ЛАБОРАТОРНИЙ ПРАКТИКУМ
АДАПТИВНІ СИСТЕМИ ОПРАЦЮВАННЯ ІНФОРМАЦІЇ

Лабораторна робота № 3

Дослідження градієнтних методів мінімізації

Мета роботи: ознайомлення з градієнтними методами багатомірної мінімізації цільових функцій, реалізація алгоритму і програми для цих методів, тестування програми на прикладі тестових функцій.

1. Теоретична частина

Чисельні методи багатомірної оптимізації. Задача багатомірної безумовної оптимізації формулюється у вигляді:

$$\min_{x \in X} f(x), \tag{3.1}$$

де $x = \{x_1, x_2, ..., x_n\}$ - точка в n - мірному просторі $x \in R^n$, тобто цільова функція $f(x) = f(x_1, x_2, ..., x_n)$ - функція n аргументів.

Чисельні методи пошуку мінімуму, як правило, полягають в побудові послідовності точок $\left\{x^{(k)}\right\}$, що задовольняють умові $f\left(x^{(0)}\right) > f\left(x^{(1)}\right) > ... > f\left(x^{(m)}\right) > ...$. Методи побудови таких послідовностей називаються методами спуску. У цих методах точки послідовності $\left\{x^{(k)}\right\}$ обчислюються за формулою:

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} p^{(k)},\ k = 0, 1, 2, ... \tag{3.2}$$

де $p^{(k)}$ - напрям спуску, $\alpha^{(k)}$ - довжина кроку в цьому напрямку.

Різні методи спуску відрізняються один від одного способами вибору напряму спуска $p^{(k)}$ і довжини кроку $\alpha^{(k)}$ уздовж цього напряму. Алгоритми безумовної мінімізації прийнято ділити на класи, залежно від максимального порядку похідних цільової функції. Так, методи, що використовують тільки значення самої цільової функції, відносять до методів нульового порядку (іноді їх називають також методами прямого пошуку). Якщо, крім того, потрібно обчислення перших похідних функції, що мінімізується, то ми маємо справу з методами першого порядку; якщо ж додатково використовуються другі похідні, то це методи другого порядку і т. д.

Градієнтні методи. Загальна схема градієнтного спуску.

Як відомо, градієнт функції в деякій точці $x^{(k)}$ спрямований убік найшвидшого локального зростання функції і перпендикулярний лінії рівня (поверхня постійного значення функції $f(x)$, що проходить через точку $x^{(k)}$).

Вектор, протилежний градієнту, називається антиградієнтом і він спрямований у бік найшвидшого спадання функції $f(x)$. Вибираючи як напрям спуску $p^{(k)}$ антиградієнт у точці $x^{(k)}$, отримуємо ітераційний процес виду:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} f'\left(x^{(k)}\right),\ \alpha^{(k)} > 0, k = 0,1,2,... \qquad (3.3)$$

Вираз (3.3) можна представити у вигляді:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \nabla f\left(x^{(k)}\right),\ \alpha^{(k)} > 0, k = 0,1,2,... \ . \qquad (3.4)$$

Всі ітераційні процеси, в яких напрям руху на кожному кроці збігається з антиградієнтом функції, називаються градієнтними методами. Вони відрізняються один від одного тільки способом вибору кроку $\alpha^{(k)}$. Існує багато різних способів вибору $\alpha^{(k)}$, але найбільш поширені серед них: метод з постійним кроком, метод із діленням кроку і метод найшвидшого спуску.

Градієнтний метод з постійним кроком. Основна проблема градієнтних методів - це вибір кроку $\alpha^{(k)}$. Досить малий крок $\alpha^{(k)}$ забезпечує спадання функції, тобто виконання нерівності:

$$f\left(x^{(k)} - \alpha^{(k)} f'\left(x^{(k)}\right)\right) < f\left(x^{(k)}\right) \qquad (3.5)$$

але може привести до неприйнятно великої кількості ітерацій, необхідних для досягнення точки мінімуму. З іншого боку, занадто великий крок може викликати несподіване зростання функції (невиконання умови спадання) або привести до коливань близько точки мінімуму. Однак перевірка умови убування на кожній ітерації є досить трудомісткою, тому в методі градієнтного спуску з постійним кроком задають величину кроку $\alpha = \alpha^{(k)}$ постійною і досить малою, щоб можна було використовувати цей крок на будь ітерації. При цьому доводиться миритися з можливо великою кількістю ітерацій.

