LEC002 Demand Forecasting

VG441 SS2020

Cong Shi
Industrial & Operations Engineering
University of Michigan

Properties

- ***Trend***: A long-term increase or decrease in the data. This can be seen as a slope (doesn't have to be linear) roughly going through the data.

- ***Seasonality***: A time series is said to be seasonal when it is affected by seasonal factors (hour of day, week, month, year, etc.). Seasonality can be observed with nice cyclical patterns of fixed frequency. ski / clothes

- ***Cyclicity***: A cycle occurs when the data exhibits 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". The duration of these fluctuations is usually at least 2 years. boom → recession

- ***Residuals***: Each time series can be decomposed in two parts:
  - A forecast, made up of one or several *forecasted* values
  - Residuals. They are the difference between an observation and its predicted value at each time step. Remember that

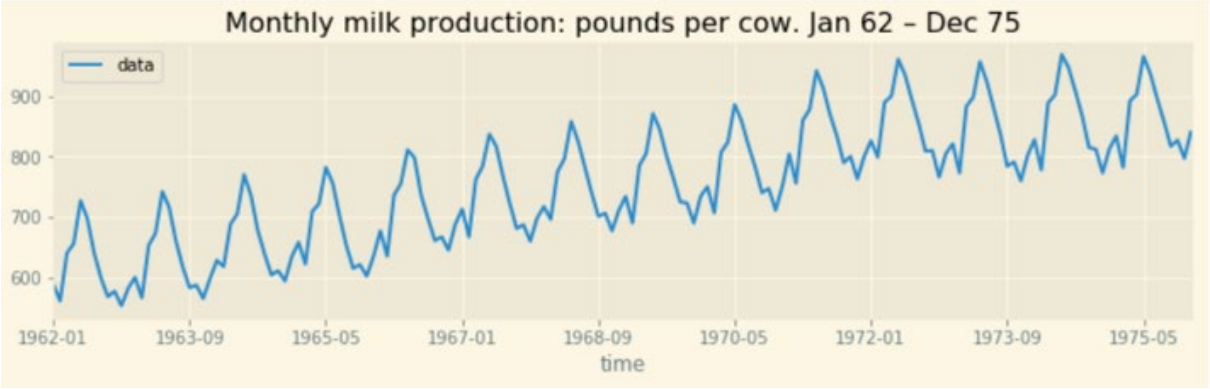
Value of series at time t = Predicted value at time t + Residual at time t

背

value = predict + residual

Examples

trend



Sesonal.

Examples

cyclic

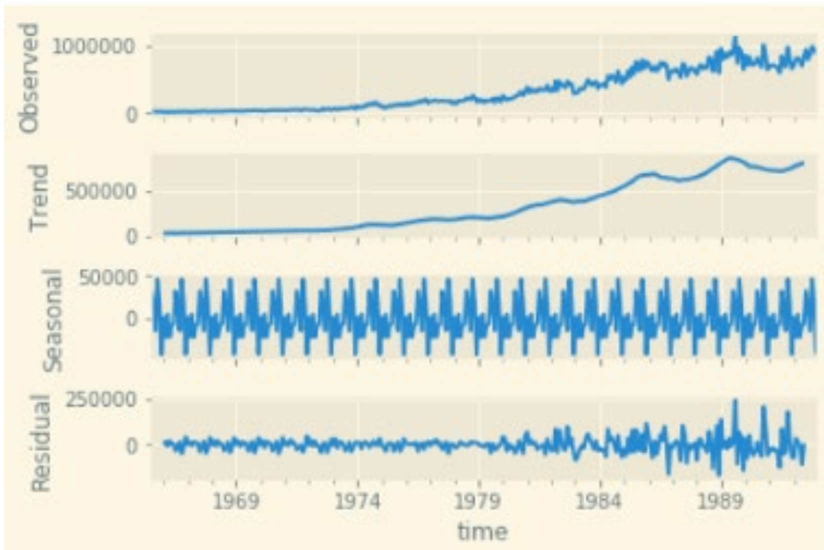
stationary

Decomposition of Time Series

Each time series can be thought as a mix between several parts :

