Practical Optimization Algorithms and Applications

Chapter II: Fundamentals of Unconstrained Optimization

Lingfeng NIU

Research Center on Fictitious Economy & Data Science,
University of Chinese Academy of Sciences

niulf@ucas.ac.cn

What is a solution for unconstrained optimization?

In unconstrained optimization, we minimize an objective function that depends on real variables, with no restriction at all on the values of these variables.

The mathematical formulation for unconstrained optimization is

$$\min_{x} f(x) \tag{1}$$

where

$x \in \Re^n$ is a real vector with $n \geq 1$ components;
$f : \Re^n \to \Re$ is a continuous function.

Solution Definition

- A point x^* is a **global minimizer** if $f(x^*) \leq f(x)$ for all x;
- A point x^* is a **(weak) local minimizer** if there is a neighborhood $\mathcal{N}$ of x^* such that $f(x^*) \leq f(x)$ for all $x \in \mathcal{N}$;
- A point x^* is a **strict local minimizer** (also called a *strong local minimizer*) if there is a neighborhood $\mathcal{N}$ of x^* such that $f(x^*) < f(x)$ for all $x \in \mathcal{N}$ with $x \neq x^*$;
- A point x^* is an **isolated local minimizer** if there is a neighborhood $\mathcal{N}$ of x^* such that x^* is the only local minimizer in $\mathcal{N}$;

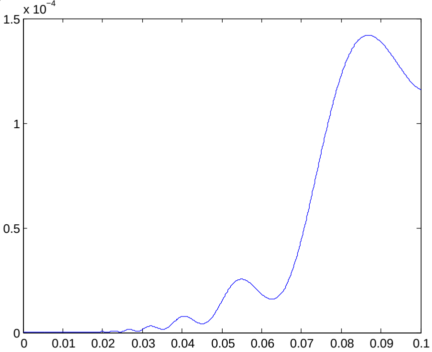
A Counter Example

All isolated local minimizer are strict. But strict minimizer are not always isolated.

For example, for function

$$f(x) = x^4 cos(\frac{1}{x}) + 2x^4, \qquad f(0) = 0.$$

$x = 0$ is a strict local minimizer. However, there are strict local minimizer at many nearby points x_j, and we can label these points so that $x_j \to 0$ as $j \to \infty$.

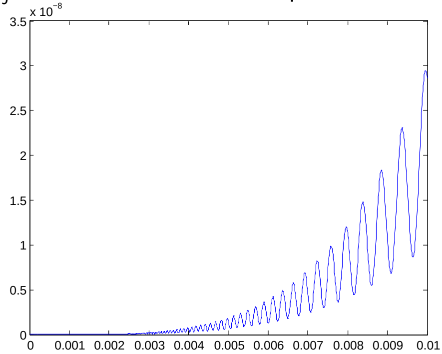


A Counter Example

All isolated local minimizer are strict. But strict minimizer are not always isolated.
For example, for function

$$f(x) = x^4 cos(\frac{1}{x}) + 2x^4, \qquad f(0) = 0.$$

$x = 0$ is a strict local minimizer. However, there are strict local minimizer at many nearby points x_j, and we can label these points so that $x_j \to 0$ as $j \to \infty$.



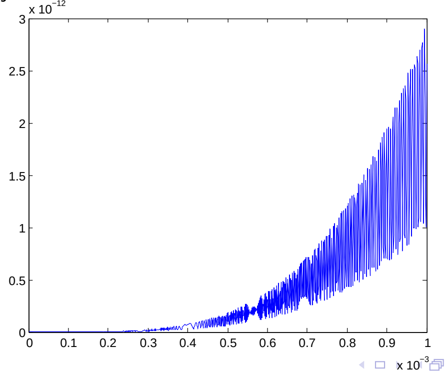
A Counter Example

All isolated local minimizer are strict. But strict minimizer are not always isolated.

