Time Series and Sequence Learning

General AR models, Estimation

Fredrik Lindsten, Linköping University

Likelihood decomposition

Recall, the joint pdf $p(y_{1:n})$ can be factorized as

$$p(y_{1:n}) = \prod_{t=1}^{n} p(y_t \mid y_{1:t-1}).$$

A model for $p(y_t \mid y_{1:t-1})$ tells us how the current value y_t depends on the past values $y_{1:t-1}$.

ex) AR(1): model:

$$p(y_t \mid y_{1:t-1}) = p(y_t \mid y_{t-1}) = \mathcal{N}\left(y_t \mid a y_{t-1}, \sigma_\varepsilon^2\right).$$

Auto-regressive models of higher order

Idea: Generalize the first-order AR model and assume a linear dependence on **a fixed number of the most recent values.**

Def: A linear auto-regressive (AR) model of **order** p is given by

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \dots a_p y_{t-p} + \varepsilon_t, \qquad \varepsilon_t \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_\varepsilon^2).$$

Auto-regressive models of higher order

Idea: Generalize the first-order AR model and assume a linear dependence on a fixed number of the most recent values.

Def: A linear auto-regressive (AR) model of **order** p is given by

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \dots a_p y_{t-p} + \varepsilon_t, \qquad \varepsilon_t \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_\varepsilon^2).$$

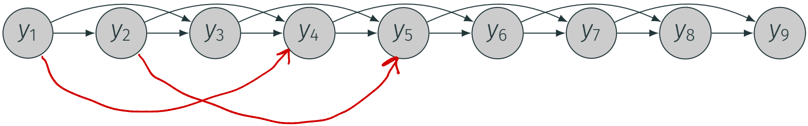
Equivalently, we can write the **AR(p) model** as

$$p(y_t \mid y_{1:t-1}) = \mathcal{N}\left(y_t \;\middle|\; \sum_{j=1}^{p} a_j y_{t-j}, \sigma_\varepsilon^2\right).$$

Graphical representation

The dependencies of an AR(p) model can be illustrated graphically.

ex) AR($\cancel{2}$): AR(3):



Estimating an AR(p) model

Assume that we have **observed** $y_{1:n}$ and wish to fit an AR(p) model to the data.

The log-likelihood given by

$$\log p(y_{1:n};\theta) = \sum_{t=1}^{n} \log \underbrace{p(y_t \mid y_{1:t-1};\theta)}_{= N(y_t \mid \sum_{j=1}^{p} a_j y_{t-j},\ \sigma_\varepsilon^2)}$$

$$= -\frac{1}{2\sigma_\varepsilon^2}\sum_{t=1}^{n}\left(y_t - \sum_{j=1}^{p} a_j y_{t-j}\right)^2$$

since log of normal distN(x|u,sigma^2) = log N(x,u,sig^2) = -1/2 log(s pi sig^2) - (x-u)^2/(2sig^2)
replace x=yt and u=sum(a_j y_{t-j}), so sum all, we have log(y_{1:n},theta) = sum(log(...))
const term -n/2 log(2*pi*sig^2) is ignored, so it become like above

- We get a standard least-squares regression problem!
- For the t:th term, $y_{t-p:t-1}$ can be viewed as a known input and y_t as the output.

Least-squares estimation of AR(p)

The least squares loss for estimating an AR(p) model is given by

$$L(\theta) = \frac{1}{n}\sum_{t=1}^{n}\left(y_t - \underbrace{\theta^\top \phi_t}_{\sum_{j=1}^{p} a_j y_{t-j}}\right)^2$$

with

$$\theta = \begin{pmatrix} a_1 & \dots & a_p \end{pmatrix}^\top \qquad \text{and} \qquad \phi_t = \begin{pmatrix} y_{t-1} & \dots & y_{t-p} \end{pmatrix}^\top$$

Least-squares estimation of AR(p)

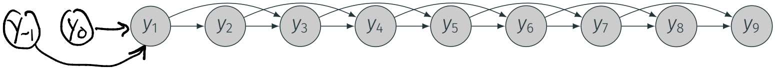
The least squares loss for estimating an AR(p) model is given by

$$L(\theta) = \frac{1}{n} \sum_{t=1}^{n} \left(y_t - \theta^\top \phi_t \right)^2$$

with

$$\theta = \begin{pmatrix} a_1 & \dots & a_p \end{pmatrix}^\top \qquad \text{and} \qquad \phi_t = \begin{pmatrix} y_{t-1} & \dots & y_{t-p} \end{pmatrix}^\top$$

Caveat! The input ϕ_t depends on $y_0, y_{-1}, \dots, y_{-p+1}$ for $t \leq p$.



