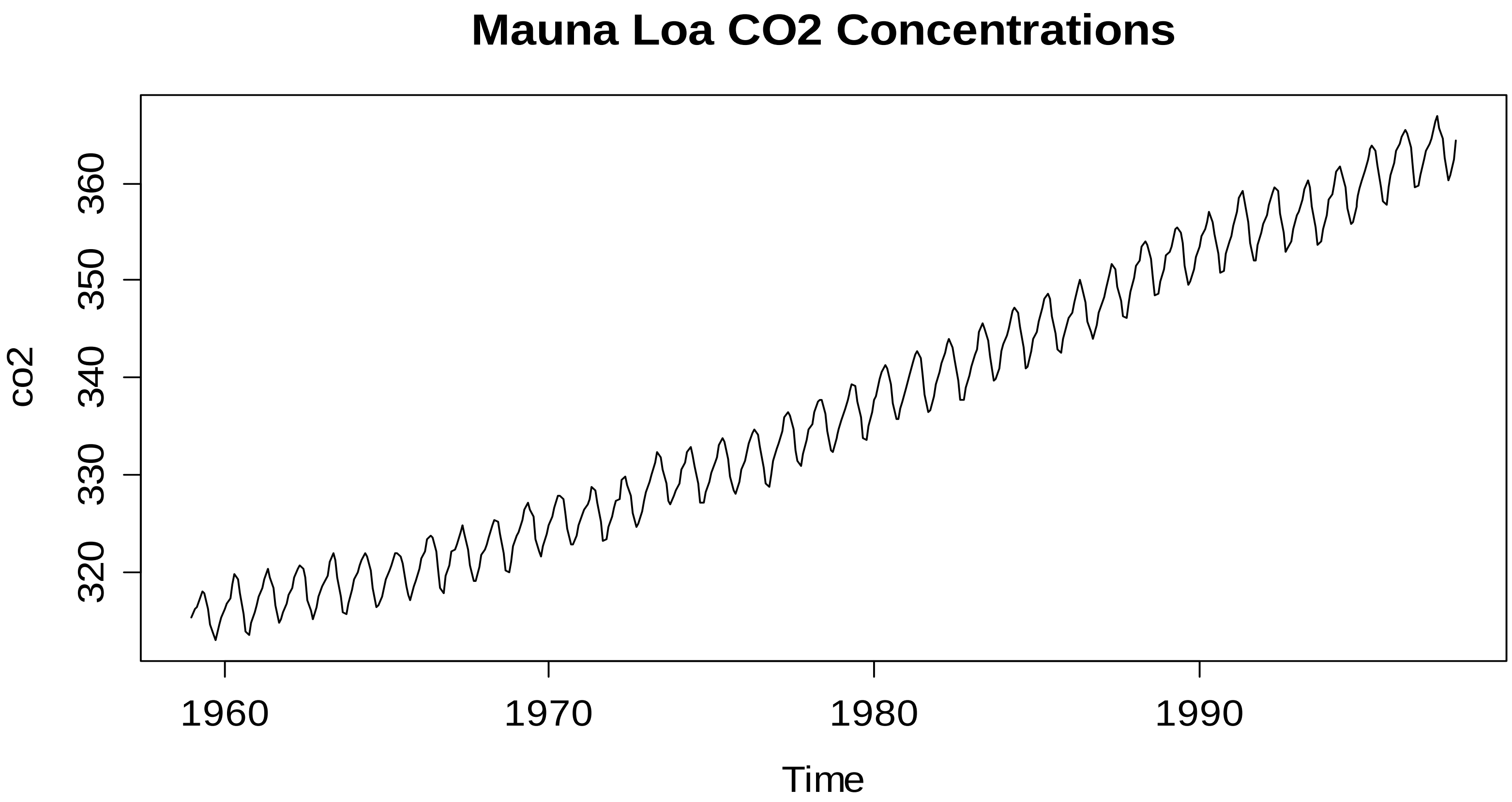
Time series with seasonal effects can be made stationary through differencing by comparing to the previous periods' value.

$$Y_t = (1 - B^p) X_t = X_t - X_{t-p}$$

- Here, p is the frequency of the series.
- A potential trend which is exactly linear will be removed by the above form of seasonal differencing.
- In practice, trends are rarely linear but slowly varying: $m_t \approx m_{t-1}$ However, here we compare m_t with m_{t-p}, which means that seasonal differencing often fails to remove trends completely.

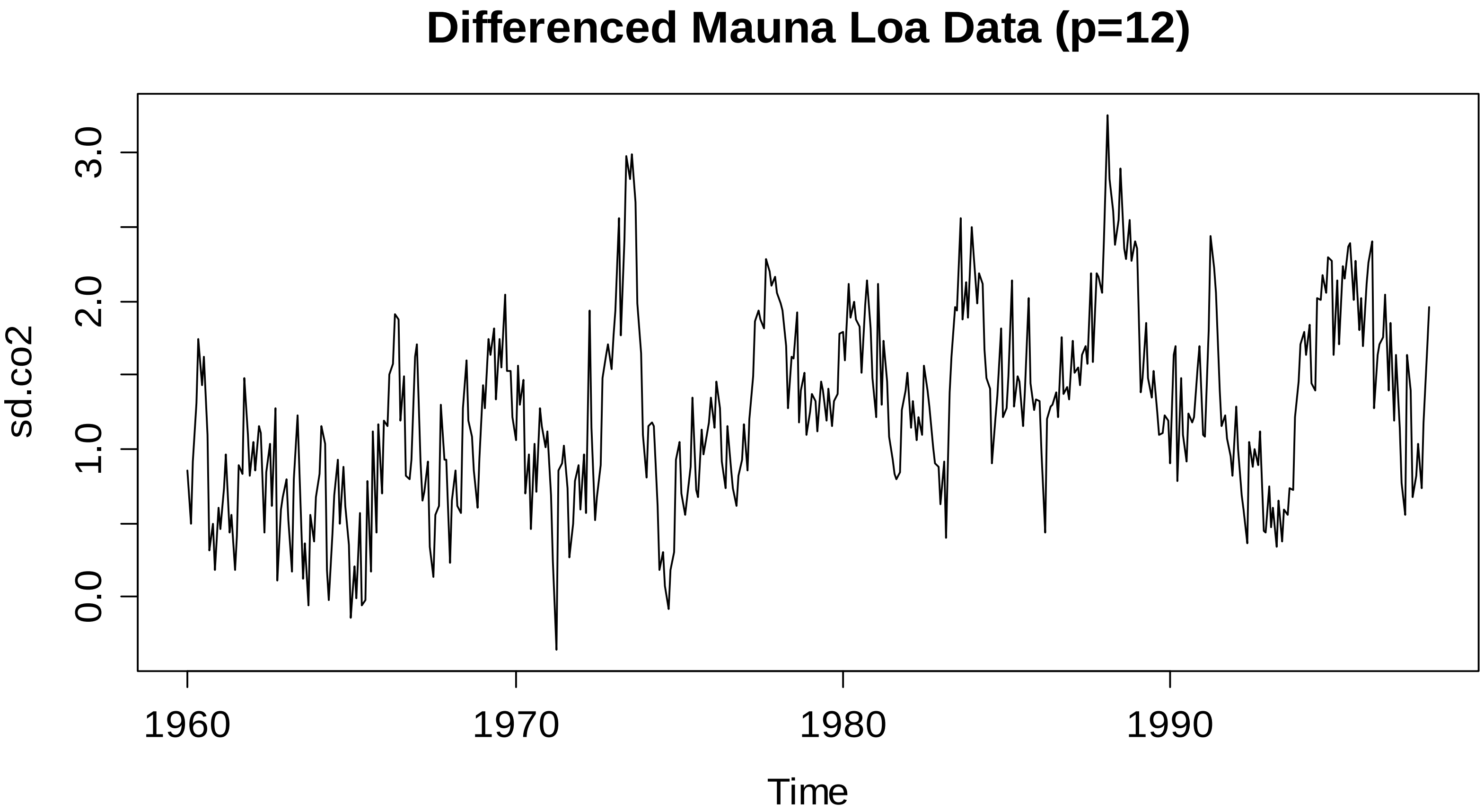
Seasonal Differencing: Example

```
> data(co2); plot(co2, main=…)
```