For example, for function

$$f(x) = x^4 cos(\frac{1}{x}) + 2x^4, \qquad f(0) = 0.$$

$x = 0$ is a strict local minimizer. However, there are strict local minimizer at many nearby points x_j, and we can label these points so that $x_j \to 0$ as $j \to \infty$.



Recognizing a Local Minimum

From the definition, it might seem that the only way to find out whether a point x^* is a local minimum is to examine all the points in its immediate vicinity, to make sure that none of them has a smaller function value.

- When the function f is *smooth*(i.e continuously differentiable,), however, there are more efficient and practical ways to identify local minima.
  - In particular, if f is twice continuously differentiable, we may be able to tell that x^* is a local minimizer (and possibly a strict local minimizer) by examining just the gradient $\nabla f(x^*)$ and the Hessian $\nabla^2 f(x^*)$.

The mathematical tool used to study minimizers of smooth functions is Taylor's theorem.

Recognizing a Local Minimum

Theorem (Taylor's Theorem)

Suppose that $f : \Re^n \to \Re$ is continuously differentiable and that $p \in \Re^n$. Then we have that

$$f(x+p) = f(x) + \nabla f(x+tp)^T p, \tag{2}$$

for some $t \in (0,1)$. Moreover, if f is twice continuously differentiable, we have that

$$\nabla f(x+p) = \nabla f(x) + \int_0^1 \nabla^2 f(x+tp) p dt, \tag{3}$$

and that

$$f(x+p) = f(x) + \nabla f(x)^T p + \frac{1}{2} p^T \nabla^2 f(x+tp) p, \tag{4}$$

for some $t \in (0,1)$.

Recognizing a Local Minimum - Necessary Conditions

Theorem (First-Order Necessary Conditions)

If x^* is a local minimizer and f is continuously differentiable in an open neighborhood of x^*, then $\nabla f(x^*) = 0$.

Recognizing a Local Minimum - Necessary Conditions

Theorem (First-Order Necessary Conditions)

If x^* is a local minimizer and f is continuously differentiable in an open neighborhood of x^*, then $\nabla f(x^*) = 0$.

We call x^* a **stationary point** if $\nabla f(x^*) = 0$. According to the above theorem, any local minimizer must be a stationary point.

Recognizing a Local Minimum - Necessary Conditions

Theorem (First-Order Necessary Conditions)

If x^* is a local minimizer and f is continuously differentiable in an open neighborhood of x^*, then $\nabla f(x^*) = 0$.

We call x^* a **stationary point** if $\nabla f(x^*) = 0$. According to the above theorem, any local minimizer must be a stationary point.

Theorem (Second-Order Necessary Conditions)

If x^* is a local minimizer and f and $\nabla^2 f(x)$ exits and is continuous in an open neighborhood of x^*, then $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive semidefinite.

Recognizing a Local Minimum - Sufficient Conditions

We now describe *sufficient conditions*, which are conditions on the derivatives of f at the point x^* that guarantee that x^* is a local minimizer.

Recognizing a Local Minimum - Sufficient Conditions

We now describe *sufficient conditions*, which are conditions on the derivatives of f at the point x^* that guarantee that x^* is a local minimizer.

Theorem (Second-Order Sufficient Conditions)

Suppose that $\nabla^2 f(x)$ is continuous in an open neighborhood of x^* and that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite. Then x^* is a local minimizer of f.

Recognizing a Local Minimum - Sufficient Conditions

We now describe *sufficient conditions*, which are conditions on the derivatives of f at the point x^* that guarantee that x^* is a local minimizer.

Theorem (Second-Order Sufficient Conditions)

Suppose that $\nabla^2 f(x)$ is continuous in an open neighborhood of x^* and that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite. Then x^* is a local minimizer of f.

Note that the second-order sufficient conditions are not necessary: A point x^* may be a local minimizer, and yet may fail to satisfy the sufficient conditions.

