Linear Regression

Christos Dimitrakakis

October 6, 2024

Outline

Simple linear regression

Input and output

- Data pairs $(\boldsymbol{x}_t, y_t)$, $t = 1, \dots, T$.
- Input $\boldsymbol{x}_t \in \mathbb{R}^n$
- Output $y_t \in \mathbb{R}$.

Predicting the conditional mean $\mathbb{E}[y_t|\boldsymbol{x}_t]$

- Parameters $\beta \in \mathbb{R}^n$
- Function $f_\beta : \mathbb{R}^n \to \mathbb{R}$, defined as

$$f_\beta(\boldsymbol{x}_t) = \beta^\top \boldsymbol{x}_t = \sum_{i=1}^{n} \beta_i x_{t,i}$$

Two views of the problem

Learning as Optimisation

Miniminise mean-squared error.

$$\min_{\beta} \sum_{t=1}^{T} [y_t - f_\beta(\boldsymbol{x}_t)]^2$$

Learning as inference

Assume a Gaussian noise model:

$$y_t = f(\boldsymbol{x}_t) + \epsilon_t, \qquad \epsilon_t \sim \mathrm{Normal}(0, \sigma)$$

This leads to the conditional density

$$p_\beta(y_t|\boldsymbol{x}_t) \propto \exp(-[y_t - f_\beta(\boldsymbol{x}_t)]^2/2\sigma^2)$$

Maximising the log-likelihood is equivalent to minimising mean-squared error:

$$\arg\max_{\beta} \sum \ln p_\beta(y_t|\boldsymbol{x}_t) = \arg\min_{\beta} \sum_t |y_t - f_\beta(\boldsymbol{x}_t)|^2$$

Gradient descent algorithm

Minimising a function

$$\min_{\beta} f(\beta) \geq f(\beta') \forall \beta', \qquad \beta^* = \arg\min_{\beta} f(\beta) \Rightarrow f(\beta^*) = \min_{\beta} f(\beta)$$

Gradient descent for minimisation

- Input β_0
- For $n = 0, \ldots, N$:
- $\beta_{n+1} = \beta_n - \eta_n \nabla_{\beta} f(\beta_n)$

Step-size η_n

- η_n fixed: for online learning
- $\eta_n = c/[c + n]$ for asymptotic convergence
- $\eta_n = \arg\min_{\eta} f(\theta_n + \eta \nabla_{\beta})$: Line search.

Gradient desecnt for squared error

Cost gradient

Using the chain rule of differentiation:

$$
\begin{aligned}
\nabla_\beta \ell(\beta) &= \nabla \sum_{t=1}^{T} [y_t - \pi_\beta(\boldsymbol{x}_t)]^2 \\
&= \sum_{t=1}^{T} \nabla [y_t - \pi_\beta(\boldsymbol{x}_t)]^2 \\
&= \sum_{t=1}^{T} 2[y_t - \pi_\beta(\boldsymbol{x}_t)][-\nabla \pi_\beta(\boldsymbol{x}_t)]^2
\end{aligned}
$$

Parameter gradient

For a linear regressor:

$$
\frac{\partial}{\partial \beta_j} \pi_\beta(\boldsymbol{x}_t) = x_{t,j}.
$$

Stochastic gradient descent algorithm

When f is an expectation

$$f(\beta) = \int_X dP(x) g(x, \beta).$$

Replacing the expectation with a sample:

$$\begin{aligned}\nabla f(\beta) &= \int_X dP(x) \nabla g(x, \beta) \\ &\approx \frac{1}{K} \sum_{k=1}^{K} \nabla g(x^{(k)}, \beta), \qquad\qquad x^{(k)} \sim P.\end{aligned}$$

Some matrix algebra

The identity matrix $I \in \mathbb{R}^{n\times n}$

- For this matrix, $I_{i,i} = 1$ and $I_{i,j} = 0$ when $j \neq i$.
- $Ix = x$ and $IA = A$.

The inverse of a matrix $A \in \mathbb{R}^{n\times n}$

A^{-1} is called the inverse of A if

- $AA^{-1} = I$.
- or equivalently $A^{-1}A = I$.

The pseudo-inverse of a matrix $A \in \mathbb{R}^{n\times m}$

- $\tilde{A}^{-1}$ is called the left pseudoinverse of A if $\tilde{A}^{-1}A = I$.
$$\tilde{A}^{-1} = (A^\top A)^{-1}A^\top, \qquad n > m$$
- $\tilde{A}^{-1}$ is called the right pseudoinverse of A if $A\tilde{A}^{-1} = I$.
$$\tilde{A}^{-1} = A^\top (AA^\top)^{-1}, \qquad m > n$$

Analytical Least-Squares Solution

We need to solve the following equations for A:

$$\begin{aligned} y_1 &= \boldsymbol{x}_1^\top \beta \\ \dots & \quad \dots \\ y_t &= \boldsymbol{x}_t^\top \beta \\ \dots & \quad \dots \\ y_T &= \boldsymbol{x}_T^\top \beta \end{aligned}$$

We can rewrite it in matrix form:

$$\begin{pmatrix} y_1 \\ \vdots \\ y_t \\ \vdots \\ y_T \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_t^\top \\ \vdots \\ \boldsymbol{x}_T^\top \end{pmatrix} \beta$$

Resulting in

$$\boldsymbol{y} = X\beta$$

So we can use the left-pseudo inverse $\tilde{X}^{-1}$ to obtain

$$\beta = \tilde{X}^{-1}\boldsymbol{y}$$

The coefficients

- β_i tells us how much y is correlated with $x_{t,i}$
- However, multiple correlations might be evident.

Linear regression exercises

- Exercises 8, 13 from ISLP
- A variant of Ex. 13 but with Y generated independently of X.