02417: Time Series Analysis

Modelling Reference

Peder Bacher
DTU Compute

Based on material previous material from the course

February 14, 2025

Outline

- NOT BOLD:
  - Upper-case is random variable: Y_t
  - Lower-case is a value (e.g. observation of a random variable): y_t, x_t
- BOLD:
  - Upper-case is a matrix: $\boldsymbol{X}$, $\boldsymbol{R}_t$
  - Lower-case is a vector (column): $\boldsymbol{x}$, $\boldsymbol{h}_t$

Hence, in bold there is no destinction of random variables vs. known values (e.g. observations).

Greek letters are usually used for parameters: not bold (single value) vs. bold (usually a vector, exception $\boldsymbol{\Sigma}$ is a matrix):

- NOT BOLD: θ is a random variable and σ is a known value
- BOLD: $\boldsymbol{\theta}$ is a random vector, but it could be a matrix!

General linear model (GLM)

On “per time point” form:

$$Y_t = \theta_1 x_{1,t} + \theta_2 x_{2,t} + \ldots + \theta_p x_{p,t} + \varepsilon_t$$

where $t = 1, \ldots, N$ and $\varepsilon_t \sim N(0, \sigma^2)$ and i.i.d.

- Y_t: is a random variable: Model output
- $x_{i,t}$: is a “known” value: Model input
- θ_i: is a fixed value: Parameter (or coefficient)
- ε_t is a random variable: Error

The model has p inputs and p parameters.

Note, the parameters (θ_i) can be considered random variables e.g. when carying out tests: Take a new sample then they will vary.

General linear model (GLM)

On matrix form:

$$\boldsymbol{y} = \boldsymbol{X}^T \boldsymbol{\theta} + \boldsymbol{\varepsilon}$$

$$\boldsymbol{y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix} \quad \boldsymbol{X} = \begin{bmatrix} x_{1,1} & x_{2,1} & \cdots & x_{p,1} \\ x_{1,2} & x_{2,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1,N} & x_{2,N} & \cdots & x_{p,N} \end{bmatrix} \quad \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix} \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix}$$

$\boldsymbol{X}$ is called the design matrix: One column per input.

General linear model (GLM)

Ordinary least squares (OLS) parameter estimation:

$$S_t(\boldsymbol{\theta}) = \sum_{t=1}^{N} \varepsilon_t^2$$

$$\widehat{\boldsymbol{\theta}}_t = \arg\min_{\theta} S_t(\boldsymbol{\theta}) = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

We simply multiply matrices (and some inversion): The estimates that minimize sum of squares are always found ($\boldsymbol{X}$ must have full rank).

In addition we have for statistics:

$$\widehat{\sigma}^2 = \boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}/(N-p)$$

$$V[\widehat{\boldsymbol{\theta}}] = \widehat{\sigma}^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}$$

For prediction we have the point value and the prediction interval:

$$\widehat{Y}_{t+l} = E(Y_{t+l}|\boldsymbol{x}_{t+l}) = \boldsymbol{x}_{t+l}^T\hat{\boldsymbol{\theta}}$$

$$\widehat{Y}_{t+l} \pm t_{\alpha/2}\sqrt{V\hat{\varepsilon}_{t+l}} = \widehat{Y}_{t+l} \pm t_{\alpha/2}\hat{\sigma}\sqrt{1+\boldsymbol{x}_{t+l}^T(\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{x}_{t+l})^{-1}}$$

where $t_{\alpha/2}$ is the $\alpha/2$ quantile of the t-distribution with $(N-p)$ degrees of freedom.

LS-estimate at time t

GLM on matrix form at time t:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_t \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{2,1} & \cdots & x_{p,1} \\ x_{1,2} & x_{2,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1,t} & x_{2,t} & \cdots & x_{p,t} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_t \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_t \end{bmatrix}$$

set

$$\boldsymbol{x}_t^T = \begin{bmatrix} x_{1,t} & x_{2,t} & \dots & x_{p,t} \end{bmatrix}$$

then:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_t^T \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_t \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_t \end{bmatrix}$$

$$\boldsymbol{y}_t = \boldsymbol{X}_t^T \boldsymbol{\theta} + \boldsymbol{\varepsilon}_t$$

From one to the next time step

LS-estimate based on t observations:

$$\widehat{\boldsymbol{\theta}}_t = (\boldsymbol{X}_t^T \boldsymbol{X}_t)^{-1} \boldsymbol{X}_t^T \boldsymbol{y}_t = \boldsymbol{R}_t^{-1} \boldsymbol{h}_t$$