Seasonal Differencing: Example

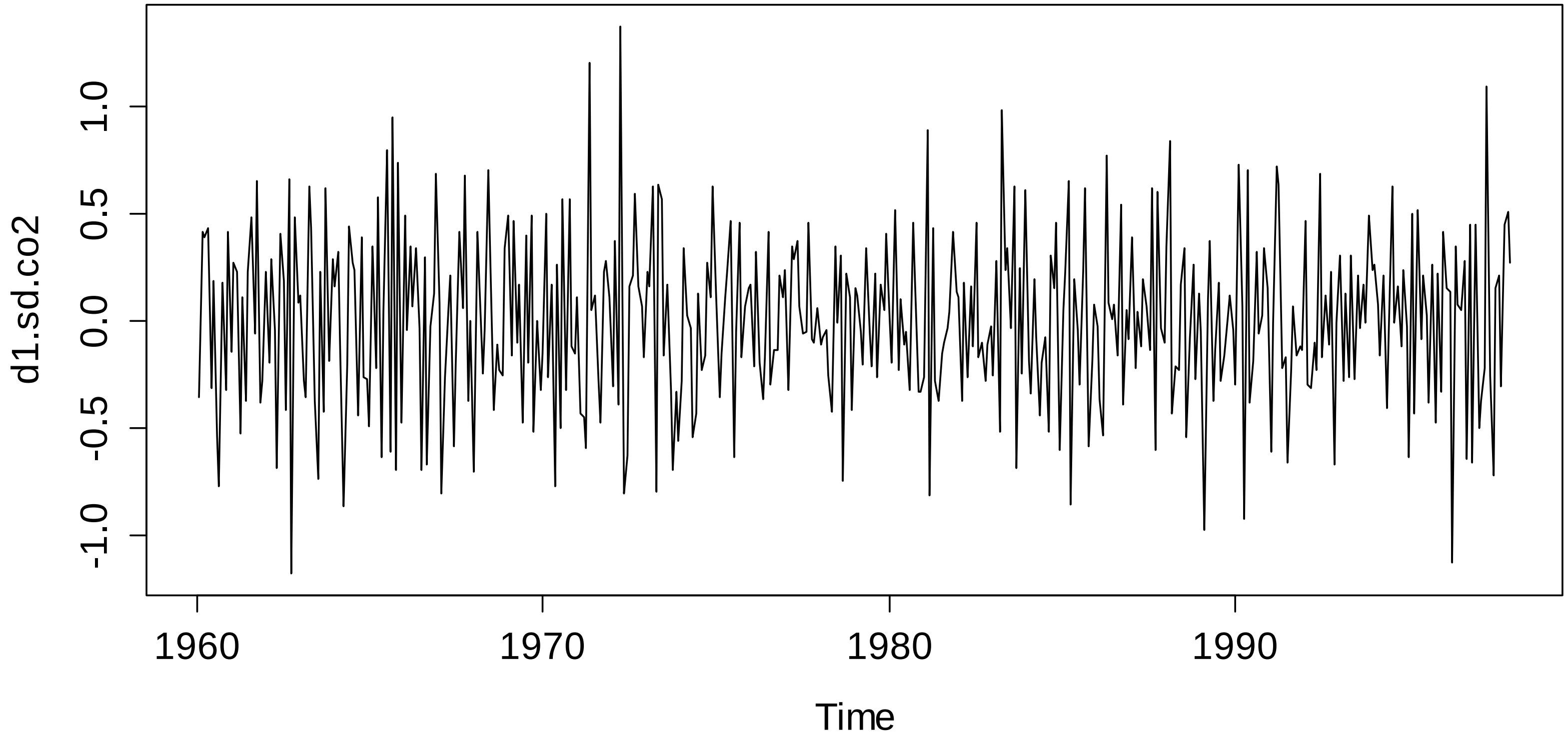
```
> sd.co2 <- diff(co2, lag=12)
```



Seasonal Differencing: Example

This is: $Z_t = (1-B)Y_t = (1-B)(1-B^{12})X_t$

Twice Differenced Mauna Loa Data (p=12, p=1)



Differencing: Remarks

Some advantages and disadvantages:

+ trend and seasonal effect can be removed

+ procedure is very quick and very simple to implement

- $\hat{m}_t$, $\hat{s}_t$ and $\hat{R}_t$ are not known, and cannot be visualised

- resulting time series will be shorter than the original

- differencing leads to strong artificial dependencies

- extrapolation of $\hat{m}_t$, $\hat{s}_t$ is not possible

Smoothing, Filtering: Part 1

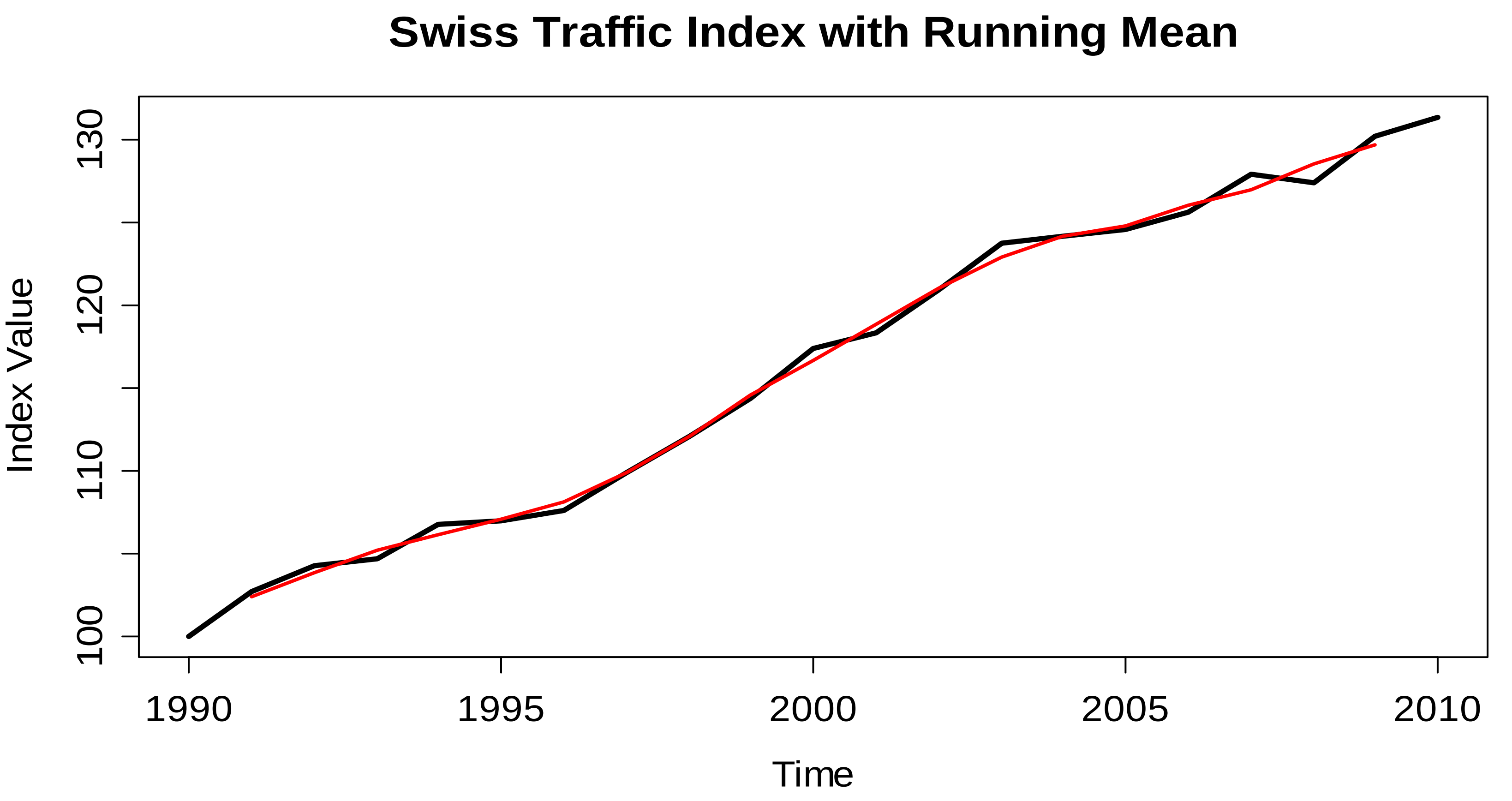
In the absence of a seasonal effect, the trend of a non-stationary time series can be determined by applying any **additive, linear filter**. We obtain a new time series $\hat{m}_t$, representing the trend:

$$\hat{m}_t = \sum_{i=-p}^{q} a_i X_{t+i}$$

- the window, defined by p and q, can or can‘t be symmetric
- the weights, given by a_i, can or can‘t be uniformly distributed
- other smoothing procedures can be applied, too.

Trend Estimation with the Running Mean

```
> trd <- filter(SwissTraffic, filter=c(1,1,1)/3)
```



Smoothing, Filtering: Part 2

In the presence a seasonal effect, smoothing approaches are still valid for estimating the trend. We have to make sure that the sum is taken over an entire season, i.e. for monthly data:

$$\hat{m}_t = \frac{1}{12}\left(\frac{1}{2}X_{t-6} + X_{t-5} + \ldots + X_{t+5} + \frac{1}{2}X_{t+6}\right) \ for \ t = 7,...,n-6$$

