Complexidade de métodos iterativos para problemas de otimização

Majela Pentón

UFBA

Salvador, Novembro 2023.

Problema de minimização

Problema de minimização

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e $D \subset \mathbb{R}^n$

$$(P) \qquad \begin{aligned} &\min f(x) \\ &\text{s.a. } x \in D \end{aligned}$$

Problema de minimização

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e $D \subset \mathbb{R}^n$

$$(P) \qquad \begin{aligned} &\min f(x) \\ &\text{s.a. } x \in D \end{aligned}$$

- f função objetivo

Problema de minimização

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e $D \subset \mathbb{R}^n$

$$(P) \qquad \begin{aligned} &\min f(x) \\ &\text{s.a. } x \in D \end{aligned}$$

- f função objetivo
- D conjunto viável do problema

Problema de minimização

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e $D \subset \mathbb{R}^n$

$$(P) \qquad \begin{aligned} &\min f(x) \\ &\text{s.a. } x \in D \end{aligned}$$

- f função objetivo
- D conjunto viável do problema
- Problema irrestrito: $D = \mathbb{R}^n$

Problema de minimização

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e $D \subset \mathbb{R}^n$

$$(P) \qquad \begin{aligned} &\min f(x) \\ &\text{s.a. } x \in D \end{aligned}$$

- f função objetivo
- D conjunto viável do problema
- Problema irrestrito: $D = \mathbb{R}^n$
- Problema com restrições: $D \subset \mathbb{R}^n$

Tipos de soluções

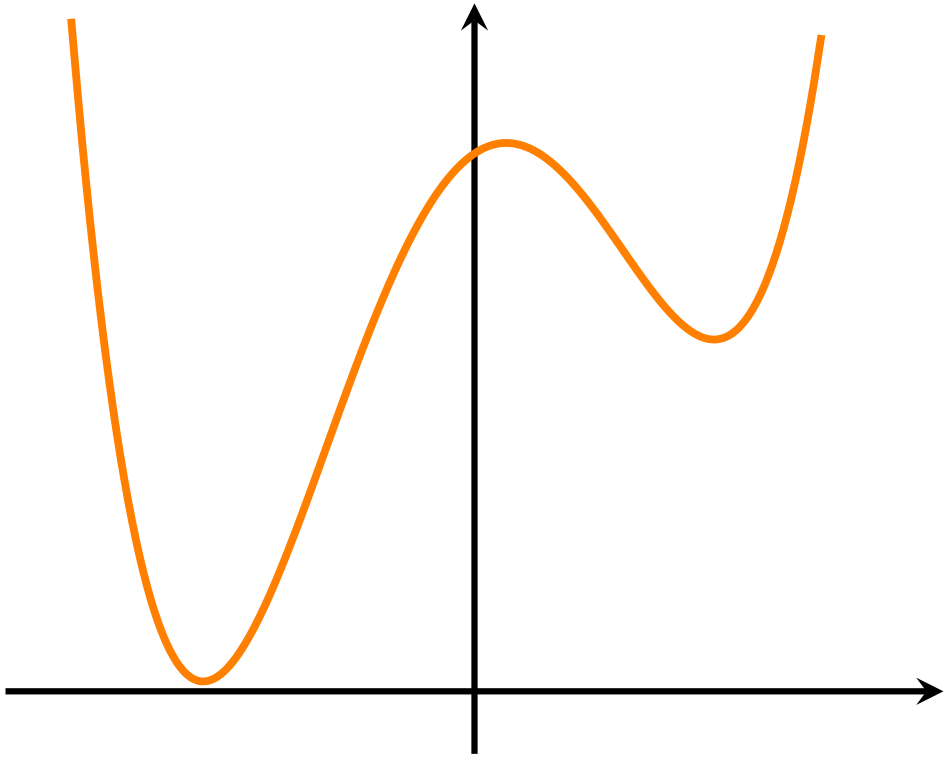
- x^* mínimo global: $f(x^*) \leq f(x)$ para todo $x \in D$.
  - $f^* = f(x^*)$ é chamado de valor ótimo do problema.

Tipos de soluções

- x^* mínimo global: $f(x^*) \leq f(x)$ para todo $x \in D$.
  - $f^* = f(x^*)$ é chamado de valor ótimo do problema.
- x^* mínimo local: $f(x^*) \leq f(x)$ numa vizinhança de V de x^*.

Tipos de soluções

- x^* mínimo global: $f(x^*) \leq f(x)$ para todo $x \in D$.
  - $f^* = f(x^*)$ é chamado de valor ótimo do problema.
- x^* mínimo local: $f(x^*) \leq f(x)$ numa vizinhança de V de x^*.



- Em geral, problemas de otimização são difíceis de resolver.

