Optimization Concepts in Artificial Intelligence

Aymen Negadi

1 Optimization Concepts

1.1 Gradient

1.1.1 Definition of the Gradient

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a scalar-valued function of several variables. The **gradient** of f is the vector composed of all its first-order partial derivatives.

It is defined as:

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

The gradient indicates the direction of the steepest increase of the function, and its magnitude represents the rate of this increase.

1.1.2 Why the Gradient Is Important in Artificial Intelligence

In Artificial Intelligence and Machine Learning, models are trained by minimizing a scalar loss function $L(w_1, w_2, \ldots, w_n)$.

The gradient plays a central role because it allows:

- Measuring how the loss function changes with respect to each parameter
- Updating model parameters during training
- Implementing optimization algorithms such as Gradient Descent

Thus, the gradient provides the necessary information to guide the learning process toward optimal solutions.

1.1.3 Numerical Example

Consider the following scalar function:

$$f(x, y) = x^2 + y^2$$

The partial derivatives are:

$$\frac{\partial f}{\partial x} = 2x \quad \text{and} \quad \frac{\partial f}{\partial y} = 2y$$

Therefore, the gradient of f is:

$$\nabla f(x, y) = \begin{pmatrix} 2x \\ 2y \end{pmatrix}$$

At the point $(x, y) = (1, 2)$, we obtain:

$$\nabla f(1, 2) = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

This result shows that the function increases more rapidly in the direction of y than in the direction of x at this point.

1.1.4 Why We Need the Jacobian and the Hessian

The gradient is defined only for **scalar-valued functions**. However, in more advanced situations, this is not sufficient.

- When dealing with **vector-valued functions** $F : \mathbb{R}^n \to \mathbb{R}^m$, the gradient is replaced by the **Jacobian matrix**, which contains all first-order partial derivatives.
- When second-order information is required for faster convergence or curvature analysis, the **Hessian matrix** is used. It contains all second-order partial derivatives of a scalar function.

In Machine Learning, Jacobians and Hessians are essential for advanced optimization techniques, sensitivity analysis, and understanding the behavior of complex models.

1.1.5 Example Where the Gradient Cannot Be Used

Consider the following function:

$$F(x, y) = \begin{pmatrix} x^2 + y \\ xy \end{pmatrix}$$

This function is a **vector-valued function**:

$$F : \mathbb{R}^2 \to \mathbb{R}^2$$

Since the gradient is defined only for **scalar-valued functions** $f : \mathbb{R}^n \to \mathbb{R}$, it cannot be applied to $F(x, y)$.

In this case, the appropriate tool is the **Jacobian matrix**, which contains all first-order partial derivatives of the vector function.

The Jacobian matrix of F is given by:

$$J_F(x, y) = \begin{pmatrix} \frac{\partial(x^2+y)}{\partial x} & \frac{\partial(x^2+y)}{\partial y} \\ \frac{\partial(xy)}{\partial x} & \frac{\partial(xy)}{\partial y} \end{pmatrix}$$

After computation, we obtain:

$$J_F(x, y) = \begin{pmatrix} 2x & 1 \\ y & x \end{pmatrix}$$

This example illustrates that when a function has multiple outputs, the gradient is no longer sufficient, and the Jacobian matrix must be used instead.

1.2 Hessian Matrix

1.2.1 Definition of the Hessian

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a twice differentiable scalar-valued function. The **Hessian matrix** of f is the square matrix containing all second-order partial derivatives.

It is defined as:

$$H_f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

The Hessian matrix captures the **curvature** of the function.

1.2.2 Why the Hessian Is Important in Artificial Intelligence

While the gradient provides first-order information (direction of steepest change), the Hessian provides second-order information about the function.

In Artificial Intelligence and Machine Learning, the Hessian is useful for:

- Analyzing the curvature of the loss function
- Determining whether a critical point is a minimum, maximum, or saddle point
- Accelerating optimization algorithms such as Newton's method

Second-order methods can converge faster than first-order methods, especially near the optimum.