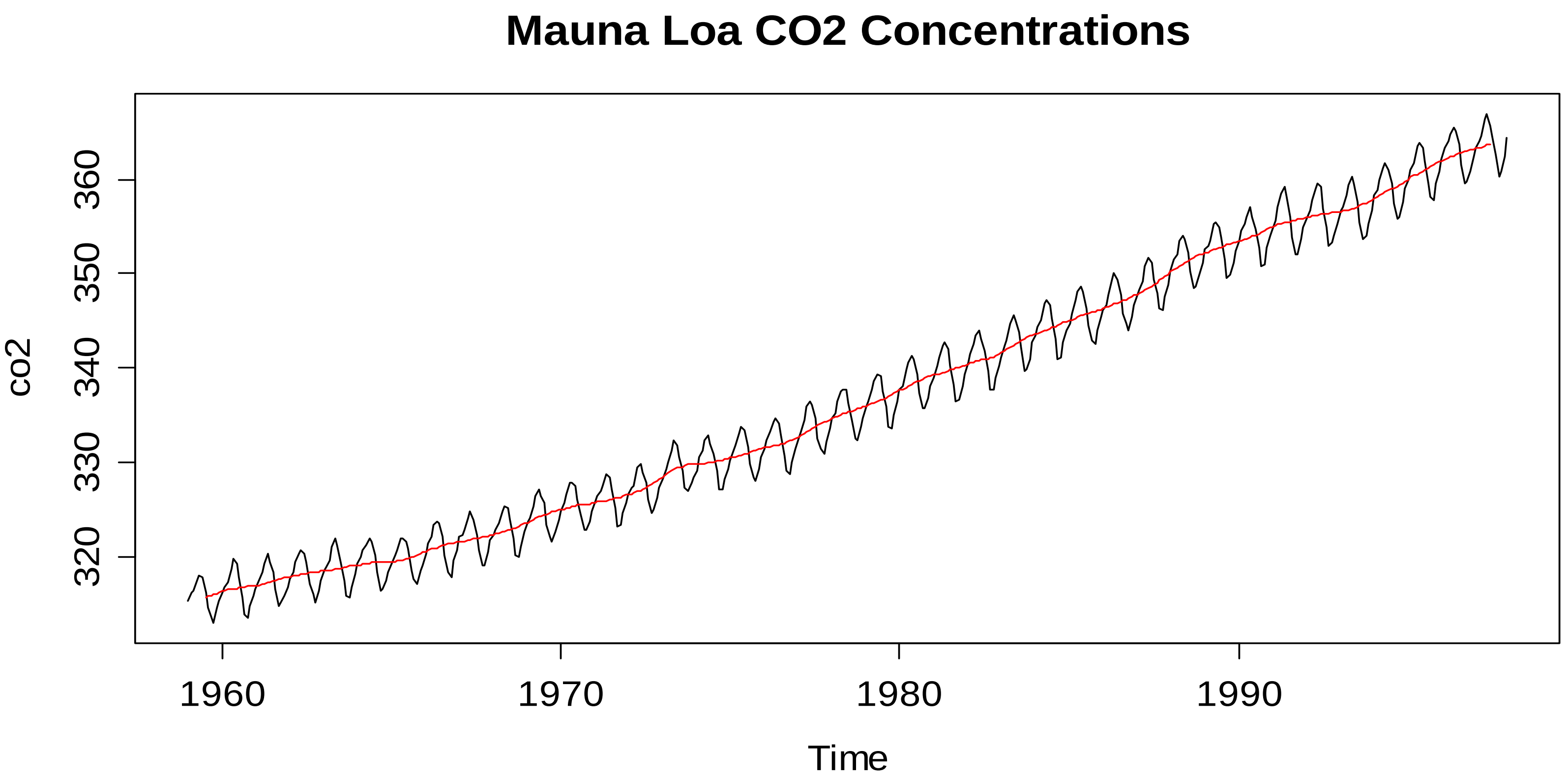
An estimate of the seasonal effect s_t at time t can be obtained by:

$$\hat{s}_t = x_t - \hat{m}_t$$

By averaging these estimates of the effects for each month, we obtain a single estimate of the effect for each month.

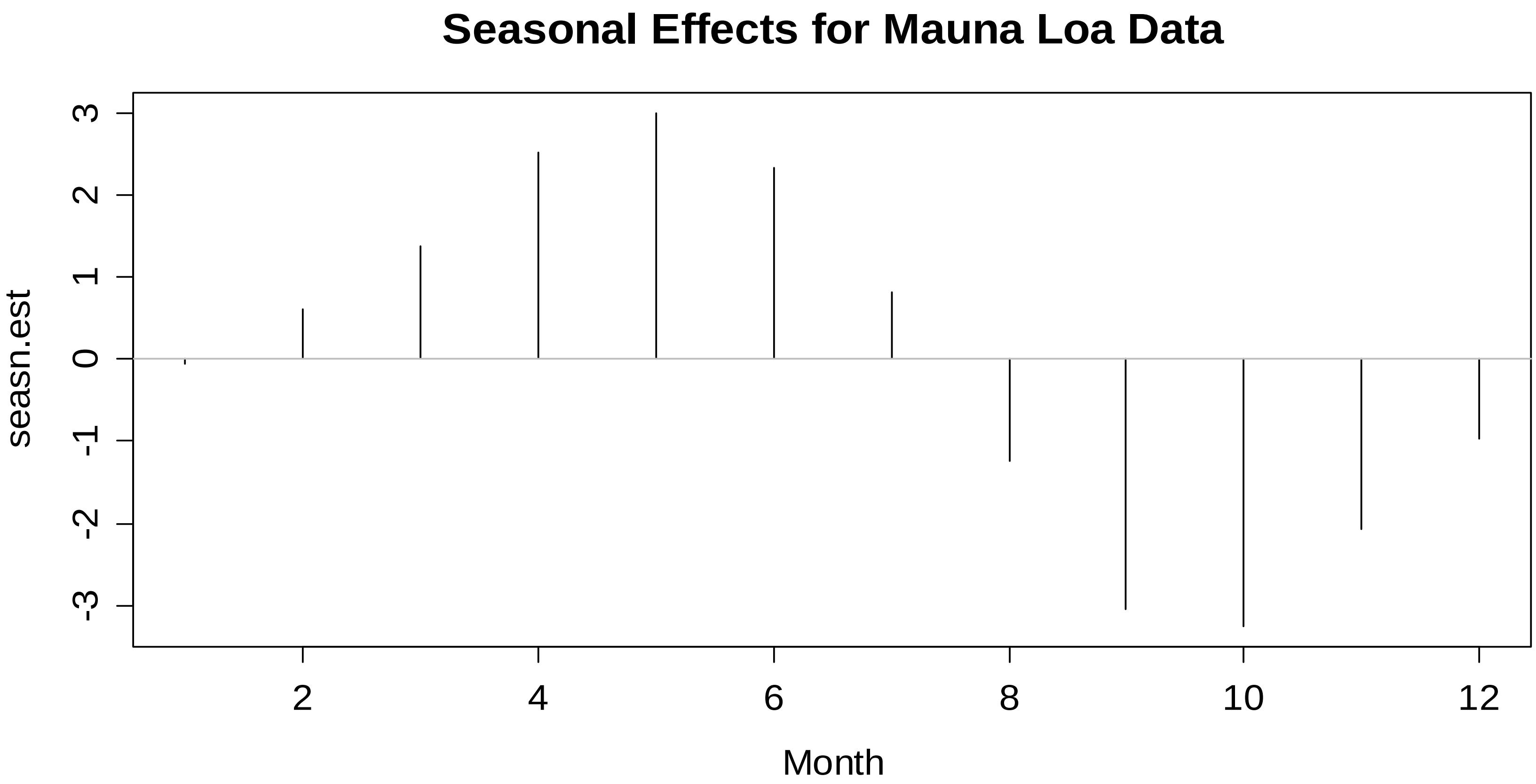
Trend Estimation for Mauna Loa Data

```
> wghts  <- c(.5,rep(1,11),.5)/12
> trd    <- filter(co2, filter=wghts, sides=2)
```



Estimating the Seasonal Effects

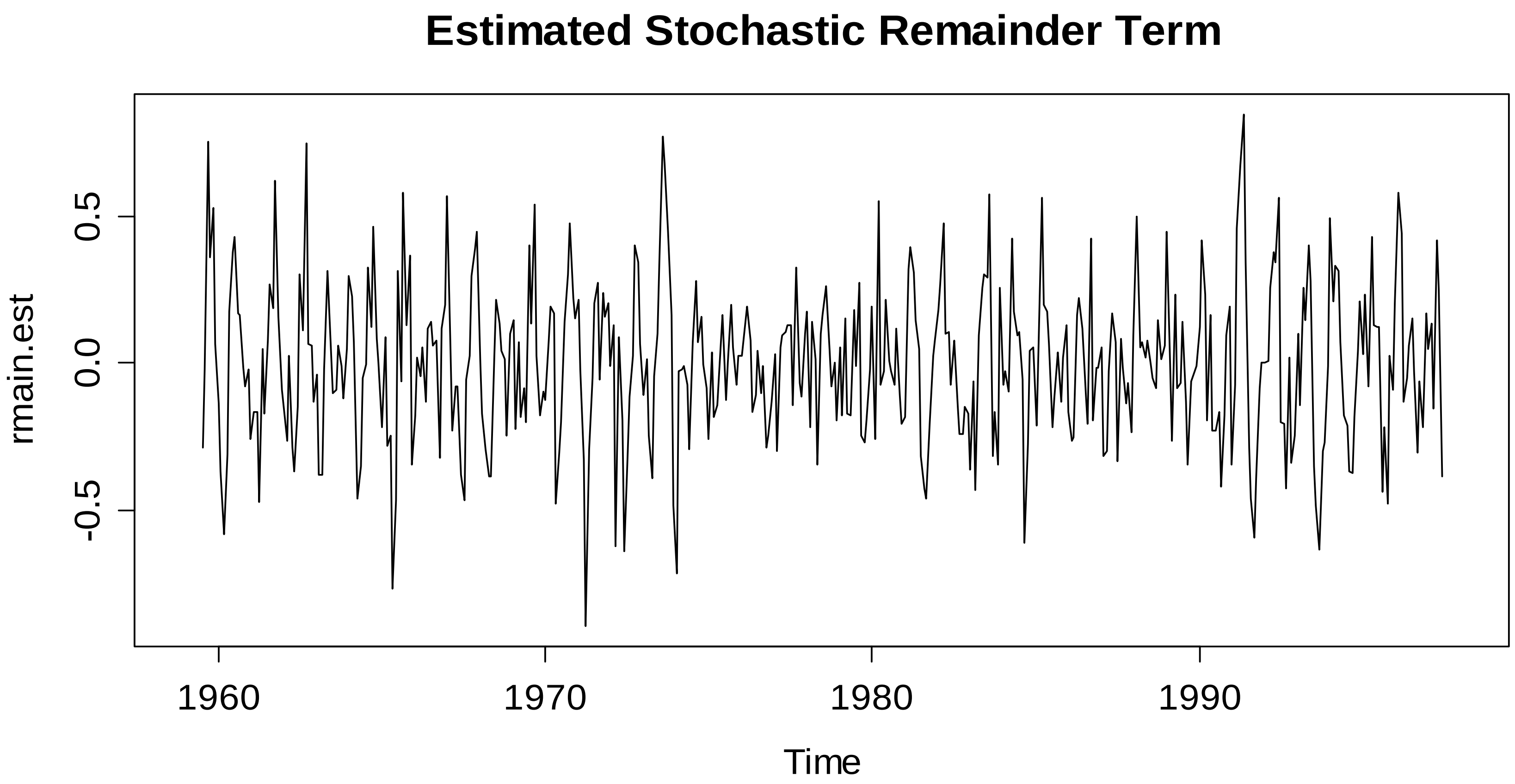
$$\hat{s}_{Jan} = \hat{s}_1 = \hat{s}_{13} = \ldots = \frac{1}{39} \cdot \sum_{j=0}^{38} (x_{12j+1} - \hat{m}_{12j+1})$$