The trick is defining a matrix:

$$\boldsymbol{R}_t = \boldsymbol{X}_t^T \boldsymbol{X}_t = \boldsymbol{x}_1 \boldsymbol{x}_1^T + \boldsymbol{x}_2 \boldsymbol{x}_2^T + \ldots + \boldsymbol{x}_t \boldsymbol{x}_t^T = \sum_{s=1}^{t-1} \boldsymbol{x}_s \boldsymbol{x}_s^T + \boldsymbol{x}_t \boldsymbol{x}_t^T$$

$$= \boldsymbol{R}_{t-1} + \boldsymbol{x}_t \boldsymbol{x}_t^T$$

and a vector:

$$\boldsymbol{h}_t = \boldsymbol{X}_t^T \boldsymbol{y}_t = \boldsymbol{x}_1 Y_1 + \boldsymbol{x}_2 Y_2 + \ldots + \boldsymbol{x}_t Y_t = \sum_{s=1}^{t-1} \boldsymbol{x}_s Y_s + \boldsymbol{x}_t Y_t$$

$$= \boldsymbol{h}_{t-1} + \boldsymbol{x}_t Y_t$$

So in each time step we can update $\boldsymbol{R}_t$ and $\boldsymbol{h}_t$ with the new data and calculate estimates again.

Eliminating $\boldsymbol{h}_t$ we can get the RLS algorithm:

$$\boldsymbol{R}_t = \boldsymbol{R}_{t-1} + \boldsymbol{x}_t \boldsymbol{x}_t^T$$

$$\widehat{\boldsymbol{\theta}}_t = \widehat{\boldsymbol{\theta}}_{t-1} + \boldsymbol{R}_t^{-1} \boldsymbol{x}_t (Y_t - \boldsymbol{x}_t^T \widehat{\boldsymbol{\theta}}_{t-1})$$

Steps on the way to the Kalman filter

You will appriciate it later in the course! ;-)

Weighted Least Squares (WLS)

We can actually put different weights on data points.

The model change from:

$$\text{Equal variance: } V[\varepsilon_t] = \sigma^2 \text{ for all } t = 1, \dots, N$$
$$\text{into: } V[\boldsymbol{\varepsilon}] = \sigma^2 \boldsymbol{\Sigma} \text{ where } \boldsymbol{\Sigma} \text{ is known.}$$

So, if we know some e.g. that the variance is higher for some parts of the data, we can take that into account.

- The parameter estimates are then

$$\widehat{\boldsymbol{\theta}}_{\text{WLS}} = (\boldsymbol{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{x})^{-1} \boldsymbol{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{Y}$$

  (if $\boldsymbol{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{x}$ is invertible)

- An estimate of σ^2 is

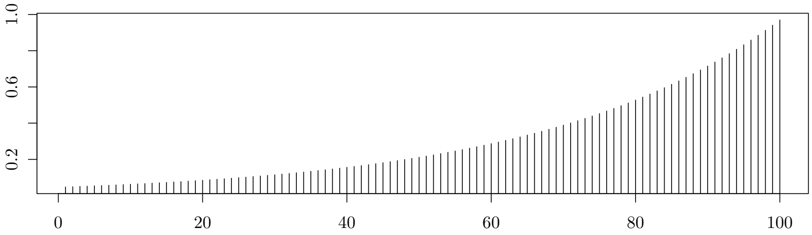
$$\widehat{\sigma}^2 = \frac{1}{N-p} (\boldsymbol{Y} - \boldsymbol{x}\widehat{\boldsymbol{\theta}})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{Y} - \boldsymbol{x}\widehat{\boldsymbol{\theta}})$$

Time adaptive estimation

- Let's use weights that decrease back in time:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \lambda^t & 0 & \cdots & 0 \\ 0 & \lambda^{t-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda^1 \end{bmatrix}$$

- $\lambda = 1$: What we did with the basic OLS
- $0 < \lambda < 1$: We "forget" in an exponential manner



The RLS with forgetting

We can achieve the same exponential forgetting with RLS by:

$$\boldsymbol{R}_t = \lambda(t)\boldsymbol{R}_{t-1} + \boldsymbol{x}_t\boldsymbol{x}_t^T$$
$$\widehat{\boldsymbol{\theta}}_t = \widehat{\boldsymbol{\theta}}_{t-1} + \boldsymbol{R}_t^{-1}\boldsymbol{x}_t(Y_t - \boldsymbol{x}_t^T\widehat{\boldsymbol{\theta}}_{t-1})$$

It's a super computational effective and fast scheme: We get new data Y_t and $\boldsymbol{x}_t$, update the parameters, down-weighting old data...without keeping all data in the computer memory.

We do need a more techniques

We will need estimation using:

- Maximum likelihood estimation

We will ...