- A trend (upward or downwards movement)
- A seasonal component
- Residuals

use python
可以自动



做
⇒ prediction

* stationary is easy to analyze

Simple Average

- Stationary model $D_t = I + \epsilon_t$

base signal

i.i.d shock

independent identically distributed

- Static forecast $\hat{y} = \frac{\sum_{t=1}^{N} D_t}{N}$

predictor

平均一下

- Derived based on minimizing MSE

residual/error

$$e_t = D_t - \hat{y}$$

求导 =0

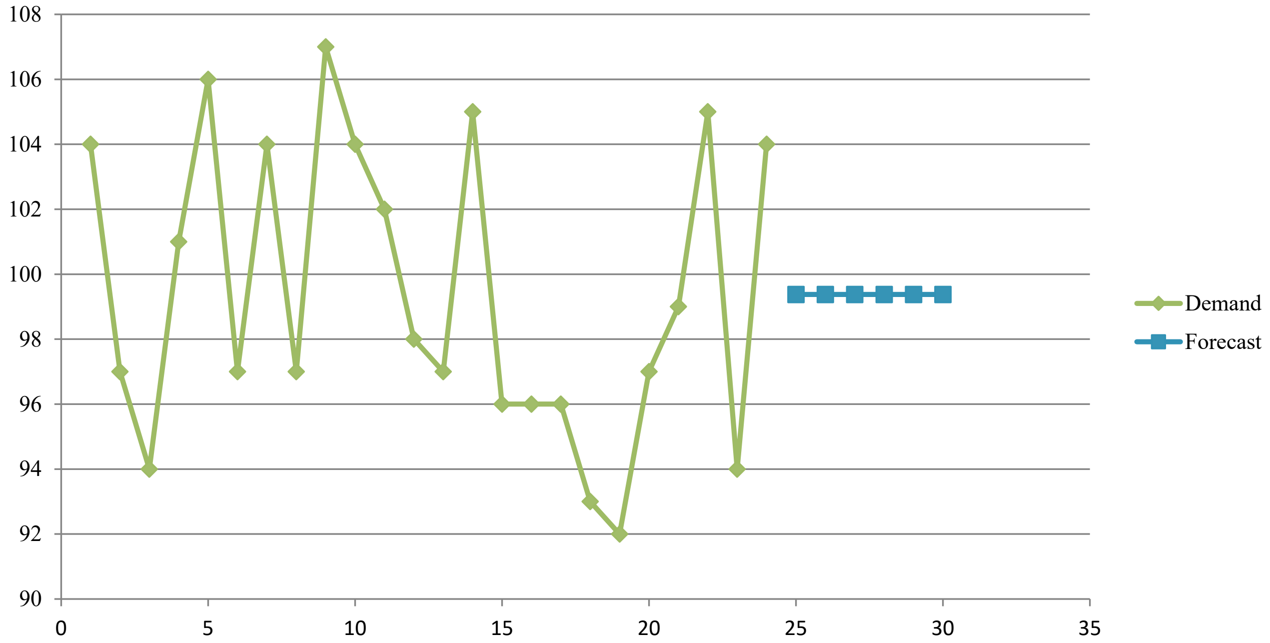
$$\frac{d\left(\sum_{t=1}^{N} e_t^2\right)}{d\hat{y}} = \frac{d\left[\sum_{t=1}^{N} (d_t - \hat{y})^2\right]}{d\hat{y}} = -2\sum_{t=1}^{N}(d_t - \hat{y}) = 0$$

min

mean square error

$\sum e_t^2$

Simple Average Model



Moving Average (MA)