Applied Time Series Analysis
SS 2014 – Week 02

Estimating the Remainder Term

$$\hat{R}_t = x_t - \hat{m}_t - \hat{s}_t$$



Applied Time Series Analysis
SS 2014 – Week 02

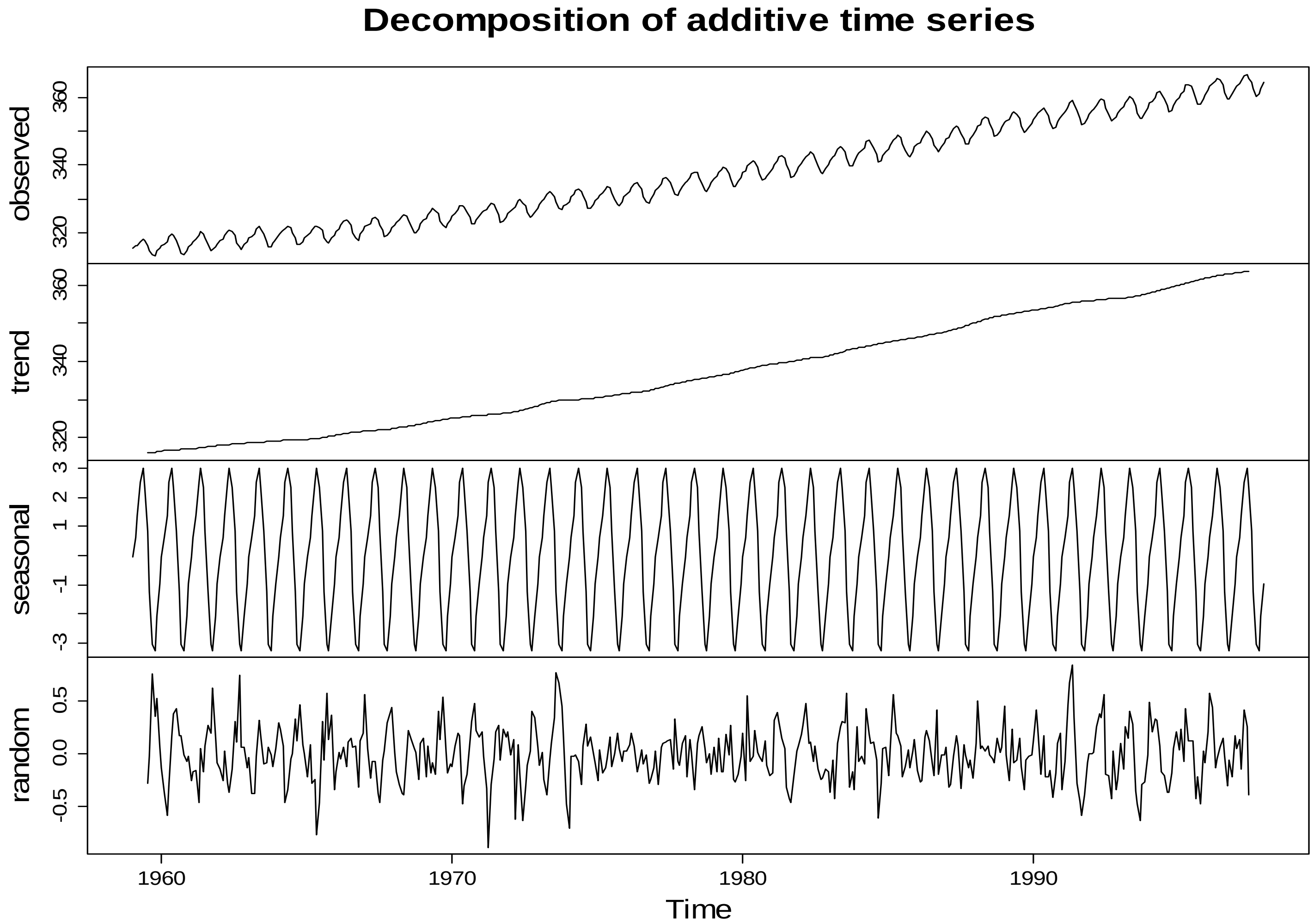
Smoothing, Filtering: Part 3

- The smoothing approach is based on estimating the trend first, and then the seasonality.
- The generalization to other periods than $p = 12$, i.e. monthly data is straighforward. Just choose a symmetric window and use uniformly distributed coefficients that sum up to 1.
- The sum over all seasonal effects will be close to zero. Usually, it is centered to be exactly there.
- This procedure is implemented in R with function: **`decompose()`**

Applied Time Series Analysis
SS 2014 – Week 02

Estimating the Remainder Term

```
> plot(decompose(co2))
```



Smoothing, Filtering: Remarks

Some advantages and disadvantages:

+ trend and seasonal effect can be estimated
+ $\hat{m}_t$, $\hat{s}_t$ and $\hat{R}_t$ are explicitly known, can be visualised
+ procedure is transparent, and simple to implement

- resulting time series will be shorter than the original
- the running mean is not the very best smoother
- extrapolation of $\hat{m}_t$, $\hat{s}_t$ are not entirely obvious

Smoothing, Filtering: STL-Decomposition

The Seasonal-Trend Decomposition Procedure by Loess

- is an iterative, non-parametric smoothing algorithm
- yields a simultaneous estimation of trend and seasonal effect

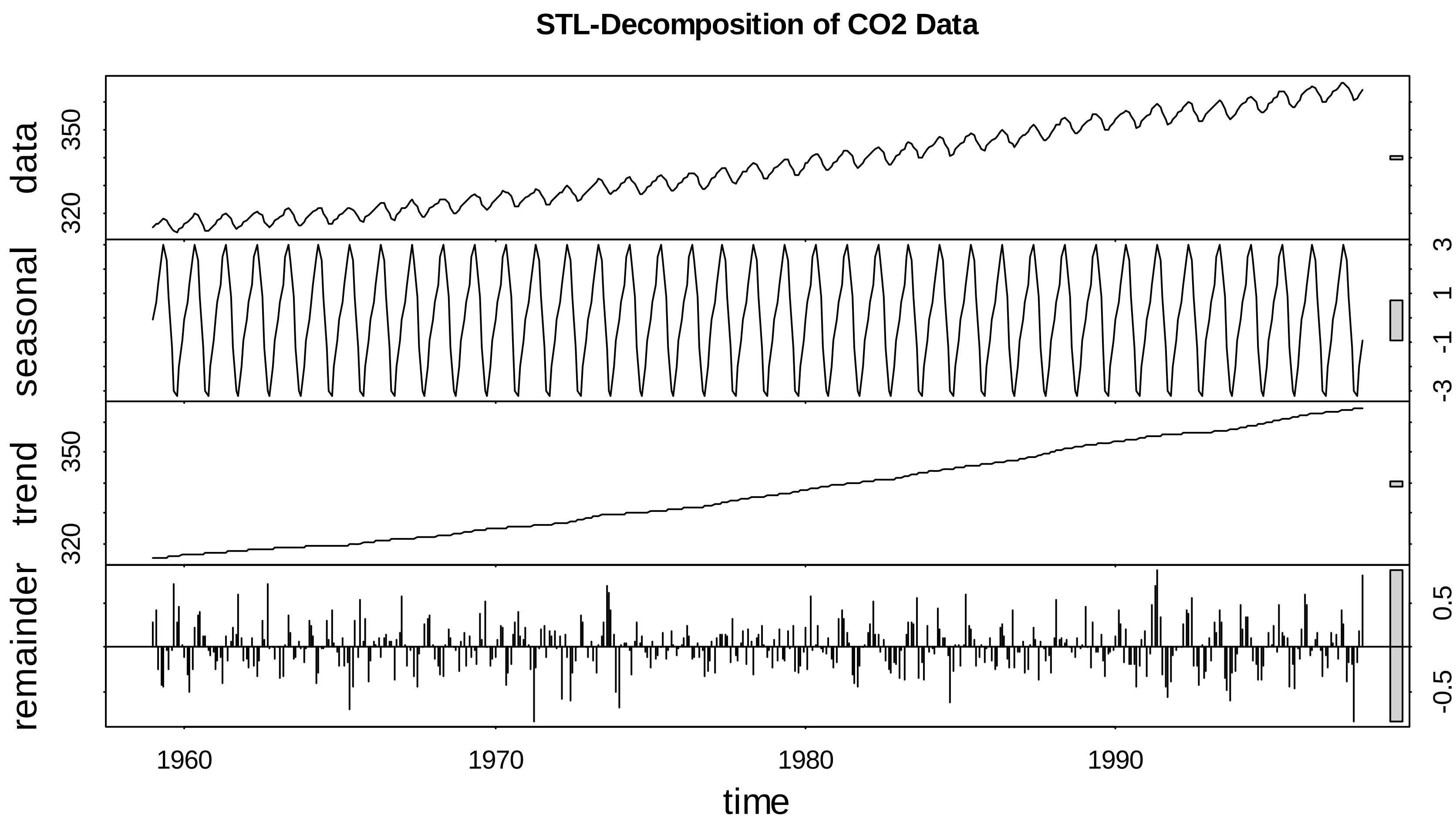
→ similar to what was presented above, but more **robust**!

