

ZADANIA OPTYMALIZACJI BEZ OGRANICZEŃ

Maciej Patan

Instytut Sterowania i Systemów Informatycznych
Uniwersytet Zielonogórski

WSTĘP

Zadanie minimalizacji bez ograniczeń

$$f(\hat{x}) = \min_{x \in \mathbb{R}^n} f(x)$$

$f : \mathbb{R}^n \rightarrow \mathbb{R}$ – funkcja ograniczona z dołu

Algorytm rozwiązywania

Rekurencyjny proces obliczeniowy, który na podstawie znajomości bieżącego punktu x^k generuje nowy punkt x^{k+1} , w rezultacie czego startując z punktu x^1 zostaje utworzony ciąg punktów $\{x^k\}_{k=1}^{\infty}$

Warunki konieczne i dostateczne dla zadań bez ograniczeń

Tw. 1. Niech $f : \mathbb{R}^n \rightarrow \mathbb{R}^1$ będzie funkcją klasy C^1 . Warunkiem koniecznym na to, aby punkt $x^* \in \mathbb{R}^n$ był minimum lokalnym funkcji f jest

$$\nabla f(x^*) = 0 \quad (1)$$

Jeżeli dodatkowo funkcja f jest wypukła, to warunek (1) jest warunkiem koniecznym i wystarczającym na to, aby x^* był minimum globalnym

Tw. 2. Niech $f : \mathbb{R}^n \rightarrow \mathbb{R}^1$ będzie funkcją klasy C^2 . Jeżeli punkt $x^* \in \mathbb{R}^n$ jest minimum lokalnym to

$$\nabla f(x^*) = 0 \quad (2)$$

oraz hesjan

$$\nabla^2 f(x^*) \quad (3)$$

jest macierzą dodatnio określoną.

Podział metod

1. metody losowe (przypadkowe)

2. metody deterministyczne

- tworzenie kierunków poszukiwań
 - ◇ modyfikowana baza – bezgradientowe
 - ◇ modyfikowany kierunek – gradientowe
- kolejny punkt wzdłuż danego kierunku poszukiwań
 - ◇ metody dyskretne (poszukiwań prostych)
 - ◇ metody z minimalizacją (kierunków poprawy)

METODA ZŁOTEGO PODZIAŁU

- Metoda stosowana dla funkcji unimodalnych
- W każdej iteracji generuje się dwa punkty x_1 i x_2 leżące wewnątrz rozpatrywanego przedziału $[a, b]$
- Na podstawie wartości funkcji $f(x_1)$ i $f(x_2)$ określa się, czy minimum leży w przedziale $[x_1, b]$, czy też $[a, x_2]$
- W każdej następnej iteracji wyznaczany podprzedział obejmujący poszukiwane minimum zmniejsza się o stały czynnik $\alpha = 0,618$
- Minimum można określić z żądaną dokładnością

Algorytm:

Krok 0. Określ przedział $[a, b]$ tak, aby obejmował minimum funkcji $f(x^*)$

Krok 1. Wyznacz $x_1 = a + (1 - \alpha)(b - a)$, $x_2 = a + \alpha(b - a)$
gdzie $\alpha = 0.618$ oraz oblicz $f(x_1)$ i $f(x_2)$.

Krok 2. Jeśli $f(x_1) < f(x_2)$ to idź do kroku 3, w przeciwnym przypadku do kroku 4.

Krok 3. Podstaw $b := x_2$ i $x_2 := x_1$, oblicz $x_1 = a + (1 - \alpha)(b - a)$
oraz $f(x_1)$ i przejdź do kroku 5.

Krok 4. Podstaw $a := x_1$ i $x_1 := x_2$, oblicz $x_2 = a + \alpha(b - a)$
oraz $f(x_2)$ i przejdź do kroku 5.

Krok 5. Jeżeli $b - a < \varepsilon$ (ε - zadana dokładność) to zakończ obliczenia,
przyjmując $x^* = x_1$ jeśli $f(x_1) < f(x_2)$ lub $x^* = x_2$ jeśli $f(x_1) > f(x_2)$;
w przeciwnym przypadku przejdź do kroku 2.