Recognizing a Local Minimum - Sufficient Conditions

We now describe *sufficient conditions*, which are conditions on the derivatives of f at the point x^* that guarantee that x^* is a local minimizer.

Theorem (Second-Order Sufficient Conditions)

Suppose that $\nabla^2 f(x)$ is continuous in an open neighborhood of x^* and that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite. Then x^* is a local minimizer of f.

Note that the second-order sufficient conditions are not necessary: A point x^* may be a local minimizer, and yet may fail to satisfy the sufficient conditions.

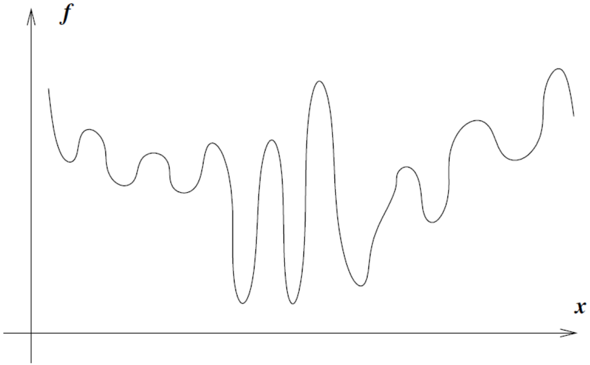
A simple example

Consider the problem

$$\min_x f(x) = x^4,$$

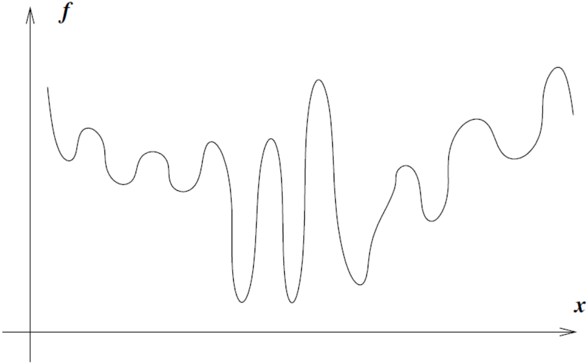
for which the point $x^* = 0$ is a strict minimizer at which the Hessian matrix vanishes (and is therefore not positive definite).

Global Minimizer vs Local Minimizer



This figure illustrates a function with many local minimizer. It's usually difficult to find the global minimizer for such functions, because algorithms tend to be "trapped" at local minimizers.

Global Minimizer vs Local Minimizer



This figure illustrates a function with many local minimizer. It's usually difficult to find the global minimizer for such functions, because algorithms tend to be "trapped" at local minimizers.

Sometimes we have additional "global" knowledge about f that may help in identifying global minima. An important special case is that of convex functions, for which every local minimizer is also a global minimizer.

The term "convex" can be applied both to sets and to functions.

- A set $S \in \Re^n$ is a *convex set* if for any two points $x \in S$ and $y \in S$, we have $\alpha x + (1-\alpha)y \in S$ for all $\alpha \in [0,1]$.
- The function f is a *convex function* if its domain S is a convex set and if for any two points x and y in S, the following property if satisfied:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y) \qquad \forall \alpha \in [0,1]. \tag{5}$$

We say that f is *strictly convex* if the inequality in (5) is strict whenever $x \neq y$ and α is in the open interval $(0,1)$. A function f is said to be *concave* if $-f$ is convex.

Convex programming

The term *convex programming* is used to describe a special case of the general constrained optimization problem in which

- the objective function is convex,
- the equality constraint functions $c_i(\cdot)$, $i \in \mathcal{E}$, are linear, and
- the inequality constraint functions $c_i(\cdot)$, $i \in \mathcal{I}$, are concave.

Practice

Prove

$$f(x) = \frac{1}{2}x^T x - b^T x,$$

is a convex function.

Practice

Prove

$$f(x) = \frac{1}{2}x^T x - b^T x,$$

is a convex function.

