CS6301: R For Data Scientists

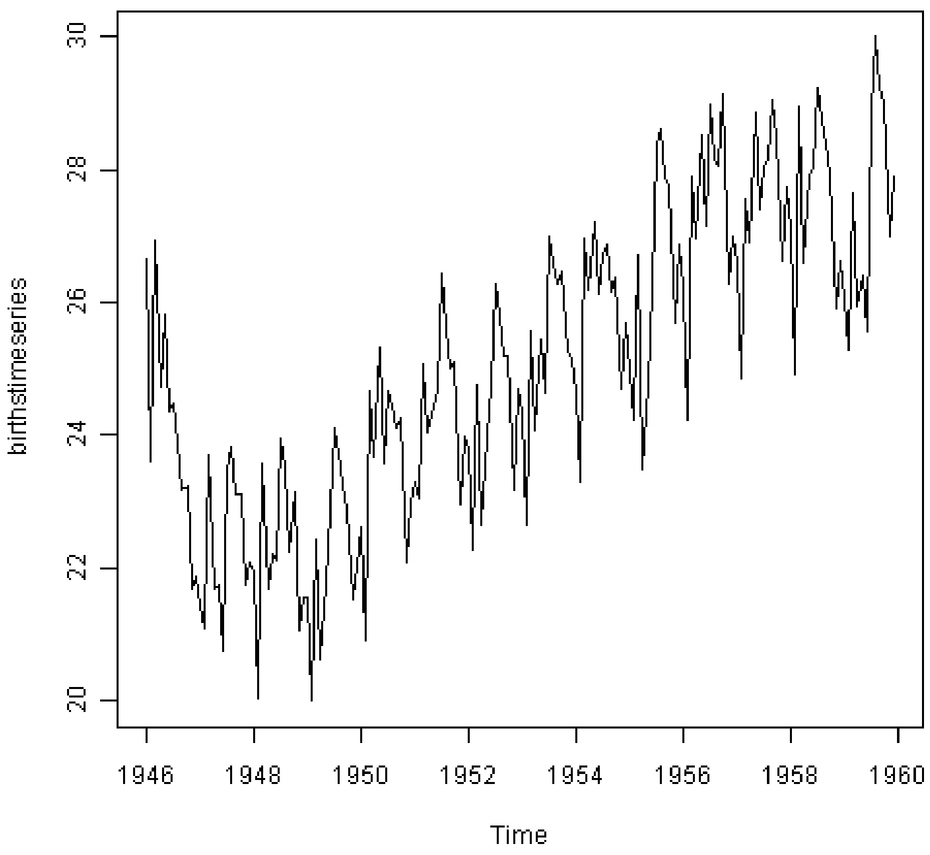
LECTURE 26: TIME SERIES I

Basics of Time Series Data

Time Series: Successive values in the dataset are measures of some quantity taken at equally spaced time intervals

Main goals:

- Analyze trends, use data to help understand the phenomenon we are measuring
- Identify "seasonal" trends
- Build models that can be used to forecast future values



Structure

The basic model consists of (possibly) three terms:

