Project "Hessian Matrices and : "
Title
IB3702 Mathematics for Machine Learning

Evertjan Karman John Stegink

15 November, 2025

1 Introduction

The basis of a machine learning algorithm that it tries to predict the right output using a certain input. First the algorithm will have to be trained using correct data. During the training process, the difference between the predicted value and the actual value must be minimized. A cost function is used to quantize the difference, this difference must be minimized. To find the minimum value, the machine learning algorithm iterates until it has found the minimum value. The methods for this iteration are numerous, the most well known algorithm is gradient descent (sections 3.1 and 3.4). For the training to be as effective as possible it is necessary to find the minimum in little iterations. The method for finding the minimum that is discussed in this report is the use of Hessian Matrices (section 3.5). Both gradient descent and the Hessian matrix make use of the Newton method (sections 3.2 and 3.6).

First a description of gradient descent and Newton's will be given for functions using one variable, this is to make the principle clear. Normally for machine learning, 1 variable is not sufficient, the loss function mostly contains multiple variables. The gradient descent, Newton's method and Hessian matrices will be described from the calculus point of view. The linear algebra part will not be discussed (especially eigenvalue and eigenvectors). Reading this report a requires basic understanding of machine learning, especially the proces of machine learning (training and deployment) and the principles of supervised learning.

Using the Hessian Matrix finds the minimum in far less iterations than using gradient descent. The problem of using Hessian Matrices is that calculating a Hessian Matrix is much more complicated and time consuming than using just the derivative as is done using gradient descent.

2 Preliminaries

The notation is similar to the notation of the study guide of IB3702 Mathematics for Machine Learning. The derivatives of multi variable functions are not mentioned. $\frac{\partial^2 f}{\partial x \partial y}$ denotes the partial derivative of function ($f(x, y)$ with regard to y en then to x.

The techniques used and explained in this report are:

- Gradient decent with one variable: $x_{k+1} = x_k - f'(x_k) \cdot \alpha$
- Newton's method with one variable: $x_{k+1} = x_k - (f(x_k)/f'(x_k))$
- Derivates of multi variable functions: $\frac{\partial f}{\partial x} = f_x(x, y) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x, y) - f(x,y)}{\Delta x}$
- Hessian matrix: $\mathbf{H}_f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$

After reading the report, please solve the following problems:
Question 1: Compute the Hessian Matrix for the function $f(x) = x^3 + y^3 + 2xy$ in the point $x = 1, y = 2$.
Answer 1: The Hessian matrix becomes $H = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 6x & 2 \\ 2 & 6y \end{bmatrix}$ The values for $x = 1, y = 2$ are $H = \begin{bmatrix} 12 & 2 \\ 2 & 12 \end{bmatrix}$
Question 2: Compute the Hessian Matrix for the function $f(x) = x^3 + y^3 + 2xy$ in the point $x = 1, y = 2$.
Answer 2: The Hessian matrix becomes $H = [\mathbf{3}$ *Methods*

3.1 Gradient descent with one variable

The purpose of using gradient descent is to find a local or global minimum of a differentiable function by iteratively adjusting the parameters in the direction of the steepest descent. This a mouthful.
It can be clarified by the following metaphor: When you want to find the path from the top to the foot of the mountain in the mist. You walk a few meters down the path with the steepest descent. Then you determine the next path with the steepest descent[1].

[1] https://en.wikipedia.org/wiki/Gradient_descent

Appendix A.1 contains a more detailed version of this metaphor.

Assume that we have a continuous function f defined on R (fig 1).This function:

- is differentiable with derivative $f'(x)$.
- has a starting point x_0.

Then we get x_1 by subtracting $f'(x_0) \cdot \alpha$ from $\mathrm{x}_0, where\alpha$ is called the learning rate. This is equal to walking the path in the steepest descent by α meters in the metaphore. The value of α will be chosen before starting the procedure. Usual values for α are 0.01 or 0.05.