1.2.3 Numerical Example

Consider the function:

$$f(x, y) = x^2 + y^2$$

The first-order partial derivatives are:

$$\frac{\partial f}{\partial x} = 2x \quad \text{and} \quad \frac{\partial f}{\partial y} = 2y$$

The second-order partial derivatives are:

$$\frac{\partial^2 f}{\partial x^2} = 2, \quad \frac{\partial^2 f}{\partial y^2} = 2, \quad \frac{\partial^2 f}{\partial x \partial y} = 0$$

Thus, the Hessian matrix is:

$$H_f = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

This Hessian matrix is positive definite, which indicates that the function has a minimum at the critical point $(0, 0)$.

1.2.4 Gradient vs Hessian

- The **gradient** provides first-order information and indicates a direction.
- The **Hessian** provides second-order information and describes curvature.

In practice, Gradient Descent is preferred for large-scale problems, while Hessian-based methods are used when higher precision and faster convergence are required.

1.3 Jacobian Matrix

1.3.1 Definition of the Jacobian

Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a vector-valued function defined as:

$$F(x_1, x_2, \ldots, x_n) = \begin{pmatrix} f_1(x_1, x_2, \ldots, x_n) \\ f_2(x_1, x_2, \ldots, x_n) \\ \vdots \\ f_m(x_1, x_2, \ldots, x_n) \end{pmatrix}$$

The **Jacobian matrix** of F is the matrix containing all first-order partial derivatives of each component function with respect to each variable.

It is defined as:

$$J_F = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}$$

1.3.2 Why the Jacobian Is Important in Artificial Intelligence

In Artificial Intelligence and Machine Learning, many models involve **vector-valued functions**, especially in neural networks where layers map vectors to vectors.

The Jacobian matrix is essential because it allows:

- Measuring how each output varies with respect to each input
- Propagating gradients through vector-valued transformations
- Applying the Chain Rule in multivariable and multilayer models

The Jacobian plays a central role in the Backpropagation algorithm.

1.3.3 Numerical Example

Consider the vector-valued function:

$$F(x,y) = \begin{pmatrix} x^2 + y \\ xy \end{pmatrix}$$

The partial derivatives are:

$$\frac{\partial(x^2+y)}{\partial x} = 2x, \quad \frac{\partial(x^2+y)}{\partial y} = 1$$

$$\frac{\partial(xy)}{\partial x} = y, \quad \frac{\partial(xy)}{\partial y} = x$$

Thus, the Jacobian matrix is:

$$J_F(x,y) = \begin{pmatrix} 2x & 1 \\ y & x \end{pmatrix}$$

1.3.4 Gradient, Jacobian, and Hessian: Summary

- The **Gradient** is used for scalar-valued functions and produces a vector.
- The **Jacobian** is used for vector-valued functions and produces a matrix of first-order derivatives.
- The **Hessian** is used for scalar-valued functions and produces a matrix of second-order derivatives.

Together, these tools form the mathematical foundation of optimization and learning algorithms in Artificial Intelligence.

1.4 Comparison Between Gradient, Jacobian, and Hessian

Concept	**Type de fonction**	**Résultat**	**Exemple simple**
Gradient	$f : \mathbb{R}^n \to \mathbb{R}$	Vecteur	$f(x,y) = x^2 + y^2$
Jacobian	$F : \mathbb{R}^n \to \mathbb{R}^m$	Matrice	$F(x,y) = \begin{pmatrix} x^2 + y \\ xy \end{pmatrix}$
Hessian	$f : \mathbb{R}^n \to \mathbb{R}$	Matrice	$f(x,y) = x^2 - y^2$

1.4.1 Associated Derivatives for Each Concept

- **Gradient example**:

$$\nabla f(x, y) = \begin{pmatrix} 2x \\ 2y \end{pmatrix}$$

- **Jacobian example**:

$$J_F(x, y) = \begin{pmatrix} 2x & 1 \\ y & x \end{pmatrix}$$

- **Hessian example**:

$$H_f(x, y) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$

1.4.2 Interpretation in Artificial Intelligence

- The **gradient** is used to update model parameters during training.
- The **Jacobian** is used to propagate derivatives through vector-valued transformations.
- The **Hessian** provides curvature information and is useful for advanced optimization methods.