Lecture Note 1 on Gradient Descent

Harry Li, Ph.D.
Department of Computer Engineering, College of Engineering
San Jose State University, San Jose, CA 95192, USA
Email[†]: harry.li@ctione.com

***Abstract*—In this lecture note, we give a gradient descent example for its applications in Neural Networks (NN), e.g., the concept of the negative gradient -∇f follows the direction of steepest descent.**

I. Introduction

In this lecture note, we give a gradient descent example for Neural Networks (NN) applications. In particular, the basic concept of the negative gradient -∇f follows the direction of steepest descent of a given function f which can be an error function.

II. Partial Derivative vs. Gradient

Given a scalar-valued multivariable functions, e.g. the function with a multidimensional input $x_1, x_2, ..., x_n$, and a one-dimensional output as $y = f(x_1, x_2, ..., x_))$, where $f : R^n \to R$. The partial derivative of $f(x_1, x_2, ..., x_n)$ with respect to x_i for $i = 1, 2, ..., n$:

$$\frac{\partial f}{\partial x_i} = \lim_{\delta x \to 0} \frac{f(x_1, .., x_i + \delta x_i, .., x_n) - f(x_1, ..., x_i, ..., x_n)}{\delta x_i} \tag{1}$$

The gradient of the function ∇f, is the collection of all its partial derivatives into a vector form, e.g., vector valued function [1].

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ ... \\ \frac{\partial f}{\partial x_i} \\ ... \\ \frac{\partial f}{\partial x_n} \end{pmatrix} \tag{2}$$

Suppose standing on a surface of $f(x_1, x_2, ..., x_n)$ at a point $(x_1, x_2, ..., x_n)$, ∇f tells you which direction to travel to increase the value of f most rapidly. Hence, we make the following claim.

Claim 1. *The negative gradient -∇f follows the direction of steepest descent [2].*

III. Gradient Example

Example 1. Given

$$y = f(x_1, x_2) = x_1^2 + x_1 x_2 \tag{3}$$

Find its gradiant as follows [1]:

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 2x_1 - x_2 \\ -x_1 \end{pmatrix} \tag{4}$$

IV. Gredient Steepest Descent for Minimization

Now, let's define f as an error function, and we would like to minimize it. So we can try to change its inputs (x_1, x_2) by iteration steps:

$$(x_1^{k+1}, x_2^{k+1}) = (x_1^k, x_2^k) + (-\eta \nabla f) \tag{5}$$

which will reduce the function value f. To verify this, write f in terms of Taylar expansion as follows

$$f(x_1, x_2) \simeq f(a, b) + \frac{\partial f}{\partial x_1}(x_1 - a) + \frac{\partial f}{\partial x_2}(x_2 - b) \tag{6}$$

use simplified notation for the partial derivative f_{x_1} and f_{x_2}, we have

$$f(x_1, x_2) \simeq f(a, b) + f_{x_1}(a, b) * (x_1 - a) + f_{x_2}(a, b) * (x_2 - b) \tag{7}$$

we want to update (x_1^k, x_2^k) to (x_1^{k+1}, x_2^{k+1}) such that $f(x_1^{k+1}, x_2^{k+1}) < f(x_1^k, x_2^k)$. From equation (6), replace (x_1, x_2) by (x_1^{k+1}, x_2^{k+1}), and let $(x_1^k, x_2^k) = (a, b)$, so we have

$$f(x_1, x_2) - f(a, b) \simeq f_{x_1}(a, b) * (x_1 - a) + f_{x_2}(a, b) * (x_2 - b) \tag{8}$$

Or,

$$f(x_1, x_2) - f(a, b) = (x_1 - a, x_2 - b) \begin{pmatrix} f_{x_1}(a, b) \\ f_{x_2}(a, b) \end{pmatrix} \tag{9}$$

Which can be written as

$$f(x_1, x_2) - f(a, b) = (x_1 - a, x_2 - b) \nabla f \tag{10}$$

Based on the notion in Claim 1, let

$$\Delta x_1 = x_1 - a = -f_{x_1}, \Delta x_2 = x_2 - b = -f_{x_2} \tag{11}$$

hence, we have

$$f(x_1, x_2) - f(a, b) = (\Delta x_1, \Delta x_2) \nabla f = -(f_{x_1}^2 + f_{x_1}^2) \tag{12}$$

So appearently,

$$f(x_1, x_2) - f(a, b) = -(f_{x_1}^2 + f_{x_1}^2) < 0 \tag{13}$$

which shows the error function satisfies

$$f(x_1, x_2) < f(a, b) \tag{14}$$

e.g,

$$f(x_1^{k+1}, x_2^{k+1}) < f(x_1^k, x_2^k). \tag{15}$$

V. Conclusion

In this lecture note, we describe the basic concept of gradient and we have noted that The negative gradient -∇f follows the direction of steepest descent.

Acknowledgment

The author would like to thank Kevin Lee for the discussion on gradient descent algorithm while on their way back from Berkeley, California to his office.

References

[1] Gradient and partial derivatives, https://www.khanacademy.org/math/ multivariable-calculus/ multivariable-derivatives/ partial-derivative-and-gradient- articles/a/the-gradient

[2] Neural Networks, An Introduction, by B. Muller and J. Reinhardt, Springer-Verlag, 1990.

[3] *B.K.P. Horn, Robot Vision*. MIT Press, 1982.