


Introduction to Line Search

Let's begin with the first question: "Given a Hessian, how do we decide in which part of the domain it is positive definite?" This question seems to stem from a previous example, where a function resembled an upward opening bowl, but at a specific point, the Hessian was not positive definite. The short answer is that there's no easy way to determine the positive definiteness of the Hessian across the domain. You would either need to come up with an analytical guarantee or numerically check at each point where you're solving the problem. This difficulty is one reason second-order methods are often avoided in high-dimensional problems.

Now, moving on to a related question: "Is the direction p arbitrary, or is it specifically p_k ?" There's been some confusion here, and I take partial responsibility for that. To clarify, when we are at x_k and moving towards x_{k+1} , the direction we choose is called p_k . At this point, we have many descent directions available, but in Newton's method, we find the best direction by minimizing a specific optimization problem. The p_k is the result of that minimization. Initially, it's just p , and after solving the problem, the best direction is referred to as p_k .

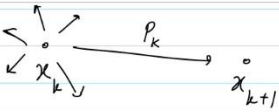


Line Search Methods.


$p?$
 \rightarrow

$$\nabla_p f(x_k + \epsilon p) = 0$$

$p_k = \operatorname{argmin}_p \|\nabla_p f(x_k + \epsilon p)\|$



OPTIMIZATION THEORY AND ALGORITHMS



Another question that came up was: "Why do we assume the Hessian is symmetric?" This assumption arises because of a theorem that ensures the symmetry of mixed partial derivatives, assuming these derivatives are continuous. So, yes, we do need to check that this condition holds.

There was also a question about the requirement for an open neighborhood around x_k when performing gradient descent. The idea behind an open neighborhood is to ensure that there are non-zero points around x_k where gradient descent can actually occur.

NPTEL

$$\nabla f_k^T p_k < 0$$

learning rate

Define: $\phi(\alpha) = f(x_k + \alpha p_k)$, $\alpha > 0$, $\because p_k$ is a descent dir

What is the desired quality of α ?

$$\phi'(\alpha) = 0$$

0 α

3/3

Next, we discuss the Newton direction. The expression for p_n is

$$p_n = -H^{-1} \nabla f$$

where H is the Hessian and ∇f is the gradient of the function. As long as the Hessian is invertible, the Newton direction is unique. If the Hessian is not invertible, the expression becomes undefined.

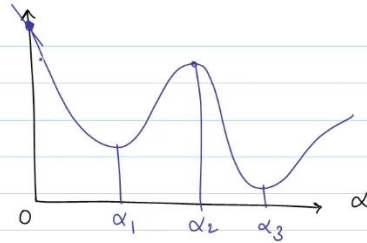


What is the desired quality of α ? descent dir

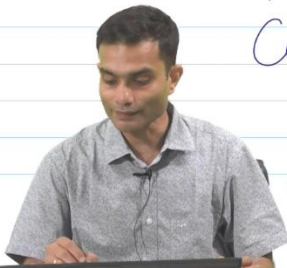
$$\phi'(\alpha) = 0$$

Starting at $\alpha = 0$

$\phi(0)$ & $\phi'(0)$



Computing $\phi(\alpha)$ and $\phi'(\alpha)$ is EXPENSIVE.



OPTIMIZATION THEORY AND ALGORITHMS

Now, moving on to the line search method. The basic idea of line search is to move from x_k to x_{k+1} by walking some distance in the direction p_k . The step length α is what needs to be determined. People in machine learning might call this the "learning rate." The goal is to find an α such that $f(x_{k+1}) < f(x_k)$, assuming we're moving in a descent direction.

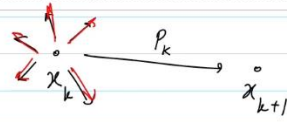
When we look at the function $\phi(\alpha)$, which is a scalar function of α , the goal is to find the α that minimizes this function. We want to find the α such that the gradient of $\phi(\alpha)$ is zero, indicating a stationary point.



Line Search Methods.

$p?$
→

$$\nabla_p f(x_k + \epsilon p) = 0$$



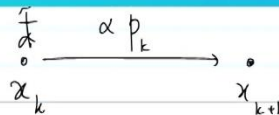
$$p_k = \operatorname{argmin}_p \|\nabla_p f(x_k + \epsilon p)\|$$

$$\left. \begin{aligned} \frac{\partial^2 f}{\partial x_1 \partial x_2} &= \frac{\partial^2 f}{\partial x_2 \partial x_1} \end{aligned} \right\}$$

$$p_N = -(\nabla^2 f)^{-1} \nabla f$$



Graphically, this might look like a curve with several points where the gradient vanishes, but in practice, computing $\phi(\alpha)$ and its derivative can be expensive, especially in real-life problems like antenna design, where each evaluation might take hours. Therefore, finding the exact value of α is often impractical, leading to the need for inexact line search methods.



$$x_{k+1} = x_k + \underbrace{(\alpha p_k)}_{\text{fixed}}$$

If we take small values of α

$$\nabla f_k^T p_k < 0$$

Step length \rightarrow TBD
"learning rate"

Define: $\phi(\alpha) = f(x_k + \alpha p_k)$, $\alpha > 0$, p_k is a descent dir



Lastly, the "Wolfe conditions" provide a set of rules to guide the selection of α . These conditions help ensure that the step length is not too large or too small, allowing for a more efficient search for the optimal α . Although the conditions are named after Wolfe, they are relatively simple and could have been discovered by anyone interested in this problem decades ago.

In summary, line search methods aim to balance computational efficiency and precision, especially in the context of large, complex problems where exact solutions are impractical.