Introduction to non-regularized regression

Jae Yun JUN KIM*

November 2, 2020

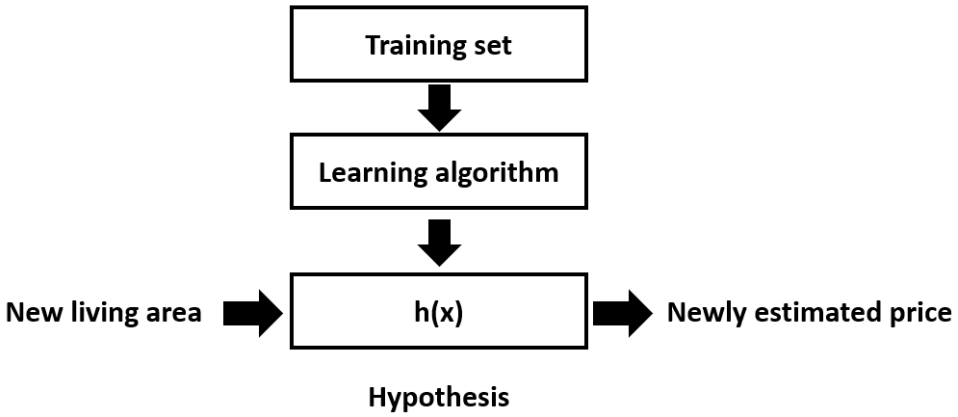
1 Motivation

Living Area [m^2]	nearest access to public transport [m]	Price [1000 Euros]
45.5	10.1	200.3
60.7	20.3	300.5
⋮	⋮	⋮

2 Notation

I	number of training examples
x	input variables/features
y	output variable / "target" variable
(x, y)	training example
$(x^{(i)}, y^{(i)})$	i^{th} training example

3 Hypothesis function



$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2, \tag{1}$$

where x_1 and x_2 are the living area size and the room number, respectively.

*ECE Paris Graduate School of Engineering, 37 quai de Grenelle 75015 Paris, France; jae-yun.jun-kim@ece.fr

For conciseness, define $x_0 = 1$ and

$$h_\theta(x) = \sum_{n=0}^{N} \theta_n x_n = \theta^T x, \tag{2}$$

where N is the number of features, and θ are called **parameters**.

4 Cost minimization problem

What we would like to do is to find θ's that minimize

$$E(\theta) = \frac{1}{2} \sum_{i=0}^{I} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right)^2. \tag{3}$$

That is,

$$\min_\theta E(\theta). \tag{4}$$

Now that the cost minimization problem is stated, we turn our attention to the question to how we can solve this problem.

There are mainly three approaches to solve this problem.

- Batch gradient descent
- Stochastic gradient descent
- Closed-form solution

5 Batch gradient descent

$$\theta_n \leftarrow \theta_n - \alpha \frac{\partial}{\partial \theta_n} E(\theta), \tag{5}$$

where

$$\begin{aligned} \frac{\partial}{\partial \theta_n} E(\theta) &= \frac{\partial}{\partial \theta_n} \frac{1}{2} \left(h_\theta(x) - y \right)^2 \\ &= 2 \frac{1}{2} \left(h_\theta(x) - y \right) \frac{\partial}{\partial \theta_n} \left(h_\theta(x) - y \right) \\ &= \left(h_\theta(x) - y \right) \frac{\partial}{\partial \theta_n} \left(\theta_0 x_0 + \cdots + \theta_n x_n + \cdots + \theta_N x_N - y \right) \\ &= \left(h_\theta(x) - y \right) x_n. \end{aligned} \tag{6}$$

Hence, the **update rule** is

$$\theta_n \leftarrow \theta_n - \alpha \left(h_\theta(x) - y \right) x_n \tag{7}$$

Repeat till convergence:
For $n = 0, 1, \cdots, N$:

$$\theta_n \leftarrow \theta_n - \alpha \sum_{i=1}^{I} \left(h_\theta(x^{(i)}) - y^{(i)} \right) x_n^{(i)}. \tag{8}$$

6 Stochastic gradient descent

However, when $I >> 1$, the *batch gradient descent* may be very inefficient. Alternatively, one can use the **stochastic gradient descent**.

Repeat until convergence:
For $n = 0, 1, \cdots, N$:

$$\text{Choose } i \sim \mathcal{U}(1, I_{max})$$

$$\theta_n \leftarrow \theta_n - \alpha \left(h_\theta(x^{(i)}) - y^{(i)}\right) x_n^{(i)}$$

This will give some approximate convergence, but it will converge more quickly. Hence, the choice between the *batch gradient descent* and the *stochastic gradient descent* is a trade-off between the accuracy and efficiency, respectively.