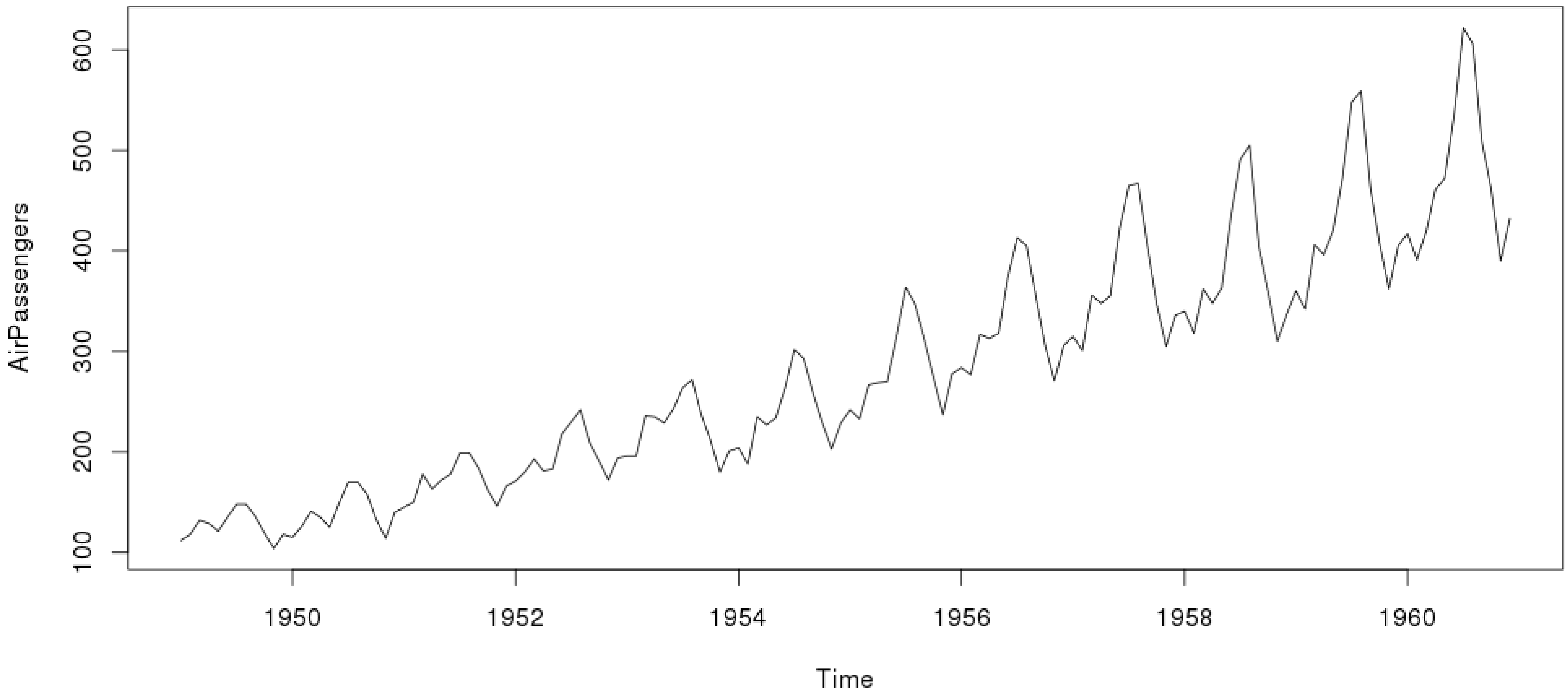
- Trend, which may be nonlinear
- Seasonality, depending on application
- Noise

Trend creates structured change in the data, but does not repeat

Seasonality also contributes structured change, but it does repeat

Noise is random and unpredictable

Example: AirPassengers



Multiplicative Seasonality

Notice the preceding chart shows a clear trend, but the amplitude of the season variation is changing

There is a correlation between the variance of the seasonality and the mean of the trend

The relative amplitude of the seasonality is related to the trend, changes over time

Autocorrelation Function (ACF)

Also known as **correlogram**

Measures correlation between lag points in time series

$$r_k = \frac{\sum_{t=k+1}^{T} (y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=1}^{T} (y_t - \bar{y})^2},$$