+ very simple to apply
+ very illustrative and quick
+ seasonal effect can be constant or smoothly varying
- model free, extrapolation and forecasting is difficult

→ Good method for „having a quick look at the data“

STL-Decomposition for Periodic Series

```
> co2.stl <- stl(co2, s.window="periodic")
> plot(co2.stl, main="STL-Decomposition of CO2 Data")
```



Using the `stl()` *Function in R*

stl {stats} R Documentation

Seasonal Decomposition of Time Series by Loess

Description

Decompose a time series into seasonal, trend and irregular components using `loess`, acronym STL.

Usage

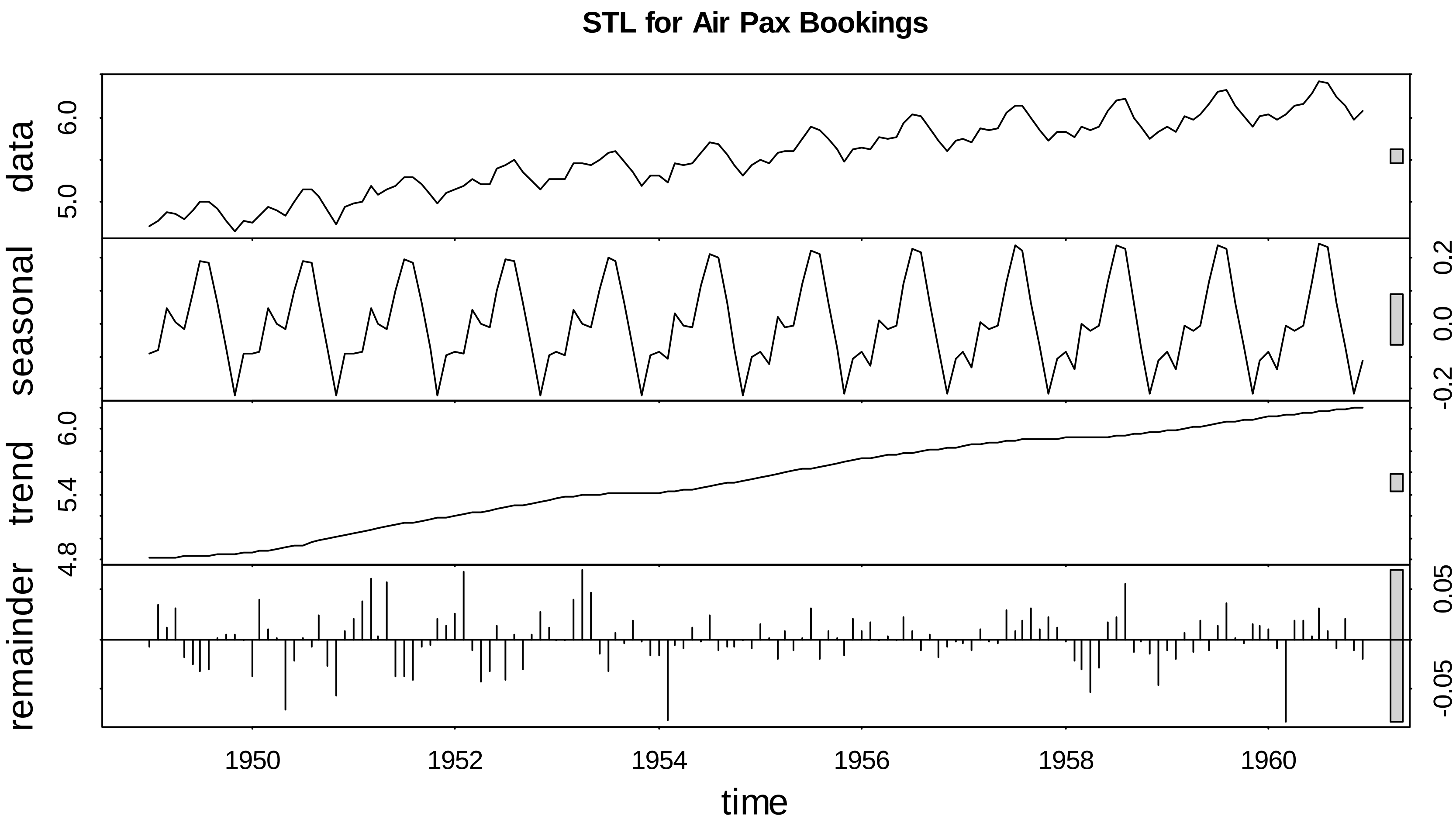
```
stl(x, s.window, s.degree = 0,
    t.window = NULL, t.degree = 1,
    l.window = nextodd(period), l.degree = t.degree,
    s.jump = ceiling(s.window/10),
    t.jump = ceiling(t.window/10),
    l.jump = ceiling(l.window/10),
    robust = FALSE,
    inner = if(robust)  1 else 2,
    outer = if(robust) 15 else 0,
    na.action = na.fail)
```

We need to supply argument `x` (i.e. the data) and `s.window` (for seasonal smoothing), either by setting it to `"periodic"` or to a numerical value. We can adjust `t.window` to a numerical value for altering the trend smoothing. Leave the rest alone!

Applied Time Series Analysis
SS 2014 – Week 02

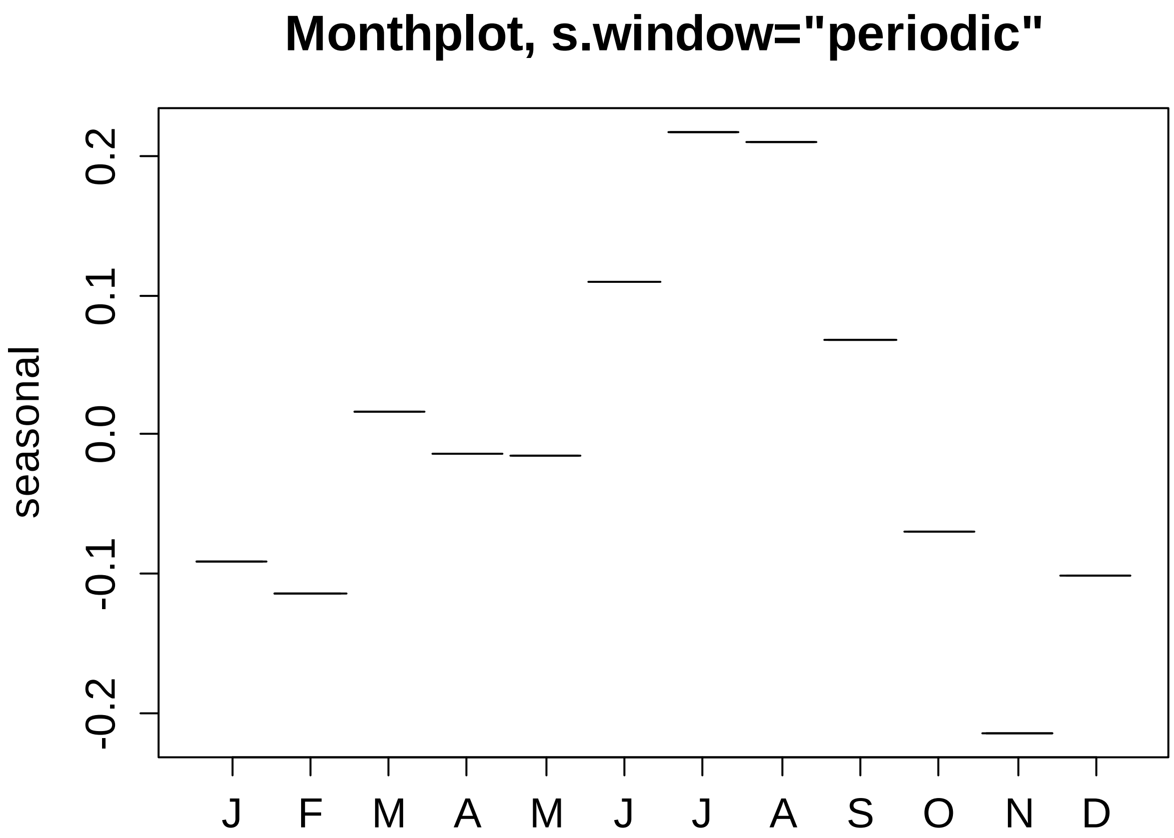
STL for Series with Evolving Seasonality

```
> lap.stl <- stl(lap, s.window=13)
> plot(lap.stl, main="STL for Air Pax Bookings")
```



STL for Series with Evolving Seasonality

```
> monthplot(stl(lap, s.window="periodic"))
```