Answer:

(i) For any $x, y \in \Re^n$, $\lambda \in [0, 1]$, we have that

$$\begin{aligned} & \lambda f(x) + (1-\lambda)f(y) - f(\lambda x + (1-\lambda)y) \\ = \quad & \frac{1}{2}\lambda(1-\lambda)(x-y)^T(x-y) \geq 0. \end{aligned}$$

Therefore, $f(\cdot)$ is a convex function.

Recognizing a Local Minimum - Convex Functions

When the objective function is convex, global minimizers are simple to characterize.

Theorem

When f is convex, any local minimizer x^* is a global minimizer of f. If in addition f is differential, then any stationary point x^* is a global minimizer of f.

Recognizing a Local Minimum - Convex Functions

When the objective function is convex, global minimizers are simple to characterize.

Theorem

When f is convex, any local minimizer x^* is a global minimizer of f. If in addition f is differential, then any stationary point x^* is a global minimizer of f.

These results provide the foundations for unconstrained optimization algorithms: all algorithms seek a point where $\nabla f(\cdot)$ vanishes in one way or another.

Algorithm

An algorithm is a procedure or formula for solving a problem.

- The word derives from the name of the mathematician, Mohammed ibn-Musa al-Khwarizmi, who was part of the royal court in Baghdad and who lived from about 780 to 850.
- In mathematics and computer science, an algorithm is a step-by-step procedure for calculations. More precisely, it is an effective method expressed as a finite list of well-defined instructions for calculating a function.

Convergence

An iterative method is called convergent if the corresponding sequence converges for given initial approximations.

Algorithm

Good algorithms should possess the following properties:

- **Robustness**. They should perform well on a wide variety of problems in their class, for all reasonable values of the starting point.
- **Efficiency**. They should not require excessive computer time or storage.
- **Accuracy**. They should be able identify a solution with precision, without being overly sensitive to errors in the data or to the arithmetic rounding errors that occur when the algorithm is implemented on a computer.

These goals may conflict. Tradeoff between convergence rate and storage requirements, and between robustness and speed, and so on, are always central issues.

Overview of Optimization Algorithms

Generally speaking, optimization algorithms are iterative.

- Firstly, begin with an initial guess of the variable;
- Then, generate a sequence of improved estimates (called "iterates");
- At last, terminate
  - when either no more progress can be made, or
  - when it seems that a solution point has been approximated with sufficient accuracy.

The strategy used move from one iterate to the next distinguishes one algorithm from another.

Algorithms for Unconstrained Optimization

All algorithms for unconstrained minimization require the user to supply a starting point, which we usually denote by x_0.

- The user with knowledge about the application and the data set may be in a good position to choose x_0 to be a reasonable estimate of the solution.
- Otherwise, the starting point must be chosen by the algorithm, either by a systematic approach or in some arbitrary manner.

Algorithms for Unconstrained Optimization

All algorithms for unconstrained minimization require the user to supply a starting point, which we usually denote by x_0.

- The user with knowledge about the application and the data set may be in a good position to choose x_0 to be a reasonable estimate of the solution.
- Otherwise, the starting point must be chosen by the algorithm, either by a systematic approach or in some arbitrary manner.

Begin at x_0, a sequence of iterates $\{x_k\}_{k=0}^{\infty}$ are generated. In deciding how to move from one iterate x_k to the next, the algorithms use information about the function f at x_k, and possibly also information from earlier iterates.

- They use this information to find a new iterate x_{k+1} with a lower function value than x_k;
- There exist *nonmonotone* algorithms that do not insist on a decrease in f at every step, but even these algorithms require f to be decreased after some prescribed number m of iterations, that is, $f(x_k) < f(x_{k-m})$.

There are two strategies for moving from the current point x_k to a new iterate x_{k+1}: *Line Search* and *Trust Region*.

Thanks for your attention!