Схема алгоритму.

1. Задається початкове наближення $x^{(k)}$, постійний крок α, умови зупинки алгоритму ε. Обчислюється значення градієнта $\nabla f\left(x^{(k)}\right) = f'\left(x^{(k)}\right)$- напрям пошуку. Присвоюється номер ітерації $k = 0$.

2. Обчислюється точка $x^{(k+1)}$ наступної ітерації:

$$x^{(k+1)} = x^{(k)} - \alpha f'\left(x^{(k)}\right)$$

або, в координатній формі:

$$x_i^{(k+1)} = x_i^{(k)} - \alpha \frac{\partial f\left(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}\right)}{dx_i^{(k)}},$$

де $i = 1,2,...,n$.

3. Обчислюється значення градієнта в точці $x^{(k+1)}$: $\nabla f\left(x^{(k+1)}\right)=f'\left(x^{(k+1)}\right)$ або, в координатній формі:

$$\left\{\frac{\partial f\left(x_1^{(k)},x_2^{(k)},...,x_n^{(k)}\right)}{dx_1^{(k)}},...,\frac{\partial f\left(x_1^{(k)},x_2^{(k)},...,x_n^{(k)}\right)}{dx_n^{(k)}}\right\}.$$

4. Якщо $\left\|f'\left(x^{(k+1)}\right)\right\|<\varepsilon$, то ітераційний процес пошуку мінімуму цільової функцій закінчується. При цьому $x^*=x^{(k+1)}$ і $y^*=f\left(x^{(k+1)}\right)$. При невиконанні цієї умови виконується перехід до кроку 2.

Градієнтний метод з діленням кроку.

У методі градієнтного спуску з діленням кроку величина кроку $\alpha^{(k)}$ вибирається так, щоб виконувалася нерівність:

$$f\left(x^{(k)}-\alpha^{(k)}f'\left(x^{(k)}\right)\right)-f\left(x^{(k)}\right)\le-\delta\alpha_k\left\|f'\left(x^{(k)}\right)\right\|^2,$$

де $0<\delta<1$ - довільно вибрана постійна (одна і та ж для всіх ітерацій). Ця умова на вибір кроку $\alpha^{(k)}$ більш жорстка, ніж умова спадання, але має той же зміст: функція повинна спадати від ітерації до ітерації. Однак при виконанні останньої нерівності функція буде зменшуватися на гарантовану величину, яка визначається правою частиною цієї нерівності.

Процес вибору кроку відбувається наступним чином. Вибираємо число $\alpha>0$, одне і те ж для всіх ітерацій. На k – й ітерації перевіряємо виконання нерівності при $\alpha^{(k)}=\alpha$. Якщо вона виконується, то вважаємо $\alpha^{(k)}=\alpha$ і переходимо до наступної ітерації. Якщо ні, то крок $\alpha^{(k)}$ зменшуємо кожен раз в два рази, до тих пір, поки вона не виконається.

Схема алгоритму.

1. Задається початкове наближення $x^{(0)}$, постійна величина δ, похибка обчислення мінімуму ε. Обчислюється значення градієнта $\nabla f\left(x^{(0)}\right)=f'\left(x^{(0)}\right)$- напрям пошуку. Присвоюється номер ітерації $k=0$.

2. Перевіряється умова:

$$f\left(x^{(k)}-\alpha f'\left(x^{(k)}\right)\right)\le-\delta\alpha\left\|f'\left(x^{(k)}\right)\right\|^2.$$

Якщо умова виконується, то потрібно перейти до кроку 3, інакше потрібно поділити значення $\alpha\,(\alpha=\alpha/2)$ і повторюємо крок 2.

3. Обчислюється точка $x^{(k+1)}$ наступної ітерації:

$$x^{(k+1)} = x^{(k)} - \alpha f'\left(x^{(k)}\right)$$

4. Обчислюється значення градієнта в точці $x^{(k+1)}$: $\nabla f\left(x^{(k+1)}\right) = f'\left(x^{(k+1)}\right)$.