Least-squares estimation of AR(p)

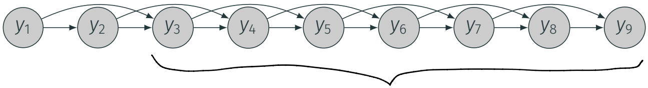
The least squares loss for estimating an AR(p) model is given by

$$L(\theta) \approx \frac{1}{n-p} \sum_{t=p+1}^{n} \left(y_t - \theta^\top \phi_t \right)^2$$

with

$$\theta = \begin{pmatrix} a_1 & \dots & a_p \end{pmatrix}^\top \qquad \text{and} \qquad \phi_t = \begin{pmatrix} y_{t-1} & \dots & y_{t-p} \end{pmatrix}^\top$$

Pragmatic solution: Ignore the first p terms of the loss function.



Least-squares estimation of AR(p)

Solution given by standard least-squares,

$$\widehat{\theta} = \left(\Phi^\top \Phi\right)^{-1} \Phi^\top \mathbf{y}$$

where

$$\mathbf{y} = \begin{pmatrix} y_{p+1} \\ y_{p+2} \\ \vdots \\ y_n \end{pmatrix}, \qquad \Phi = \begin{pmatrix} y_p & y_{p-1} & \cdots & y_1 \\ y_{p+1} & y_p & \cdots & y_2 \\ \vdots & \vdots & \ddots & \vdots \\ y_{n-1} & y_{n-2} & \cdots & y_{n-p} \end{pmatrix}$$

Least-squares estimation of AR(p)

Solution given by standard least-squares,

$$\widehat{\theta} = \left(\Phi^\top \Phi\right)^{-1} \Phi^\top \mathbf{y}$$

where

$$\mathbf{y} = \begin{pmatrix} y_{p+1} \\ y_{p+2} \\ \vdots \\ y_n \end{pmatrix}, \qquad \Phi = \begin{pmatrix} y_p & y_{p-1} & \cdots & y_1 \\ y_{p+1} & y_p & \cdots & y_2 \\ \vdots & \vdots & \ddots & \vdots \\ y_{n-1} & y_{n-2} & \cdots & y_{n-p} \end{pmatrix}$$

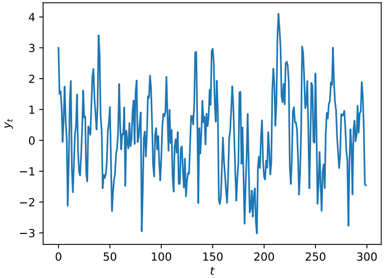
Noise variance can be estimated by the **mean squared error (MSE)**,

$$\widehat{\sigma}_\varepsilon^2 = \frac{1}{n-p} \sum_{t=p+1}^{n} \left(y_t - \widehat{\theta}^\top \phi_t\right)^2$$

ex) Toy model

We simulate an AR(3) model for $n = 300$ time steps,

$$y_t = 0.9 y_{t-1} - 0.4 y_{t-2} + 0.2 y_{t-3} + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, 1)$$



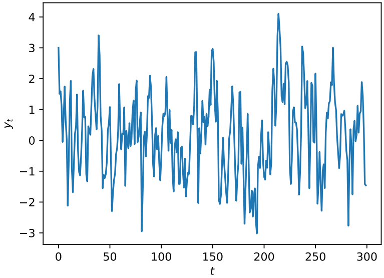
ex) Toy model

We simulate an AR(3) model for $n = 300$ time steps,

$$y_t = 0.9 y_{t-1} - 0.4 y_{t-2} + 0.2 y_{t-3} + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, 1)$$

Estimating the model parameters with OLS gives:
$\widehat{\theta} = (0.84, -0.33, 0.16)$ and $\widehat{\sigma}_\varepsilon^2 = 0.95$.

let y_t-1 = lag1_t
let y_t-2 = lag2_t
let y_t-3 = lag3_t



we construct matrix X as
lag1_4 lag2_4 lag3_4
lag1_5 lag3_5
lag1_n lag3_n
we got response y = [y4,...yn]
then hat(theta) = (X^tX)^{-1}X^{t}y
then hat(epsilon) = y-x*hat(theta)
then hat_sigma^2 = hat(epsilon)^t*hat(epsilon)/(n-3)

or use the formula provided

ex) Toy model

We simulate an AR(3) model for $n = 300$ time steps,

$$y_t = 0.9 y_{t-1} - 0.4 y_{t-2} + 0.2 y_{t-3} + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, 1)$$

Estimating the model parameters with OLS gives:
$\widehat{\theta} = (0.84, -0.33, 0.16)$ and $\widehat{\sigma}_{\varepsilon}^2 = 0.95$.