METODA APROKSYMACJI KWADRATOWEJ

- Zakłada się, że w otoczeniu minimum, funkcję można zaproksymować wielomianem drugiego stopnia
 - ❶ wartość funkcji jest wyliczana w trzech kolejnych punktach i za ich pomocą określany zostaje wielomian interpolacyjny drugiego stopnia
 - ❷ wyznaczenie przedziału zawierającego minimum, a następnie zastosowanie interpolacji kwadratowej

Założmy, że znamy wartości funkcji celu f_a , f_b i f_c w kolejnych trzech punktach x_a , x_b i x_c ($x_a < x_b < x_c$)

Wielomian interpolacyjny Lagrange'a

$$f(x) = f_a \frac{(x - x_b)(x - x_c)}{(x_a - x_b)(x_a - x_c)} + f_b \frac{(x - x_a)(x - x_c)}{(x_b - x_a)(x_b - x_c)} + f_c \frac{(x - x_a)(x - x_b)}{(x_c - x_a)(x_c - x_b)} \quad (4)$$

Warunkiem koniecznym istnienia ekstremum jest

$$\frac{df(x)}{dx} = 0 \Big|_{x=x_m} \quad (5)$$

Otrzymujemy

$$f_a \frac{2x_m - (x_b + x_c)}{(x_a - x_b)(x_a - x_c)} + f_b \frac{2x_m - (x_a + x_c)}{(x_b - x_a)(x_b - x_c)} + \quad (6)$$

$$+ f_c \frac{2x_m - (x_a + x_b)}{(x_c - x_a)(x_c - x_b)} = 0 \quad (7)$$

gdzie

$$x_m = \frac{1}{2} \frac{(x_b^2 - x_c^2)f_a + (x_c^2 - x_a^2)f_b + (x_a^2 - x_b^2)f_c}{(x_b - x_c)f_a + (x_c - x_a)f_b + (x_a - x_b)f_c} \quad (8)$$

Właściwości

- ❑ Znaleziony punkt może okazać się maksimum w kierunku
- ❑ Niewłaściwa zamiana punktów wyjściowych może doprowadzić do rozbieżności metody
- ❑ Algorytm jest jednostajnie minimalizujący – znajduje minimum w kierunku z żądaną dokładnością
- ❑ Jeżeli daną funkcję można z powodzeniem przybliżyć wielomianem drugiego stopnia, metoda jest szybko zbieżna

METODA NAJWIĘKSZEGO SPADKU

- Algorytm iteracyjny

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \eta \mathbf{d}_k \quad (9)$$

gdzie \mathbf{d}_k – kierunek poprawy w k -tej iteracji, η – krok

- Kierunek poprawy wyznacza się na podstawie znajomości gradientu funkcji celu

$$\mathbf{d}_k = -\nabla f(\mathbf{x}_k) \quad (10)$$

- Gradient jest kierunkiem największego wzrostu – przy gradiencie stosuje się znak minus
- Do rozpoczęcia procedury potrzebny jest dowolnie wybrany punkt startowy
- Warunek stopu

$$|\mathbf{x}_{k+1} - \mathbf{x}_k| < \varepsilon \quad (11)$$

METODA NEWTONA

- Metody dobierające wartość kroku uczenia automatycznie
- Rozwinięcie w szereg funkcji kosztu f względem \mathbf{x}_0

$$f(\mathbf{x}) = f_0 + (\mathbf{x} - \mathbf{x}_0)\nabla f(\mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)\mathbf{H}(\mathbf{x} - \mathbf{x}_0) + \dots \quad (12)$$

gdzie \mathbf{H} – hesjan (macierz drugich pochodnych)

- Po zróżniczkowaniu otrzymujemy

$$\nabla f(\mathbf{x}) = \nabla f(\mathbf{x}_0) + \mathbf{H}(\mathbf{x} - \mathbf{x}_0) + \dots \quad (13)$$

- Pomijając człony wyższego rzędu i przyrównując do zera

$$\nabla f(\mathbf{x}_0) + \mathbf{H}(\mathbf{x} - \mathbf{x}_0) = 0 \quad (14)$$

lub

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}\nabla f(\mathbf{x}_k) \quad (15)$$