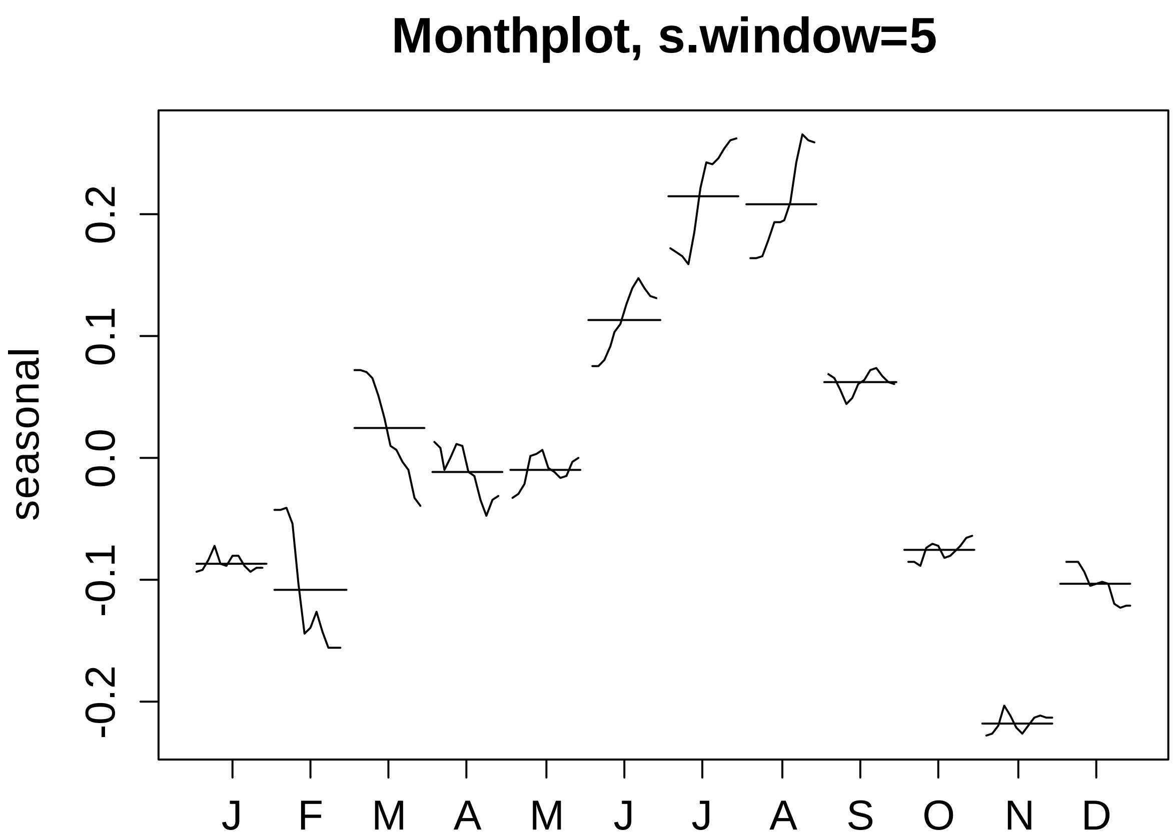


Constant Seasonality:

Check the STL plot on the previous slide for assessing whether this is reasonable or not!

STL for Series with Evolving Seasonality

```
> monthplot(stl(lap, s.window=5))
```

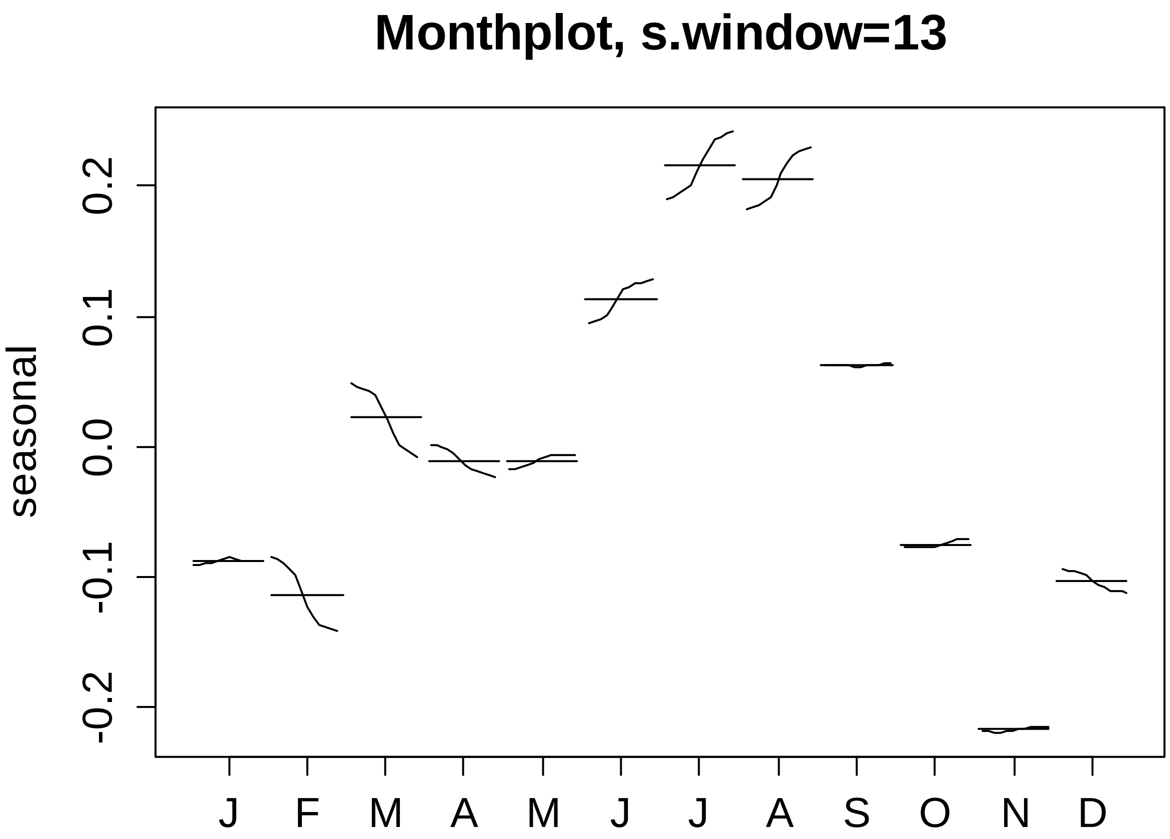


Evolving Seasonality:

Too little smoothing in the seasonal effect, the changes are irregular. As a remedy, increase parameter `s.window`

STL for Series with Evolving Seasonality

```
> monthplot(stl(lap, s.window=13))
```



Evolving Seasonality:

Adequate amount of smoothing will well chosen `s.window`

Smoothing, Filtering: Remarks

Some advantages and disadvantages:

\+ trend and seasonal effect can be estimated

\+ $\hat{m}_t$, $\hat{s}_t$ and $\hat{R}_t$ are explicitly known, can be visualised

\+ procedure is transparent, and simple to implement

\- resulting time series will be shorter than the original

\- the running mean is not the very best smoother

\- extrapolation of $\hat{m}_t$, $\hat{s}_t$ are not entirely obvious

Parametric Modelling

When to use?

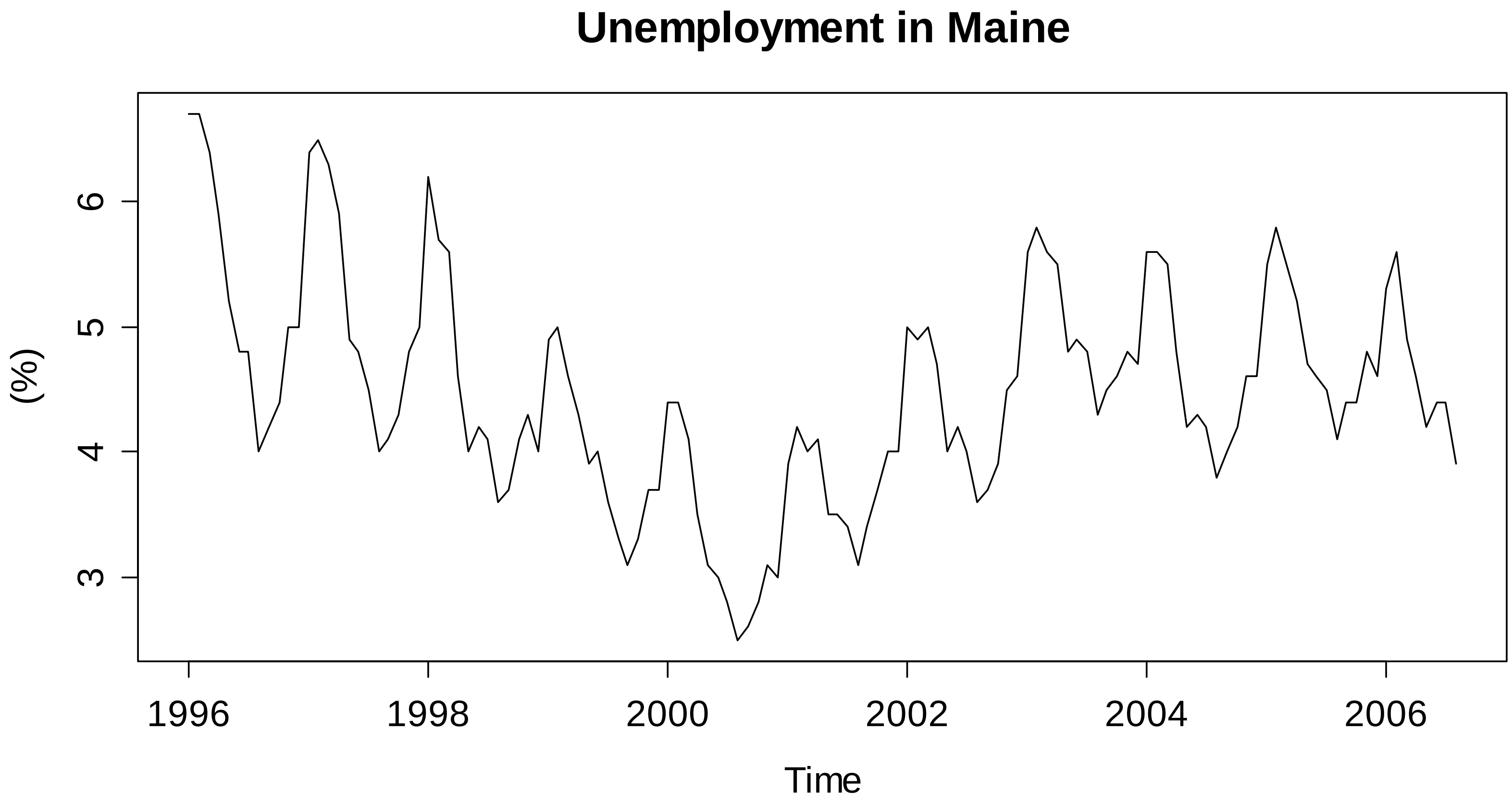
→ Parametric modelling is often used if we have previous knowledge about the trend following a functional form.