We iterate this, so that we get an **array??** which is recursively defined as:

$$x_{k+1} = x_k - f'(x_k) \cdot \alpha$$

This array will converge to the minimum of f (fig 1). The pitchfalls here are, that the procedure may end in a local minimum, while f has a stronger minimum elsewhere.
Or with a less than optimal choice for the learning rate, the array could even diverge .

3.2 Newton's method with one variable

Newton's method finds the zeroes of a function f, in cases where solving the equation is not possible. It works by iterating through the following steps (inspired by [UT()]:

1. Start at a random point on the curve.
2. Determine the tangent line at that point (using the derivative).
3. Determine the point where the tangent line hits the X-axis
4. Continue with step 2 until the value does not change significantly anymore.

To state it more mathematically: find the value x_0 where the tangent line intersects with the x-axis. That will be x_1. By iterating this procedure we get an array $(x_k)_{k=0,1,\ldots}$. From the geometrical aspect of the procedure, we can give a formula between x_{k+1} and x_k:

$$x_{k+1} = x_k - (f(x_k)/f'(x_k))$$

(that is for finding the zero of f) The idea is that the array (x_k) converges to the value of x where $f(x) = 0$.

The goal is not to find the point where f crosses the x-axis, but to find an minimum for f. This means finding the point where the derivative of f is zero. This can be

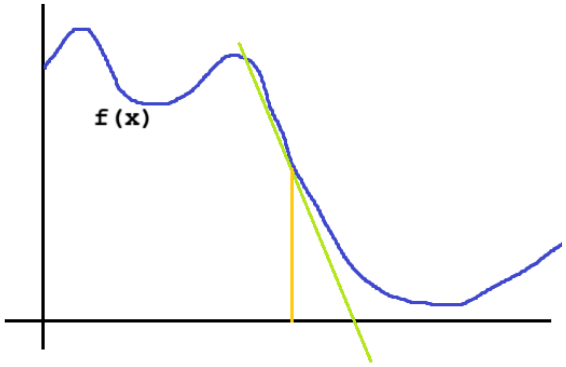

Figure 1: A continuous function f defined on R

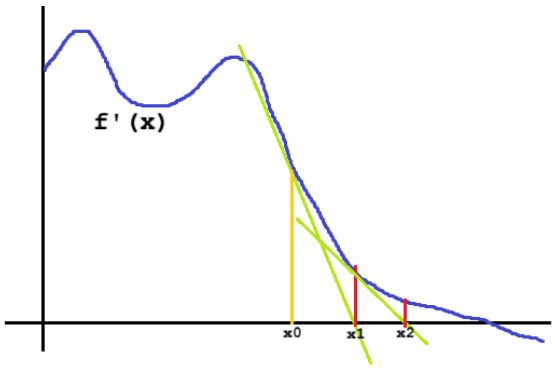

Figure 2: Newton's method

accomplished with Newton's method by substituting f' for f and f'' for f'.

$$x_{k+1} = x_k - (f'(x_k)/f''(x_k))$$

And example is depicted in (fig 2).

In order to know if $f'(x)$ points to a minimum of f, we need to look at the second derivative $f''(x)$:
$f''(x) > 0 \Rightarrow$ f has a minimum at x
$f''(x) < 0 \Rightarrow$ f has a maximum at x
$f''(x) = 0 \Rightarrow$ inconclusive, perhaps an inflection point

3.3 Functions with two or more variables and their derivatives

A function with one variable describes the result of a calculation with respect to just 1 variable. In practice the result of a function is dependent on more that one variable. For example the price of a house is not only dependent on the floor area of the house, but for example the distance to the nearest shops count, the number of crimes in the area per year etc. This means the function has multiple variables. This is what we see in machine learning most of the time too. An example of a graph of the multi value function can be seen in figure 4. This depicts the function $f(x, y) = 85 - \frac{1}{90}x^2(x-6)y^2(y-6)$. As you can see the function is defined by two variables: $f(x, y)$.