Metoda Newtona – właściwości

- metoda kosztowna obliczeniowo
 - ◇ potrzeba odwracania hesjanu
 - ◇ potrzeba obliczania drugich pochodnych
- metoda niestabilnie numerycznie w przypadku rozpoczęcia obliczeń daleko od punktu optymalnego
- algorytm niepraktyczny w przypadku wielowymiarowym
- metoda stosowana jako punkt odniesienia
- praktycznie stosuje się techniki przybliżające macierz hesjanu – metody

quasi-newtonowskie

METODY QUASI-NEWTONOWSKIE

- Metody przybliżające macierz hesjanu
- Aktualizacja rozwiązania

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{B}_k \nabla f(\mathbf{x}_k) \quad (16)$$

gdzie $\mathbf{B} \simeq \mathbf{H}^{-1}$

- Metoda BFGS (Broyden-Fletcher-Goldfarb-Shanno)

$$\mathbf{B}_k = \left[\mathbf{I} - \frac{\Delta \mathbf{x}_k \Delta \mathbf{G}_k^T}{\Delta \mathbf{G}_k^T \Delta \mathbf{x}_k} \right] \mathbf{B}_{k-1} \left[\mathbf{I} - \frac{\Delta \mathbf{G}_k \Delta \mathbf{x}_k^T}{\Delta \mathbf{G}_k^T \Delta \mathbf{x}_k} \right] \quad (17)$$

gdzie $\Delta \mathbf{x}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$ i $\Delta \mathbf{G}_k = \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1})$

➤ Metoda DFP (Davidon-Fletcher-Powell)

$$\mathbf{B}_k = \mathbf{B}_{k-1} + \frac{\Delta \mathbf{x}_{k-1} \Delta \mathbf{x}_k^T}{\Delta \mathbf{x}_k^T \Delta \mathbf{G}_k} - \frac{\mathbf{B}_{k-1} \Delta \mathbf{G}_k \Delta \mathbf{G}_k^T \mathbf{B}_{k-1}}{\Delta \mathbf{G}_k^T \mathbf{B}_{k-1} \Delta \mathbf{G}_k} \quad (18)$$

gdzie $\Delta \mathbf{x}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$ i $\Delta \mathbf{G}_k = \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1})$

Metody quasi-newtonowskie – właściwości

- ❑ Brak operacji odwracania macierzy hesjanu
- ❑ Blisko rozwiązania zbieżność jest dobra
- ❑ Początkowo zbieżność jest słaba (na początku przybliżenie odwrotności hesjanu jest słabe, polepszane z iteracji an iterację)

METODA LEVENBERGA-MARQUARDTA

- Przybliżenie metody Newtona
- Macierz hesjanu przybliża się za pomocą macierzy jacobianu

$$\mathbf{H}_k = \mathbf{J}_k^T \mathbf{J}_k \quad (19)$$

gdzie \mathbf{J} – jacobian

- reguła aktualizacji punktu

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{J}_k^T \mathbf{J}_k + \mu_k \mathbf{I})^{-1} \nabla f(\mathbf{x}_k) \quad (20)$$

gdzie μ – współczynnik

Właściwości

- ❑ Szybka zbieżność do rozwiązania
- ❑ Dla μ dużego metoda staje się metodą największego spadku
- ❑ Dla μ małego metoda staje się metodą Newtona
- ❑ W czasie uczenia μ jest zmniejszany każdorazowo po wykonaniu kroku zmniejszającego wartość funkcji celu
- ❑ Dzięki zmniejszaniu μ algorytm staje się metodą Newtona blisko rozwiązania
- ❑ μ jest zwiększany tylko w przypadkach kiedy za dużą jego wartość powoduje wzrost wartości funkcji celu

GRADIENT, HESJAN, JAKOBIAN

Gradient

$$\nabla f(\mathbf{x}) = \left[\frac{\partial f(\mathbf{x})}{\partial x_1} \quad \frac{\partial f(\mathbf{x})}{\partial x_2} \quad \cdots \quad \frac{\partial f(\mathbf{x})}{\partial x_n} \right] \quad (21)$$

Hesjan

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{bmatrix} \quad (22)$$

Jakobian

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \frac{\partial f_1(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_m} \\ \frac{\partial f_2(\mathbf{x})}{\partial x_1} & \frac{\partial f_2(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_2(\mathbf{x})}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(\mathbf{x})}{\partial x_1} & \frac{\partial f_n(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_n(\mathbf{x})}{\partial x_m} \end{bmatrix} \quad (23)$$