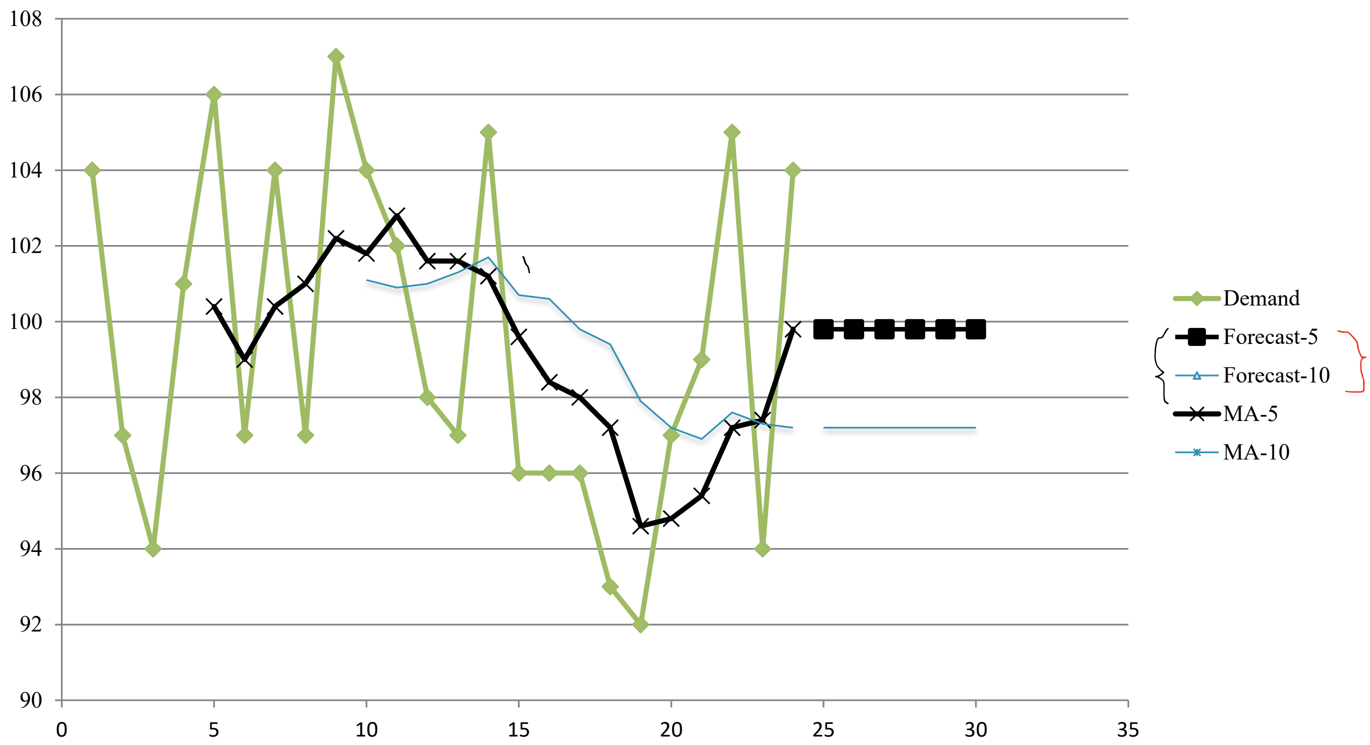
- Average only the most recent data points

$$y_t = \frac{1}{N} \sum_{i=t-N}^{t-1} D_i$$

取今天
之前的
N天

- Smooth out noise
- Can respond to change in process

Moving Average (MA)



Weighted Moving Average

- A generalization of MA with weights

simple MA

即把 W =1

$$y_t = \frac{\sum_{i=t-N}^{t-1} w_i D_i}{\sum_{i=t-N}^{t-1} w_i}$$

weight 的例子

e.g.,

$$w_{t-1} = N, w_{t-2} = N - 1, \ldots, w_{t-N} = 1$$

越近，weight ↑

Exponential Smoothing

- Adjust forecast based on the recent data point

observed　　　　forcast

$$y_t = \alpha D_{t-1} + (1-\alpha) y_{t-1}$$

$\alpha \in (0,1)$

- It is a weighted average of all historical data points, with the weight decreasing exponentially with age