Finding the slope in a point using a function with one variable is easily done by determining the derivative. For multi value functions this is done in only 1 direction at a time. This means the derivatie has te determined with regard to 1 variable the rest of the variables is considered as a constant. [cal(2022)] describes this very well.

Mathematically:

$$\frac{\partial f}{\partial x} = f_x(x,y) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x, y) - f(x,y)}{\Delta x}$$

$$\frac{\partial f}{\partial y} = f_y(x,y) = \lim_{\Delta y \to 0} \frac{f(x, y+\Delta y) - f(x,y)}{\Delta y}$$

3.4 Gradient descent with two or more variables

Considering the metaphore in section 3.1 the x is the number of steps to walk to the west and y is the number of meters to walk to the north, instead of just walking straight ahead.

With a function $f(x,y)$ of more variables, we can determine the gradient:

$$\nabla f(x,y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

The method is the same, but where we took the derivative for one variable, we will now take the gradient, and the recursive definition of our array (x_k, y_k) becomes:

$$(x_{k+1}, y_{k+1}) = (x_k, y_k) - \nabla f(x,y) \cdot \alpha$$

3.5 Hessian matrix

The second derivative of a function tells something about the curvature of a function in a certain point. A function is convex at a point when the second derivative is positive, and concave if it is negative. When the value of the second derivative is zero the point is an inflection point, a point at which te curvature changes sign. This is possibly a minimum or maximum of the function.

A Hessian matrix is a squared matrix containing all second derivatives of a multi valued function. Say we have function $f(x,y)$ of 2 variables. Then we get a square matrix having te following format:

$$H_f(x,y) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

Generally a Hessian Matrix has the form: $\mathbf{H}_f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$

The Hessian matrix is a symmetric matrix, because of Clairauts Theorem. This theorem states that $\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial^2 f}{\partial y \partial x}$.

With Hessian matrix local extremes of a function can be found. It helps to identify saddle points, local minima and local maxima. A saddle point (figure 3) is a function that does neither contain a local maximum or local minimum. From this point a function in some directions (using a certain set of variables) is a maximum and in some points a minimum. The type of extreme can be found by calculating the eigenvalues of the Hessian matrix (section 3.6).

The lineair algebra part of Hessian matrices is not in the scope of this report. If you want to to have a thourough explanation, [Nocedal and Wright(2006)] is a good reference. Hessian matrices can be become quite large. The matrix for a function with n different variables the matrix will have the size of $n \times n$ (in modern machine learning models the value of n can be several billions). The size of the matrix to be calculated and stored in memory will have a size of order n^2. For such situations approximations of the Hessian Matrix are being used. A often used algorithm that uses an approximation is BFGS.

3.6 Newton's method with two or more variables

The main principles of Newton's method with one variable apply to Newton's method with two or more variables. Say we have function $f(x, y)$ (containing 2 variables).

Then here we have it's Hessian matrix:

$$H_f(x, y) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

Now at a given point (x,y) we'll calculate it's eigenvalues $\lambda_1, \lambda_2, \ldots$
(A 2 by 2 matrix would have at most two eigenvalues)

If the gradient has value (0,0) at point (x,y) then:

- If all the eigenvalues of H_f at (x,y) are positive, it's a minimum
- If all the eigenvalues of H_f at (x,y) are negative, it's a maximum
- If one of the eigenvalues of H_f at (x,y) are zero, it's a saddle point
- In other cases, it's inconclusive

In Newton's method generalized to more than one variables, the formula for the next point is:

$$(x_{k+1}, y_{k+1}) = (x_k, y_k) - (H_f^{-1}(x_k, y_k) \cdot \nabla f(x_k, y_k))$$