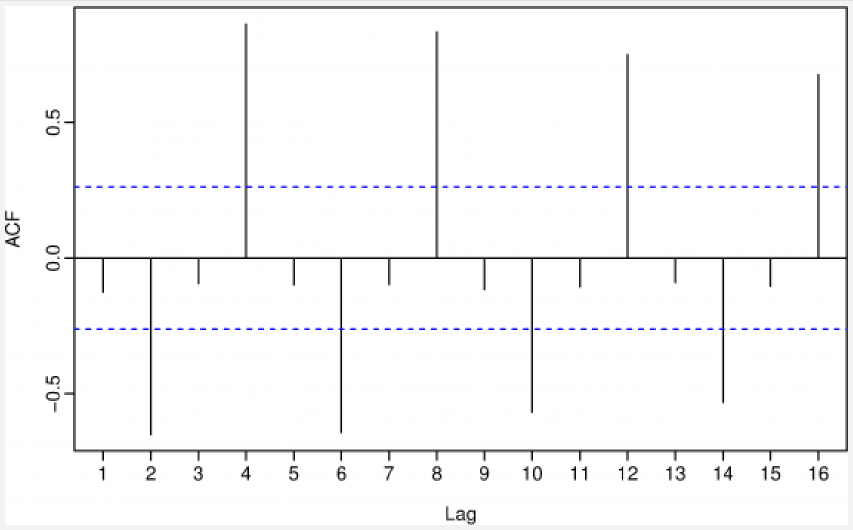


Figure 2.10: Autocorrelation function of quarterly beer production

Partial Autocorrelation Function (PACF)

If y_t and y_{t-1} are correlated, then so are y_{t-1} and y_{t-2}

But now y_t and y_{t-2} are correlated …

To overcome this problem, use partial ACF, which only measures correlation between y_t and y_{t-k}, after removing effects of other lags y_{t-1}, y_{t-2}, …, y_{t-k-1}

Stationary

A time series is **stationary** if its properties do not depend on the time it is observed

- No trend or seasonality
- Can be cyclic – cycles do not have fixed length

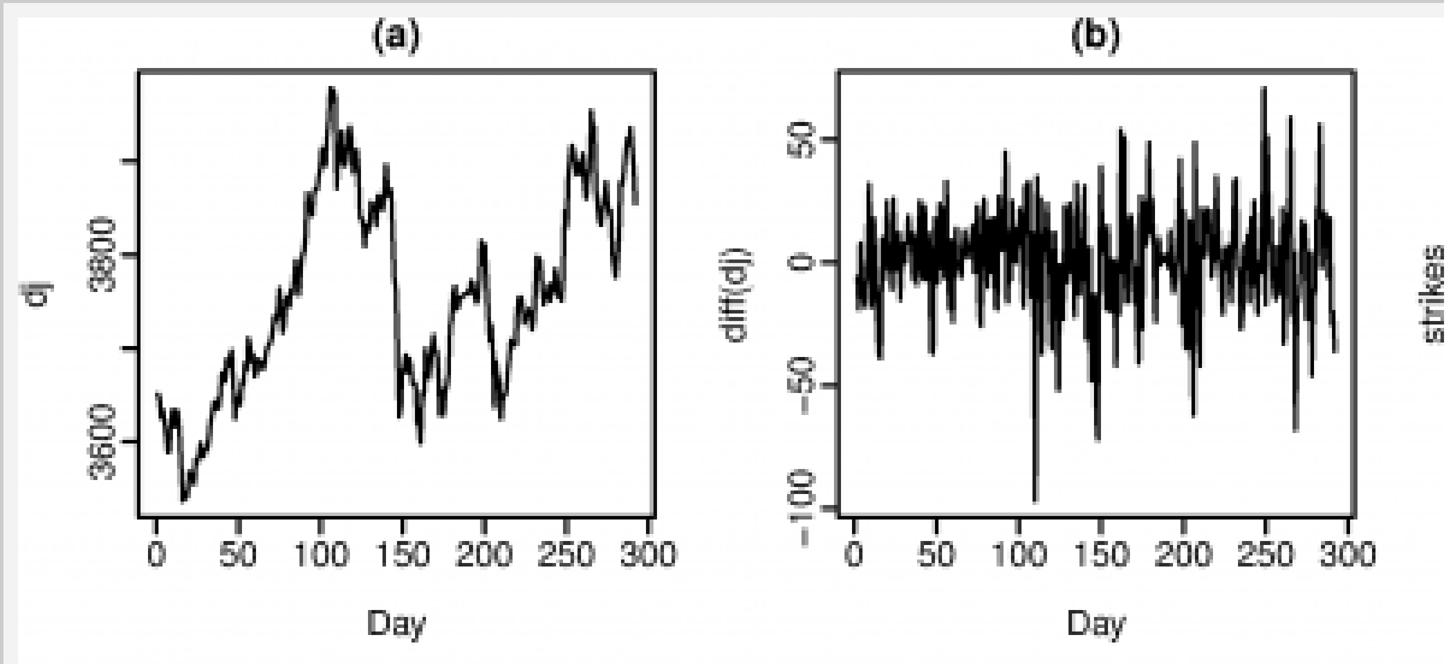
No predictable pattern in the long term

Plot will be basically horizontal, with constant variance

Differencing

Can transform a nonstationary time series to a stationary one by differencing

- This means taking the difference between successive observations – removes trend (if linear)



Trend Analysis

Possible to have no trend

Also possible to have no seasonality, just trend and noise …

How to get rid of noise?