7 Closed-form solution

One can denote the *gradient* of the error function with respect to the *parameters* as

$$\nabla_\theta(E) = \begin{bmatrix} \frac{\partial E}{\partial \theta_0} \\ \vdots \\ \frac{\partial E}{\partial \theta_N} \end{bmatrix} \in \mathbb{R}^{N+1}. \tag{9}$$

Using this notation, we can rewrite (5) as

$$\theta \leftarrow \theta - \alpha \nabla_\theta(E). \tag{10}$$

Some facts

Suppose that $\mathbf{f} : \mathbb{R}^{I \times N} \rightarrow \mathbb{R}$. Then,

$$\nabla_A f(A) = \begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \cdots & \frac{\partial f}{\partial A_{1N}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial A_{I1}} & \cdots & \frac{\partial f}{\partial A_{IN}} \end{bmatrix} \in \mathbb{R}^{N+1}. \tag{11}$$

1) If $A \in \mathbb{R}^{I \times N}$, $trA = \sum_{i=1}^{N} A_{ii}$.

2)

$$\begin{aligned} trAB &= trBA \\ trABC &= trCAB \quad = trBCA \end{aligned} \tag{12}$$

3) If $f(A) = trAB$, then $\nabla_A trAB = B^T$.

4)

$$trA = trA^T \tag{13}$$

5) If $a \in \mathbb{R}$, $tra = a$.

6)

$$\nabla_A trABA^TC = CAB + C^TAB^T. \tag{14}$$

Let

$$X = \begin{bmatrix} (x^{(1)})^T \\ (x^{(2)})^T \\ \vdots \\ (x^{(I)})^T \end{bmatrix}, \tag{15}$$

$$X\theta = \begin{bmatrix} (x^{(1)})^T\theta \\ (x^{(2)})^T\theta \\ \vdots \\ (x^{(I)})^T\theta \end{bmatrix} = \begin{bmatrix} h_\theta(x^{(1)}) \\ h_\theta(x^{(2)}) \\ \vdots \\ h_\theta(x^{(I)}) \end{bmatrix}. \tag{16}$$

Define the output vector

$$\mathbf{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(I)} \end{bmatrix}. \tag{17}$$

Now compute

$$\mathbf{X}\theta - \mathbf{y} = \begin{bmatrix} h_\theta(x^{(1)}) - y^{(1)} \\ h_\theta(x^{(2)}) - y^{(2)} \\ \vdots \\ h_\theta(x^{(I)}) - y^{(I)} \end{bmatrix}. \tag{18}$$

Recall that $\mathbf{z}^T\mathbf{z} = \sum_i z_i^2$.

Hence,

$$(\mathbf{X}\theta - \mathbf{y})^T(\mathbf{X}\theta - \mathbf{y}) = \sum_{i=1}^{I} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 \triangleq E(\theta). \tag{19}$$

Recall that

$$\theta^T\mathbf{x} = \sum_{n=1}^{N} \theta_n x_n. \tag{20}$$

Since the problem consists in finding the parameters that minimize the error function, we set

$$\nabla_\theta E(\theta) \equiv \mathbf{0}. \tag{21}$$

But,

$$\begin{aligned} \nabla_\theta E(\theta) = \nabla_\theta \frac{1}{2}(x\theta - y)^T(x\theta - y) \\ \frac{1}{2}\nabla_\theta(\theta^T x^T x\theta - \theta^T x^T y - y^T x\theta + y^T y) \\ \frac{1}{2}\nabla_\theta tr(\theta^T x^T x\theta - \theta^T x^T y - y^T x\theta + y^T y) \\ \frac{1}{2}(\nabla_\theta tr(\theta\theta^T x^T x) - \nabla_\theta tr(y\theta^T x^T) - \nabla_\theta tr(y^T x\theta)) \end{aligned}. \tag{22}$$

On the other hand,

$$\nabla_\theta tr(\theta^T \theta^T x^T x) = x^T x \theta I + x^T x \theta I^T = x^T x \theta + x^T x \theta \tag{23}$$

$$\nabla_\theta tr(y^T x \theta) = x^T y \tag{24}$$

$$\nabla_\theta tr(\theta^T x^T y) = \nabla_\theta tr(y^T x \theta). \tag{25}$$

Hence,

$$\nabla_\theta E(\theta) = \frac{1}{2}(x^T x \theta + x^T x \theta - x^T y - x^T y) = x^T x \theta - x^T y. \tag{26}$$

Finally, by imposing $\nabla_\theta E(\theta) \equiv 0$, we have $x^T x \theta = x^T y$. By isolating θ, we have

$$\theta = (x^T x)^{-1} x^T y. \tag{27}$$