3.7 Example of a function of two variables

We will look at this function (fig 4):

$$f(x, y) = 85 - \frac{1}{90}x^2(x-6)y^2(y-6)$$

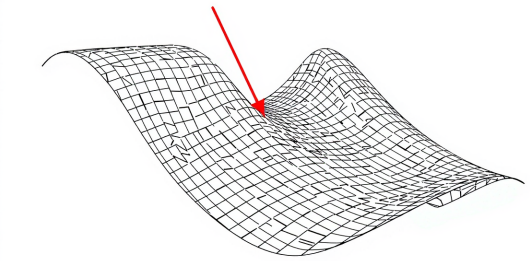

Figure 3: A saddle point

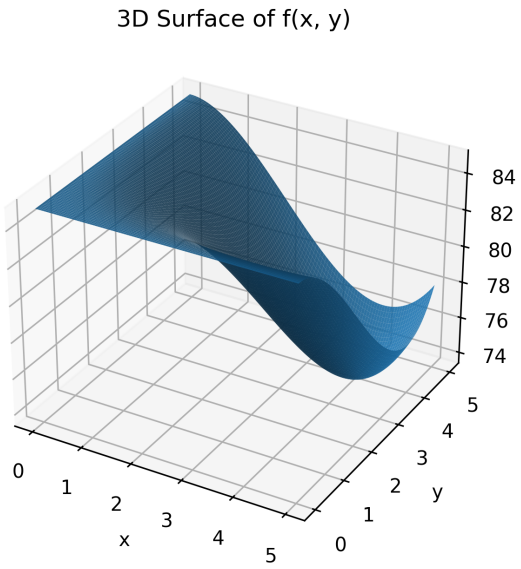

Figure 4: 3D surface of $f(x, y)$

Visually we see a possible minimum near point $(x, y) = (4, 4)$.

4 Numerical Examples

Text...

5 Collaboration

Text...

6 Reflection

6.1 Student a

Text...

6.2 Student b

Text...

References

[cal(2022)] Partial derivative fully explained w/ step-by-step examples, 1 2022. URL https://calcworkshop.com/partial-derivatives/partial-derivative/.

[Nocedal and Wright(2006)] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 2006.

[UT()] University of Texas UT. The idea of newton's method. URL https://web.ma.utexas.edu/users/m408n/CurrentWeb/LM0-0-1.php.

[Wikipedia(2025)] Wikipedia. Gradient descent, 11 2025. URL https://en.wikipedia.org/wiki/Gradient descent.

A Metaphores

A.1 Gradient descent

The basic intuition behind gradient descent can be illustrated by a hypothetical scenario. People are stuck in the mountains and are trying to get down (i.e., trying to find the global minimum). There is heavy fog such that visibility is extremely low. Therefore, the path down the mountain is not visible, so they must use local information to find the minimum. They can use the method of gradient descent, which involves looking at the steepness of the hill at their current position, then proceeding in the direction with the steepest descent (i.e., downhill). If they were trying to find the top of the mountain (i.e., the maximum), then they would proceed in the direction of steepest ascent (i.e., uphill).

Using this method, they would eventually find their way down the mountain or possibly get stuck in some hole (i.e., local minimum or saddle point), like a mountain lake. However, assume also that the steepness of the hill is not immediately obvious with simple observation, but rather it requires a sophisticated instrument to measure, which the people happen to have at that moment. It takes quite some time to measure the steepness of the hill with the instrument. Thus, they should minimize their use of the instrument if they want to get down the mountain before sunset. The difficulty then is choosing the frequency at which they should measure the steepness of the hill so as not to go off track.

In this analogy, the people represent the algorithm, and the path taken down the mountain represents the sequence of parameter settings that the algorithm will explore. The steepness of the hill represents the slope of the function at that point. The instrument used to measure steepness is differentiation. The direction they choose to travel in aligns with the gradient of the function at that point. The amount of time they travel before taking another measurement is the step size.

Copied from [Wikipedia(2025)].