Smoothing: Average data so noise cancels out

Moving Average: Replace the data point with a (possibly) weighted average of nearby points

- Can also use medians instead of mean (averages)

Seasonality Analysis

Identified as correlation between i^{th} and $(i - k)^{th}$ observations

- *k* is referred to as **lag**
- Look for pattern that repeats every *k* observations

Correlograms (plots of serial correlation coefficients) are often used

Often overlooked but usually a critical component (sales revenue, for example)

Time Series Analysis in R

Large number of packages support this, since this type of analysis comes up everywhere

For a list, check out the CRAN "task view": https://cran.r-project.org/web/views/TimeSeries.html

CRAN task views are a great way to see all CRAN packages associated with a particular task: https://cran.r-project.org/web/views/

Time Series Data in R

Given a dataset that represents time series data (i.e. a single list of numbers), we can convert it to time series in R by using the *ts()* function

- Can use "*start = x*" to specify an integer value for the first observation, used to track time
- Can use "*frequency = x*" to specify a frequency for the observations

plot() or plot.ts() will plot the time series

Decomposing Time Series

If there is no seasonality, simple averaging can be used to identify the trend

Use the “TTR” package in R

We can smooth this data using the “SMA” command

- Does a simple moving average, averages out noise

How to select n? Art …

Decomposing the Time Series

R has function that will analyze and decompose the time series data for you …

The function works by first finding a trend (by using moving averages) and removing it from the data

Then the seasonal component is found by averaging the remainder for each time period over all time intervals

- Works best is have an integer number of complete cycles

Remove seasonality to get noise component

Holt-Winters

A classic forecasting method that dates back to 1960

Uses "triple exponential smoothing"

Idea: At each data point, compute a point estimate of the expected value:

$$\hat{y}_x = \alpha \cdot y_x + (1 - \alpha) \cdot \hat{y}_{x-1}$$

Note this is recursively defined – the coefficient on the previous value is what creates the "exponential" effect

Triple Exponential Smoothing

One application of smoothing will get us the level – like β_0

Another application will get us the trend – like β_1

Triple application – accounts for seasonal component

$$
\begin{aligned}
\ell_x &= \alpha(y_x - s_{x-L}) + (1-\alpha)(\ell_{x-1} + b_{x-1}) && \text{level} \\
b_x &= \beta(\ell_x - \ell_{x-1}) + (1-\beta)b_{x-1} && \text{trend} \\
s_x &= \gamma(y_x - \ell_x) + (1-\gamma)s_{x-L} && \text{seasonal} \\
\hat{y}_{x+m} &= \ell_x + mb_x + s_{x-L+1+(m-1)modL} && \text{forecast}
\end{aligned}
$$

Model Assessment

We can look at the residuals (forecast errors), and look for a pattern (should appear random, normal)