且 $\sum \alpha_i = 1$

$$y_{t-1} = \alpha D_{t-2} + (1-\alpha) y_{t-2}$$

$\alpha \to 0$时，

$$y_t = \alpha D_{t-1} + \alpha(1-\alpha) D_{t-2} + (1-\alpha)^2 y_{t-2}$$

$\downarrow$

$$y_t = \sum_{i=0}^{\infty} \alpha(1-\alpha)^i D_{t-i-1} = \sum_{i=0}^{\infty} \alpha_i D_{t-i-1}$$ 公式

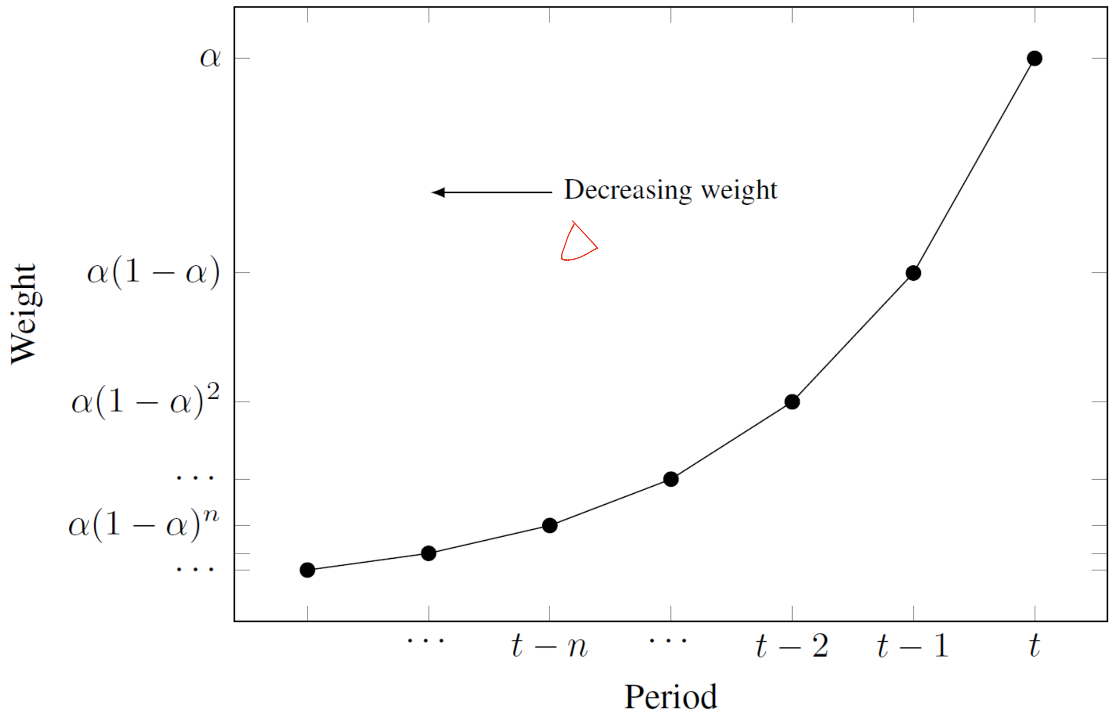
$\alpha(1-\alpha)^i$

$\sim \alpha(e^{-\alpha i}$

$\alpha_i = \alpha(1-\alpha)^i$

Exponential Smoothing

显然 decreasing，∵ $1-\alpha<1$



Double Exponential Smoothing (Holt)

带trend情况

Double exponential smoothing can be used to forecast demands with a linear trend

base signal

$$D_t = I + tS + \epsilon_t$$ — iid shock

slope

The predictor consists of base and slope:

prediction

$$y_t = I_{t-1} + S_{t-1}$$

y_{t-1}

$$\begin{cases} I_t = \alpha D_t + (1-\alpha)\left(I_{t-1} + S_{t-1}\right) \\ S_t = \beta\left(\underbrace{I_t - I_{t-1}}_{\text{recent direction}}\right) + (1-\beta)S_{t-1} \end{cases}$$

