Least squares and Maximum Likelihood Estimation (MLE)

Friday, 30 April 2021 07:42

Least Squares for a linear model

$$y = a_1 x_1 + .. + a_k x_k + \varepsilon \qquad \text{where} \quad \varepsilon \sim N(0, \sigma)$$

in other words

$$\hat{y} = a_1 x_1 + .. + a_k x_k$$

Problem: given a dataset $D = \{(\vec{x}_n, y_n)\}$
where $x_n = (x_{n1}, .., x_{nk}) \in \mathbb{R}^k$
find $a_1, .., a_k \in \mathbb{R}$ which
minimize the squared error

$$SE = \sum_n (y_n - (a_1 x_{n1} + ... + a_k x_{nk}))^2$$

SE is a quadratic function of $a_1, ..., a_k$ and the coefficient in front of the quadratic term is positive hence the minimum is in the point $(a_1, .., a_k)$ where

$$\forall_i \quad \frac{\partial}{\partial a_i} SE(a_0, .., a_k) = 0$$

For simplicity of notation consider the 2D case where we need to find (a_1, a_2)

We need to find a_1, a_2 where

$$\begin{cases} \dfrac{\partial}{\partial a_1} \sum\limits_n \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)^2 = 0 \\ \dfrac{\partial}{\partial a_2} \sum\limits_n \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)^2 = 0 \end{cases}$$

We have

$$\frac{\partial}{\partial a_1} \sum_n \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)^2 = \sum_n \frac{\partial}{\partial a_1} \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)^2$$

$$= \sum_n 2 \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)(-x_{n1})$$

$$= -2 \sum_n x_{n1} \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)$$

$$\frac{\partial}{\partial a_2} \sum_n \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)^2 = -2 \sum_n x_{n2} \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right)$$

Hence we have to solve the following system of linear equations:

$$\begin{cases} \sum\limits_n x_{n1} \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right) = 0 \\ \sum\limits_n x_{n2} \left(y_n - (a_1 x_{n1} + a_2 x_{n2})\right) = 0 \end{cases}$$

$$\Updownarrow$$

$$\begin{cases} \sum\limits_n x_{n1} y_n = \sum\limits_n x_{n1}^2 a_1 + \sum\limits_n x_{n1} x_{n2} a_2 \\ \sum\limits_n x_{n2} y_n = \sum\limits_n x_{n1} x_{n2} a_1 + \sum\limits_n x_{n2}^2 a_2 \end{cases}$$

$$\Updownarrow$$

$$\begin{cases} \sum_n x_{n1} y_n = a_1 \sum_n x_{n1}^2 + a_2 \sum_n x_{n1} x_{n2} \\ \sum_n x_{n2} y_n = a_1 \sum_n x_{n1} x_{n2} + a_2 \sum_n x_{n2}^2 \end{cases} \quad (*)$$

$$\Updownarrow$$

$$\begin{cases} \sum_n x_{n1} y_n \sum_n x_{n2}^2 = a_1 \sum_n x_{n1}^2 \sum_n x_{n2}^2 + a_2 \sum_n x_{n1} x_{n2} \sum_n x_{n2}^2 \\ \sum_n x_{n2} y_n \sum_n x_{n1} x_{n2} = a_1 \sum_n x_{n1} x_{n2} \sum_n x_{n1} x_{n2} + a_2 \sum_n x_{n2}^2 \sum_n x_{n1} x_{n2} \end{cases}$$

$$\Downarrow$$

$$a_1 = \frac{\sum_n x_{n2} y_n \sum_n x_{n1} x_{n2} - \sum_n x_{n1} y_n \sum_n x_{n2}^2}{\left(\sum_n x_{n1} x_{n2}\right)^2 - \sum_n x_{n1}^2 \sum_n x_{n2}^2}$$

Denoting $X_1 = (x_{11}, x_{21}, \dots, x_{n1})$ – the first coordinate of every datapoint
$X_2 = (x_{12}, x_{22}, \dots, x_{n2})$ – the second coordinate of every datapoint

we can write the formula for a_1 in a concise way

$$a_1 = \frac{(X_2 \cdot y)(X_1 \cdot X_2) - (X_1 \cdot y)\|X_2\|^2}{(X_1 \cdot X_2)^2 - \|X_1\|^2 \|X_2\|^2}$$

and for a_2:

$$a_2 = \frac{(X_1 \cdot y)(X_1 \cdot X_2) - (X_2 \cdot y)\|X_1\|^2}{(X_1 \cdot X_2)^2 - \|X_1\|^2 \|X_2\|^2}$$

The system of equations $(*)$ can be written in a more generic way

$$\begin{cases} \sum_n x_{n1} y_n = a_1 \sum_n x_{n1}^2 + a_2 \sum_n x_{n1} x_{n2} \\ \sum_n x_{n2} y_n = a_1 \sum_n x_{n1} x_{n2} + a_2 \sum_n x_{n2}^2 \end{cases}$$

$$\Updownarrow$$

$$\boxed{X^T y = X^T X A}$$

where

$$X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad A = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