5. Якщо $\left\| f'\left(x^{(k+1)}\right) \right\| \le \varepsilon$, то ітераційний процес пошуку мінімуму цільової функцій закінчується. При цьому $x^* = x^{(k+1)}$ і $y^* = f\left(x^{(k+1)}\right)$. При невиконанні цієї умови $k = k+1$ і виконується перехід до кроку 2.

Метод найшвидшого спуску.

У градієнтному методі з постійним кроком величина кроку, що забезпечує спадання функції $f(x)$ від ітерації до ітерації, є дуже малою, що призводить до необхідності проводити велику кількість ітерації для досягнення точки мінімуму. Тому методи спуску зі змінним кроком є більш економними. Алгоритм, на кожній ітерації якого крок $\alpha^{(k)}$ вибирається з умови мінімуму функції $f(x)$ в напрямку руху, називається методом найшвидшого спуску. Зрозуміло, що цей спосіб вибору $\alpha^{(k)}$ складніший від розглянутих раніше методів:

$$f\left(x^{(k)} - \alpha^{(k)} f'\left(x^{(k)}\right)\right) = \min_{\alpha \ge 0} f\left(x^{(k)} - \alpha f'\left(x^{(k)}\right)\right).$$

Реалізація методу найшвидшого спуску передбачає розв’язання на кожній ітерації досить трудомісткою допоміжної задачі одномірної мінімізації. Як правило, метод найшвидшого спуску, тим не менш, дає виграш в числі машинних операцій, оскільки забезпечує рух з найбільш вигідним кроком, бо рішення задачі одномірної мінімізації пов'язано з додатковими обчисленнями тільки самої функції $f(x)$, тоді як основний машинний час затрачається на обчислення її градієнта $f'\left(x^{(k)}\right)$.

Потрібно мати на увазі, що одномірну мінімізацію можна виконувати будь-яким методом одномірної оптимізації, розглянутим в лабораторній роботі №2. Це призводить до різних варіантів методу найшвидшого спуску.

Схема алгоритму.

1. Задається початкове наближення $x^{(0)}$, умова зупинки алгоритму ε. Обчислюється значення градієнта $\nabla f\left(x^{(0)}\right) = f'\left(x^{(0)}\right)$- напрям пошуку. Присвоюється номер ітерації $k = 0$.

2. Обчислюється точка $x^{(k+1)}$ наступної ітерації:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} f'\left(x^{(k)}\right),$$

де $\alpha^{(k)}$ - мінімум задачі одномірної мінімізації:

$$\alpha^{(k)} = \min_{\alpha \ge 0} f\left(x^{(k)} - \alpha f'\left(x^{(k)}\right)\right).$$

3. Обчислюється значення градієнта в точці $x^{(k+1)}$: $f'\left(x^{(k+1)}\right)$.

4. Якщо $\left\| f'\left(x^{(k+1)}\right)\right\| \le \varepsilon$, то ітераційний процес пошуку мінімуму цільової функцій закінчується. При цьому $x^* = x^{(k+1)}$ і $y^* = f\left(x^{(k+1)}\right)$. При невиконанні цієї умови $k = k+1$ і виконується перехід до кроку 2.

Метод других похідних (метод Ньютона).

Напрямок пошуку, що відповідає найшвидшому спуску, можна інтерпретувати як наслідок лінійної апроксимації цільової функції. З іншого боку, методи других похідних, серед яких найкраще відомий метод Ньютона, виникли через квадратичну апроксимацію функції $f(x)$. Вони використовують інформацію другого порядку, що міститься у других частинних похідних цільової функції $f(x)$ по незалежних змінних.

Напрямок пошуку $p^{(k)}$ в методі Ньютона вибирається наступним чином. Квадратична апроксимація функції $f(x)$ через змінні $\Delta x^{(k)} = x^{(k+1)} - x^{(k)}$ представляється наступним чином:

$$f\left(x^{(k+1)}\right) = f\left(x^{(k)}\right) + \nabla^T f\left(x^{(k)}\right)\Delta x^{(k)} + \frac{1}{2}\left(\Delta x^{(k)}\right)^T \nabla^2 f\left(x^{(k)}\right)\Delta x^{(k)} . \qquad (3.6)$$

Мінімум функції $f(x)$ в напрямку $\Delta x^{(k)}$ визначається диференціюванням $f(x)$ по кожній з компонент Δx . Останнє призводить до

$$\Delta x^{(k)} = -\left[\nabla^2 f\left(x^{(k)}\right)\right]^{-1} \nabla f\left(x^{(k)}\right) , \qquad (3.7)$$

де $\left[\nabla^2 f\left(x^{(k)}\right)\right]^{-1}$ - матриця, обернена матриці Гессе $H\left(x^{(k)}\right)$ (матриця других частинних похідних $f(x)$ по x, обчислена в точці $x^{(k)}$).