Can use correlogram to check for correlations between forecast errors

Residual plots, normal probability plots

Can do histogram of forecast errors, check normal

Holt-Winters Issues

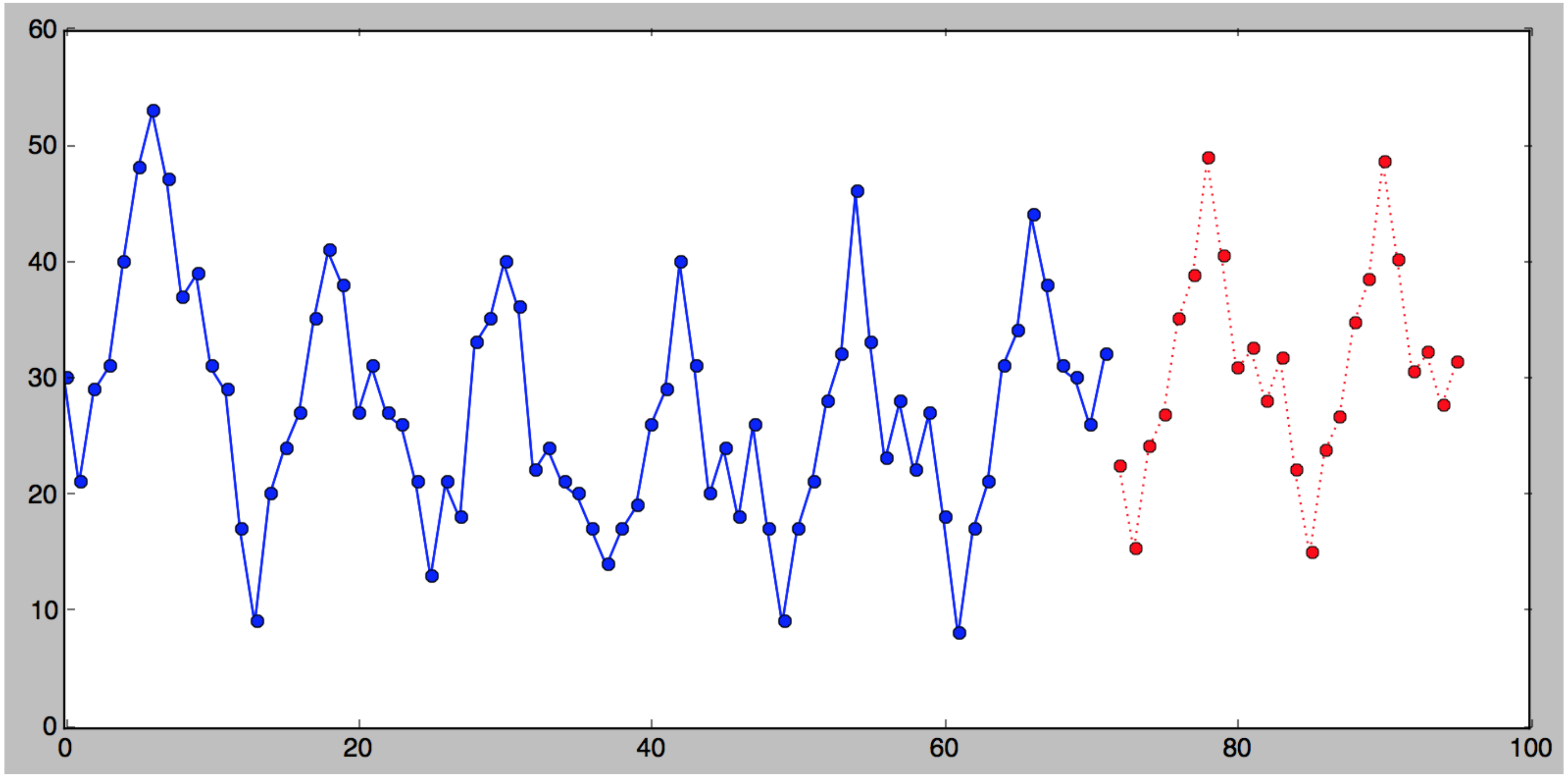
Need start up values

- May be tricky for seasonal and trend – need to select over several seasons
- Most software will do automatically, but the more data you have the better

Second problem – finding the smoothing parameters

- Trial and error?
- Can use SSE/MSE on training data to try various values

Holt-Winters



Other Approaches

Can actually do regression fits for this as well …

- Need to add dependent variable for time
- Will fit the trend line
- Can do more complicated fits to try to include seasonality (Fourier functions)
- Tricky …

Two other approaches are popular: Autoregression and Moving Average models

Links …

https://www.otexts.org/fpp/1/1

https://a-little-book-of-r-for-time-series.readthedocs.org/en/latest/

http://www.statoek.wiso.uni-goettingen.de/veranstaltungen/zeitreihen/sommer03/ts_r_intro.pdf

https://grisha.org/blog/2016/01/29/triple-exponential-smoothing-forecasting/

https://cran.r-project.org/web/packages/fpp/fpp.pdf