→ If the main goal of the analysis is forecasting, a trend in functional form may allow for easier extrapolation than a trend obtained via smoothing.

→ It can also be useful if we have a specific model in mind and want to infer it. Caution: correlated errors!

Parametric Modelling: Example

Maine unemployment data: Jan/1996 – Aug/2006



Modeling the Unemployment Data

Most often, time series are parametrically decomposed by using regression models. For the trend, polynomial functions are widely used, whereas the seasonal effect is modelled with dummy variables (= a factor).

$$X_t = \beta_0 + \beta_1 \cdot t + \beta_2 \cdot t^2 + \beta_3 \cdot t^3 + \beta_4 \cdot t^4 + \alpha_{i(t)} + E_t$$

where $t \in \{1, 2, ..., 128\}$

$i(t) \in \{1, 2, ..., 12\}$

Remark: choice of the polynomial degree is crucial!

Polynomial Order / OLS Fitting

Estimation of the coefficients will be done in a regression context. We can use the ordinary least squares algorithm, but:

- we have violated assumptions, E_t is not uncorrelated
- the estimated coefficients are still unbiased
- standard errors (tests, CIs) can be wrong

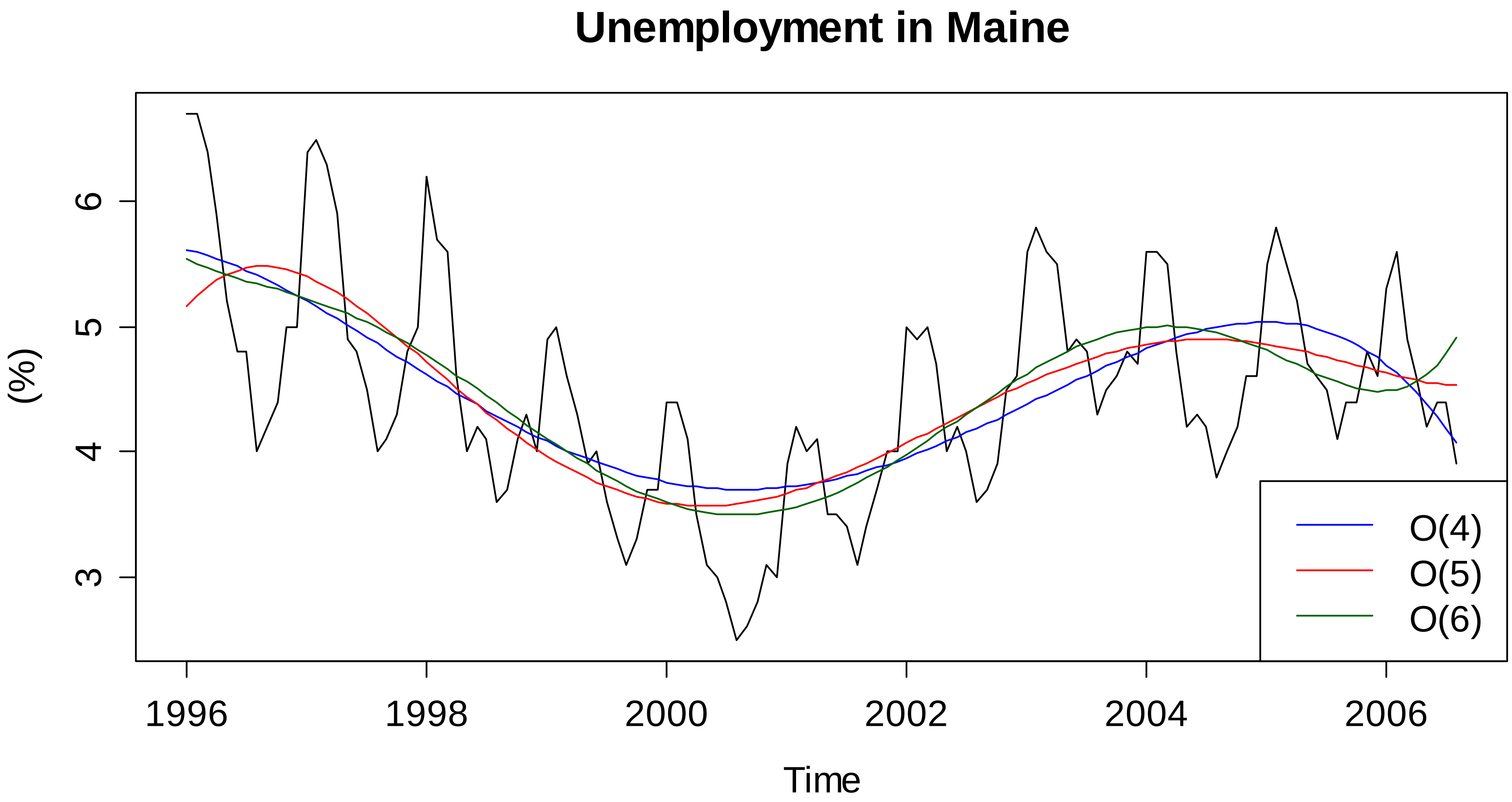
Which polynomial order is required?

Eyeballing allows to determine the minimum grade that is required for the polynomial. It is at least the number of maxima the hypothesized trend has, plus one.

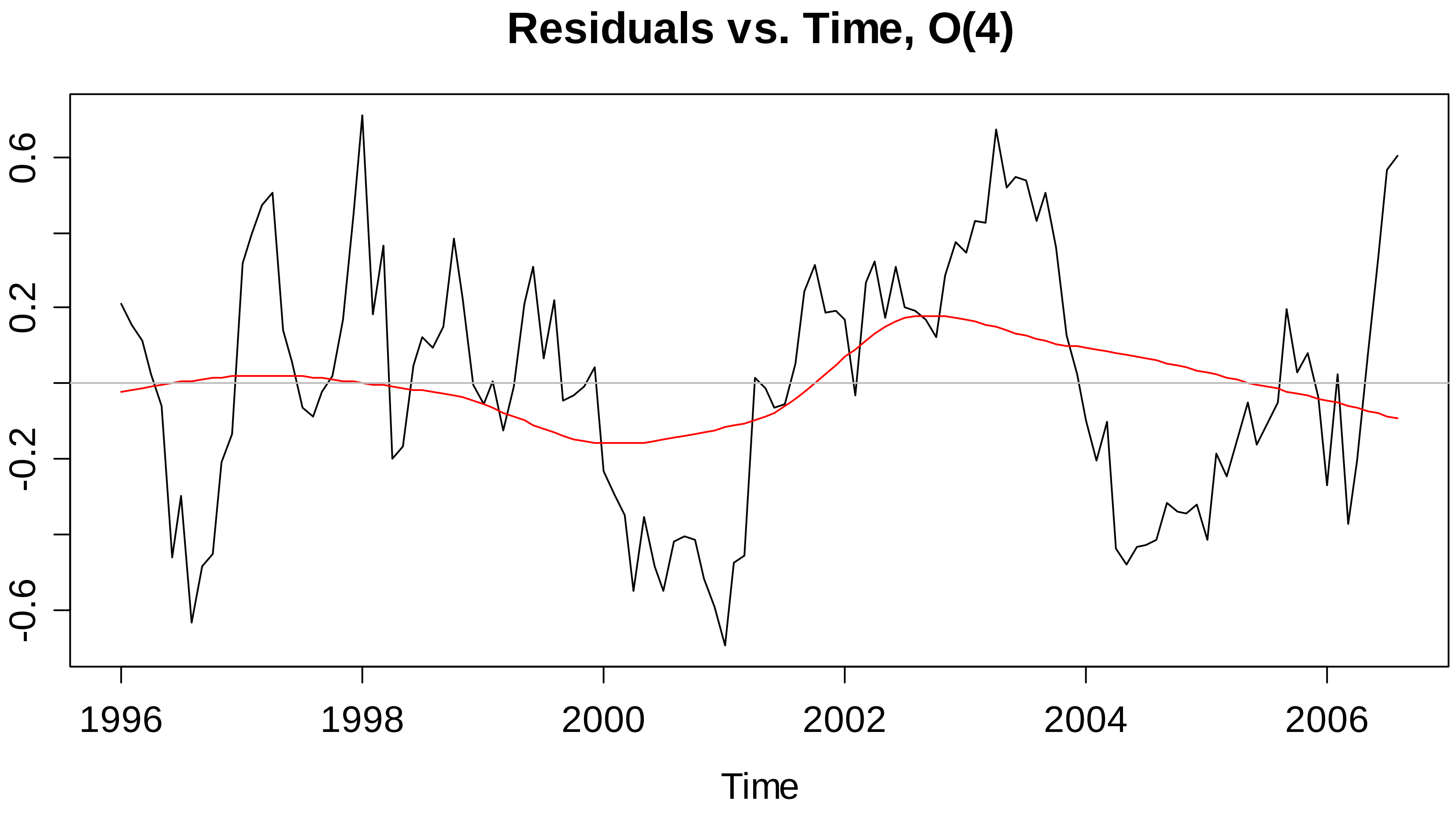
Important Hints for Fitting

- The main predictor used in polynomial parametric modeling is the time of the observations. It can be obtained by typing `time(maine)`.
- For avoiding numerical and collinearity problems, it is essential to center the time/predictors!
- R sets the first factor level to 0, seasonality is thus expressed as surplus to the January value.
- For visualization: when the trend must fit the data, we have to adjust, because the mean for the seasonal effect is usually different from zero!

