Linear Model: Least Square Approach

TF4063

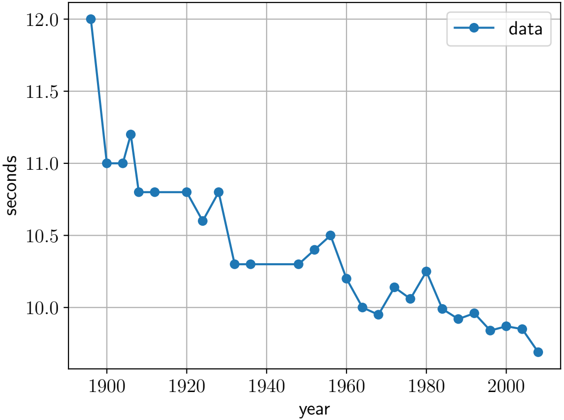
Fadjar Fathurrahman

Program Studi Teknik Fisika
Institut Teknologi Bandung

The material in this note is based on Rogers2017.

Example dataset: `olympic100m`

Let's start by studying a with a simple dataset.



Simple Linear Model

Linear model:

$$t = f(x; w_0, w_1) = w_0 + w_1 x$$

Loss function:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} (t_n - f(x_n; w_0, w_1))^2$$

Finding the parameters

Find the parameters (w_0, w_1) by using minimization procedures:

$$\arg\min_{w_0, w_1} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}_n$$

For our particular case of Eq. (**??**), we can found this analytically, i.e. calculating the first derivatives of $\mathcal{L}$ with respect to w_0 and w_1, equating them to zero, and solve the resulting equations for w_0 and w_1.

For more general cases, we can use various numerical optimization procedures such as gradient descent methods.

Finding the parameters

We begin by writing our loss function as:

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}\left(t_n - (w_0 + w_1 x_n)\right)^2$$
$$= \frac{1}{N}\sum_{n=1}^{N}\left(w_1^2 x_n^2 + 2w_1 x_n(w_0 - t_n) + w_0^2 - 2w_0 t_n + t_n^2\right)$$

Now we find the first derivatives of $\mathcal{L}$ with respect to w_0, w_1 and equating them to zero.

$$\frac{\partial \mathcal{L}}{\partial w_1} = 2w_1\frac{1}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \frac{2}{N}\left(\sum_{n=1}^{N} x_n(w_0 - t_n)\right) = 0$$
$$\frac{\partial \mathcal{L}}{\partial w_0} = 2w_0 + 2w_1\frac{1}{N}\left(\sum_{n=1}^{N} x_n\right) - \frac{2}{N}\left(\sum_{n=1}^{N} t_n\right) = 0$$

Finding the parameters

We obtain

$$
\begin{aligned}
w_1 &= \frac{\overline{xt} - \overline{x}\,\overline{t}}{\overline{x^2} - \overline{x}^2} \\
w_0 &= \overline{t} - w_1 \overline{x}
\end{aligned}
$$

where symbols with overline denotes their average value, for examples

$$
\begin{aligned}
\overline{x} &= \frac{1}{N} \sum_{n=1}^{N} x_n \\
\overline{t} &= \frac{1}{N} \sum_{n=1}^{N} t_n
\end{aligned}
$$

Application to `olympic100m`

XXX

Linear model: matrix-vector notation

Input and parameter vectors:

$$\mathbf{x}_n \equiv \begin{bmatrix} 1 \\ x_n \end{bmatrix}, \quad \mathbf{w} \equiv \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$$

Linear model:

$$f(x_n; w_0, w_1) = \mathbf{w}^\mathsf{T}\mathbf{x}_n$$

Loss function:

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n\right)^2 \tag{1}$$

Design matrix:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\mathsf{T} \\ \mathbf{x}_2^\mathsf{T} \\ \vdots \\ \mathbf{x}_N^\mathsf{T} \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix}$$

$$\mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{bmatrix}$$

Finding the parameters

$$\mathcal{L} = \frac{1}{N}\left(\mathbf{t} - \mathbf{X}\mathbf{w}\right)^{\top}\left(\mathbf{t} - \mathbf{X}\mathbf{w}\right)$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0$$

$$\mathcal{L} = \frac{1}{N}\left(\mathbf{t}^{\top}\mathbf{t} + \left(\mathbf{X}\mathbf{w}\right)^{\top}\mathbf{X}\mathbf{w} - \mathbf{t}\mathbf{X}\mathbf{w} - \left(\mathbf{X}\mathbf{w}\right)^{\top}\mathbf{t}\right)$$

$$= \frac{1}{N}\left(\mathbf{w}^{\top}\mathbf{X}^{\top}\mathbf{X}\mathbf{w} - 2\mathbf{w}^{\top}\mathbf{X}^{\top}\mathbf{t} + \mathbf{t}^{\top}\mathbf{t}\right)$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \frac{2}{N}\left(\mathbf{X}^{\top}\mathbf{X}\mathbf{w} - \mathbf{X}^{\top}\mathbf{t}\right) = 0$$

$$\mathbf{X}^{\top}\mathbf{X}\mathbf{w} = \mathbf{X}^{\top}\mathbf{t}$$

$$\boxed{\mathbf{w} = \left(\mathbf{X}^{\top}\mathbf{X}\right)^{-1}\mathbf{X}^{\top}\mathbf{t}}$$

Application to `olympic100m` (using matrix-vector notation)

```python
def hello():
  pass
```

Example slide

Column 1

Column 2