Then

$$X^T y = X^T X A \quad | \cdot (X^T X)^{-1}$$

$$\Updownarrow$$

$$\boxed{A = (X^T X)^{-1} X^T y}$$

This form holds also for k-dimensional x vectors

$X^T X$ is a $k \times k$ dimensional matrix

$\Rightarrow$ if k is small, inverting $X^T X$ is not costly

ALS - Alternating Least Squares

Recall that the matrix factorization problem is given by

$$\min_{p_u, q_i \in \mathbb{R}^d} \sum_{(u,i) \in K} (r_{ui} - q_i^T p_u)^2$$

If we fix the item representations q_i, then the problem becomes independent for every user and takes the following form:

$$\min_{p_u \in \mathbb{R}^d} \sum_{i \in K(u)} \left(r_{u,i} - (q_{i1} p_{u1} + \dots + q_{id} p_{ud}) \right)^2$$

where this expression has to be minimized for every user over all possible values of $p_u = (p_{u1}, p_{u2}, \dots, p_{ud})$

This is the Linear Least Squares problem!

Therefore

$$p_u = (X^T X)^{-1} X^T y$$

where

$$X = \begin{bmatrix} q_{11} & q_{12} & \dots & q_{1d} \\ q_{21} & q_{22} & \dots & q_{2d} \\ & & \vdots & \\ q_{m1} & q_{m2} & \dots & q_{md} \end{bmatrix} \qquad y = \begin{bmatrix} r_{u1} \\ r_{u2} \\ \vdots \\ r_{um} \end{bmatrix}$$

ALS

1. Initialize all user and item representation vectors p_u and q_i with random values
2. Iterate until convergence
   (i.e. changing of representations less than ε)
   2.a. Set all item representations q_i and solve the Linear Least Squares problem for user representations p_u
   2.b. Set all user representations p_u and solve the Linear Least Squares problem for item representations q_i

+ Scalability

Because the Linear Least Squares problem is independent for every user/item, the algorithm can be highly parallelized

Maximum Likelihood Estimation (MLE)

Consider again the following linear model

$$y = a_1 x_1 + .. + a_k x_k + \varepsilon \qquad \text{where} \quad \varepsilon \sim N(0, \sigma)$$

Assuming this model is true, the likelihood of observing a datapoint $(x_1, x_2, \dots, x_k, y)$ in the data is equal to

$$L(\varepsilon) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\varepsilon}{\sigma}\right)^2} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-(a_1x_1+a_2x_2+\dots+a_kx_k)}{\sigma}\right)^2}$$

The idea behind MLE is that for a given set of observed datapoints $\{(\vec{x}_n, y_n)\} = \{(x_{n1}, x_{n2}, \dots, x_{nk}, y_n)\}$ we want to find such model parameters $a_1, a_2, \dots, a_k$ that the likelihood of observing such dataset is maximal, i.e. we want to solve

$$\max_{a_1,\dots,a_k} \prod_n \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-(a_1x_1+a_2x_2+\dots+a_kx_k)}{\sigma}\right)^2}$$

This expression can be further simplified since:

$$\operatorname*{argmax}_{a_1,\dots,a_k} \prod_n \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-(a_1x_1+a_2x_2+\dots+a_kx_k)}{\sigma}\right)^2}$$

$$= \operatorname*{argmax}_{a_1,\dots,a_k} \prod_n e^{-\frac{1}{2}\left(\frac{y-(a_1x_1+a_2x_2+\dots+a_kx_k)}{\sigma}\right)^2}$$

$$= \operatorname*{argmax}_{a_1,\dots,a_k} e^{-\frac{1}{2}\sum_n \left(\frac{y-(a_1x_1+a_2x_2+\dots+a_kx_k)}{\sigma}\right)^2}$$

(since $e^a e^b = e^{a+b}$)

$$= \underset{a_1, \ldots, a_k}{\operatorname{argmax}} \; \ell$$

$$= \underset{a_1, \ldots, a_k}{\operatorname{argmin}} \; \sum_n \left(\frac{y - (a_1 x_1 + a_2 x_2 + \ldots + a_k x_k)}{\sigma} \right)^2$$

$$= \underset{a_1, \ldots, a_k}{\operatorname{argmin}} \; \sum_n \left(y - (a_1 x_1 + a_2 x_2 + \ldots + a_k x_k) \right)^2$$

$\left(e^a e^b = e^{a+b} \right)$

$\left(\text{since } e^{-x} \text{ is decreasing} \right)$

But this is exactly Least Squares !

MLE and Least Squares are equivalent if the noise in the data is normal (Gaussian)

Note

MLE is a powerful and general method which can be used with any probability distribution, for instance Bernoulli, Binomial, Poisson, Exponential, Gamma, Beta

This method often inspires the loss functions for various neural networks and other models