Trend of O(4), O(5) and O(6)

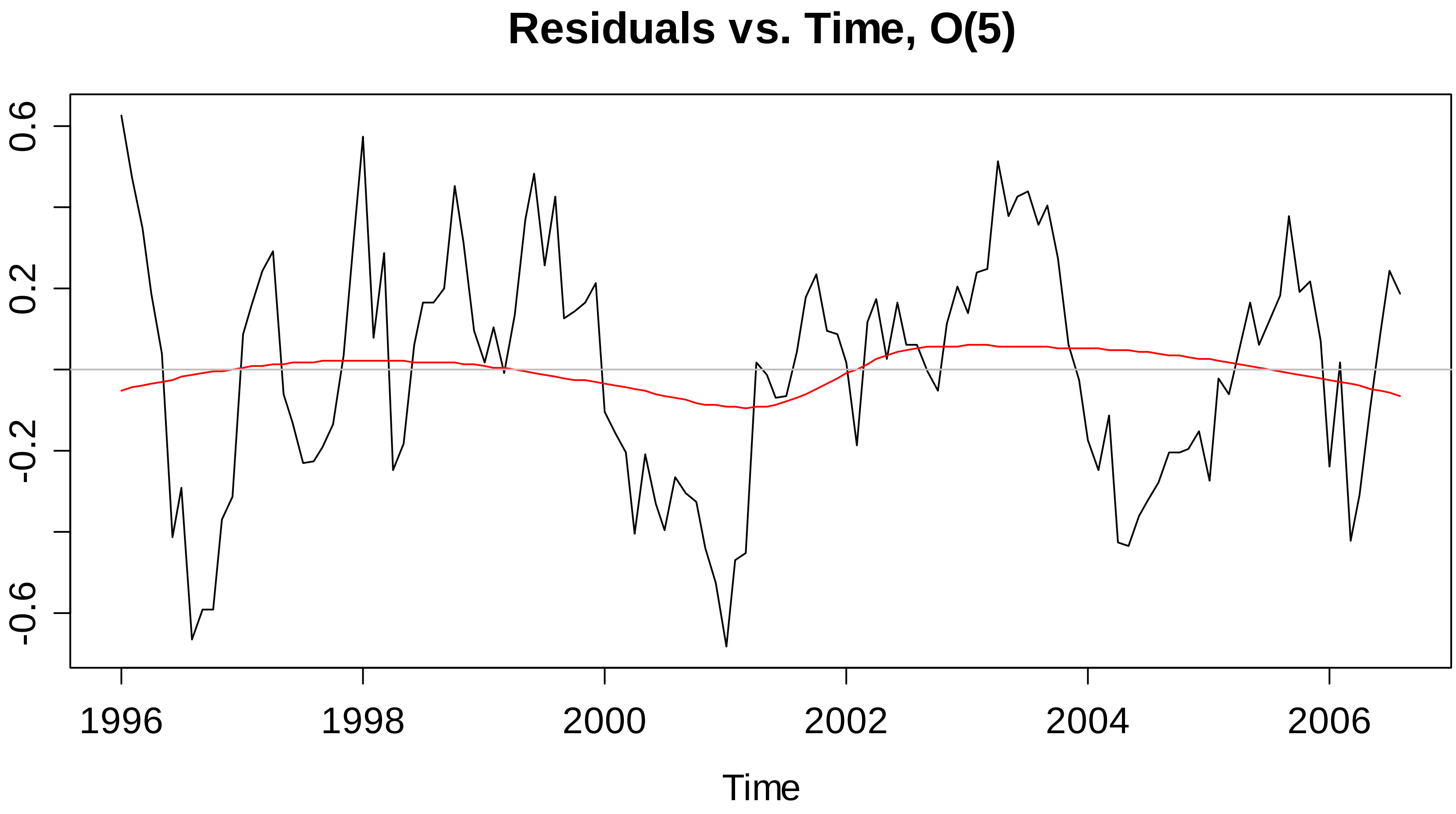


Residual Analysis: O(4)



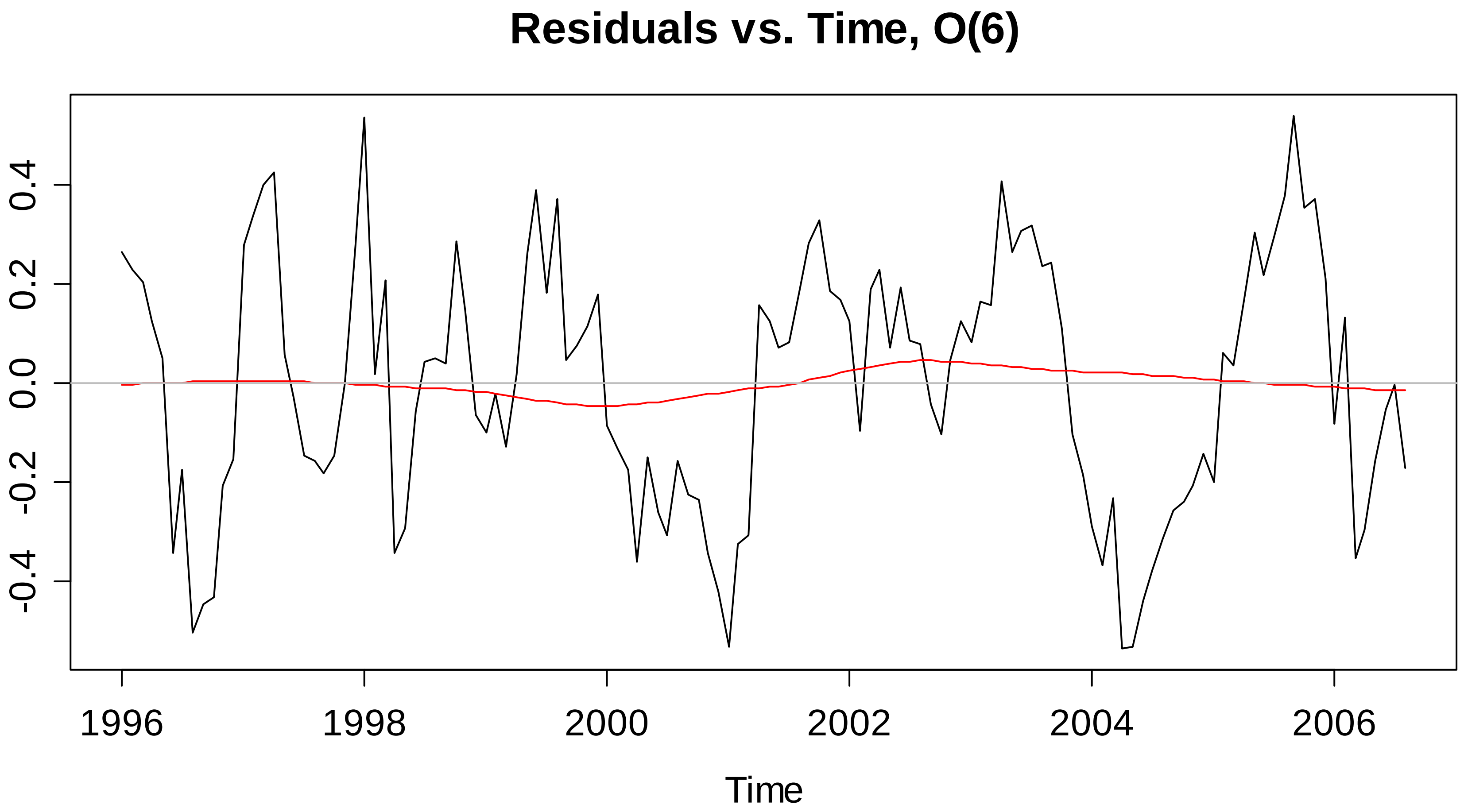
Applied Time Series Analysis
SS 2014 – Week 02

Residual Analysis: O(5)



Applied Time Series Analysis
SS 2014 – Week 02

Residual Analysis: O(6)



Parametric Modeling: Remarks

Some advantages and disadvantages:

+ trend and seasonal effect can be estimated
+ $\hat{m}_t$ and $\hat{s}_t$ are explicitly known, can be visualised
+ even some inference on trend/season is possible
+ time series keeps the original length

- choice of a/the correct model is necessary/difficult
- residuals are correlated: this is a model violation!
- extrapolation of $\hat{m}_t, \hat{s}_t$ are not entirely obvious