Перехід з $x^{(k)}$ в $x^{(k+1)}$ за методом Ньютона визначається як:

$$x^{(k+1)} = x^{(k)} - \left[\nabla^2 f\left(x^{(k)}\right)\right]^{-1} \vec{\nabla} f\left(x^{(k)}\right) . \qquad (3.8)$$

Необхідно відмітити, що тут і напрямок і величина кроку точно визначені. Якщо $f(x)$ - квадратична функція, то для досягнення мінімуму $f(x)$ достатньо тільки одного кроку. Але в разі загальної нелінійної цільової функції мінімуму $f(x)$ не можна досягнути за один крок, тому рівняння (3.8) після введення коефіцієнта довжини кроку α зазвичай записують у наступному вигляді:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \left[\nabla^2 f\left(x^{(k)}\right) \right]^{-1} \nabla f\left(x^{(k)}\right) . \qquad (3.9)$$

Рівняння (3.9) найчастіше записують у вигляді:

$$x^{(k+1)} = x^{(k)} - \lambda^{(k)} H^{-1}\left(x^{(k)}\right) \nabla f\left(x^{(k)}\right) . \qquad (3.10)$$

Зауважимо, що рівняння (3.10) включає в себе обертання матриці, і необхідно дотримуватися обережності з тим, щоб обрана процедура розв’язку забезпечувала додатну визначеність оберненої матриці. Зауважимо, що цей метод вимагає обчислення значень аналітичних других частинних похідних або їх апроксимацій, що може виявитися в деяких випадках непрактичним. Критерій, який забезпечить збіжність методу Ньютона у припущенні, що функція $f(x)$ двічі диференційована, полягає в тому, що матриця, обернена матриці Гессе цільової функції, повинна бути додатно визначеною:

$$\left[\nabla^2 f\left(x^{(k)}\right) \right]^{-1} \equiv H^{-1}\left(x^{(k)}\right) > 0 \ . \qquad (3.11)$$

Завдання до лабораторної роботи

1. Реалізувати алгоритм і програму мінімізації багатомірної функції з заданою точністю розглянутими методами. Варіанти тестових функцій приведені в табл. 3.1. Для знаходження похідних цільової функції скористатися аналітичним розв’язком.

Табл. 3.1. Цільові функції для тестування градієнтних методів оптимізації

№ п/п	Функція	Початков. вектор x^*	Значення $f(x^*)$ в т. x^*	
			$f(x^*)$	x^*
1	$100\left(x_2 - x_1^2\right)^2 + \left(1 - x_1\right)^2$	(-1.2, 1.0)	0	(1.0, 1.0)
2	$\left(x_2 - x_1^2\right)^2 + \left(1 - x_1\right)^2$	(-1.2, 1.0)	0	(1.0, 1.0)
3	$\left(x_2 - x_1^2\right)^2 + 100\left(1 - x_1\right)^2$	(-1.2, 1.0)	0	(1.0, 1.0)
4	$100\left(x_2 - x_1^3\right)^2 + \left(1 - x_1\right)^2$	(-1.2, 1.0)	0	(1.0, 1.0)

5	$\left[1.5-x_1\left(1-x_2\right)\right]^2+$ $\left[2.25-x_1\left(1-x_2^2\right)\right]^2+$ $\left[2.625-x_1\left(1-x_2^3\right)\right]^2$	(-1.2, 1.0)	0	(3.0, 0.5)
6	$\left(x_2-x_1^2\right)^2+\left(1-x_1\cdot x_2\right)^2$	(-3.0, 2.0)	0	(1.0, 1.0)
7	$\sum_{i=1}^{n-1}(100\left(x_{i+1}-x_i^2\right)^2+$ $\left(x_i-1\right)^2)$, n=4	(-1, …, -1)	0	(1.0, …,1.0)