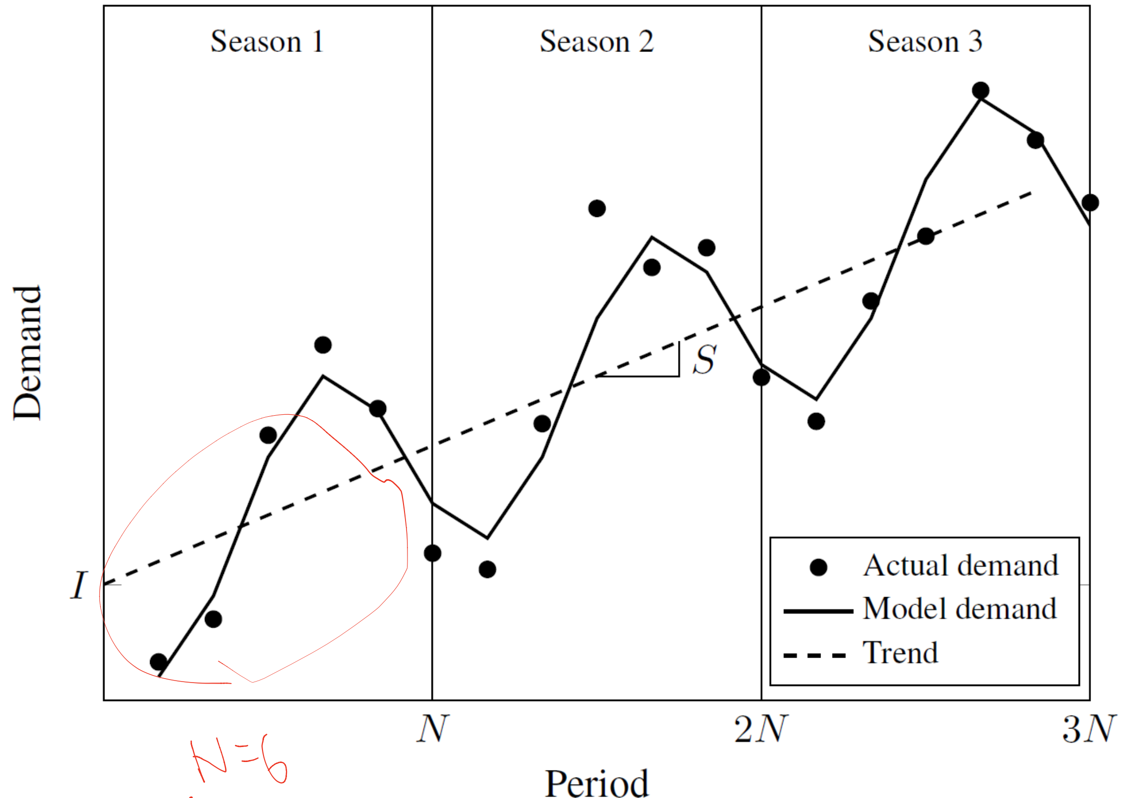
修正 base

修正 slope

Alpha is the smoothing constant and Beta is the trend constant

Triple Exponential Smoothing (Holt-Winters)

- Random demands with trend and seasonality



N=6

Triple Exponential Smoothing (Holt-Winters)

还有seasonal.

Demand model

N=2

base slope

e.g. C1=0.5

model

$$D_t = (I + tS)c_t + \epsilon_t \qquad \sum c_t = N$$

C2=1.5

seasonal factor

e.g. 一个节点

The predictor

forecast

$$y_t = (I_{t-1} + S_{t-1})\, c_{t-N}$$

Basic idea is to "de-trend" and "de-seasonalize"

$$\begin{cases} I_t = \alpha \boxed{\dfrac{D_t}{c_{t-N}}} + (1-\alpha)\left(I_{t-1} + \boxed{S_{t-1}}\right) \\ S_t = \beta\left(I_t - I_{t-1}\right) + (1-\beta)S_{t-1} \\ c_t = \gamma \dfrac{D_t}{I_t} + (1-\gamma)c_{t-N} \end{cases}$$

类似于N层周期

再看一遍!

Python Time!

- statsmodels.tsa.seasonal

- statsmodels.tsa.holtwinters

Lec1 的

最后10分钟

讲了python.

import seasonal_decompose

3: simple Exp smoothing