- Em geral, problemas de otimização são difíceis de resolver.
- Para resolver o problema P usamos um método numérico $\mathcal{M}$

- Em geral, problemas de otimização são difíceis de resolver.
- Para resolver o problema P usamos um método numérico $\mathcal{M}$
- $\mathcal{M}$ é implementado para resolver uma família $\mathcal{F}$ de problemas com características similares (lineares, quadráticos, convexos, etc)

- Em geral, problemas de otimização são difíceis de resolver.
- Para resolver o problema P usamos um método numérico $\mathcal{M}$
- $\mathcal{M}$ é implementado para resolver uma família $\mathcal{F}$ de problemas com características similares (lineares, quadráticos, convexos, etc)
- Desempenho de $\mathcal{M}$ para resolver P: esforço computacional que realiza $\mathcal{M}$ para resolver P aproximadamente, i.e. **com precisão** $\epsilon > 0$.

- Em geral, problemas de otimização são difíceis de resolver.
- Para resolver o problema P usamos um método numérico $\mathcal{M}$
- $\mathcal{M}$ é implementado para resolver uma família $\mathcal{F}$ de problemas com características similares (lineares, quadráticos, convexos, etc)
- Desempenho de $\mathcal{M}$ para resolver P: esforço computacional que realiza $\mathcal{M}$ para resolver P aproximadamente, i.e. **com precisão** $\epsilon > 0$.
- Precisão ϵ: uma **condição de parada** para o método define uma **solução aproximada** do problema.

Complexidade analítica

Número de chamadas ao **oráculo** que o método realiza para encontrar uma ϵ-solução de P.

Complexidade analítica

Número de chamadas ao **oráculo** que o método realiza para encontrar uma ϵ-solução de P.

- Para resolver P, $\mathcal{M}$ precisa obter informação sobre o problema (f, ∇f,...)
- O processo de coletar informações específicas de P é chamado de **oráculo**

Complexidade analítica

Número de chamadas ao **oráculo** que o método realiza para encontrar uma ϵ-solução de P.

- Para resolver P, $\mathcal{M}$ precisa obter informação sobre o problema (f, ∇f,...)
- O processo de coletar informações específicas de P é chamado de **oráculo**

Complexidade aritmética

Número total de operações aritméticas (o.a.) que realiza o oráculo e o método para encontrar uma ϵ-solução de P.

Exemplo

$$(\mathcal{P}) \quad \min_{x \in C_n} f(x)$$

- $C_n = \{x \in \mathbb{R}^n : 0 \leq x_j \leq 1,\ j = 1, \ldots, n\}$
- f é L-Lipschitz contínua em relação a norma ℓ_∞ em C_n:

$$|f(x) - f(y)| \leq L \|x - y\|_\infty ,$$

para todo $x, y \in C_n$.

Seja $f^* = \min_{x \in C_n} f(x)$ o valor ótimo do problema $\mathcal{P}$.

Seja $f^* = \min_{x \in C_n} f(x)$ o valor ótimo do problema $\mathcal{P}$.

Solução aproximada

Dado $\epsilon > 0$, $\overline{x} \in C_n$ é uma ϵ-**solução** de $\mathcal{P}$ se $f(\overline{x}) - f^* \leq \epsilon$.

Seja $f^* = \min_{x \in C_n} f(x)$ o valor ótimo do problema $\mathcal{P}$.

Solução aproximada

Dado $\epsilon > 0$, $\overline{x} \in C_n$ é uma ϵ-**solução** de $\mathcal{P}$ se $f(\overline{x}) - f^* \leq \epsilon$.

Qual é o melhor desempenho de um método $\mathcal{M}$ para resolver os problemas da classe de $\mathcal{P}$?

Seja $f^* = \min_{x \in C_n} f(x)$ o valor ótimo do problema $\mathcal{P}$.

Solução aproximada

Dado $\epsilon > 0$, $\overline{x} \in C_n$ é uma ϵ-**solução** de $\mathcal{P}$ se $f(\overline{x}) - f^* \leq \epsilon$.

Qual é o melhor desempenho de um método $\mathcal{M}$ para resolver os problemas da classe de $\mathcal{P}$?

- ♣ Se $\epsilon < \frac{L}{2}$, a complexidade analítica é no mínimo $[\frac{L}{2\epsilon}]^n$.

[Nes04]

$L = 2$, $n = 10$, $\epsilon = 0.01$ ([Nes04])

$L = 2$, $n = 10$, $\epsilon = 0.01$ ([Nes04])

- 10^{20} chamadas ao oráculo
- Uma chamada ao oráculo precisa de 10 o.a.
- Em total 10^{21} o.a.
- O computador realiza 10^6 o.a. por segundo
- Tempo total: 10^{15} segundos
- Um ano tem menos de $3.2 \cdot 10^7$ segundos
- Precisamos 31 250 000 anos

O que fazer?

O que fazer?

Suponhamos $D = \mathbb{R}^n$

O que fazer?

Suponhamos $D = \mathbb{R}^n$

Minimização irrestrita diferenciável

$$\begin{aligned} &\min f(x) \\ &\text{s.a. } x \in \mathbb{R}^n \end{aligned}$$

- f é diferenciável no $\mathbb{R}^n$

O que fazer?

Suponhamos $D = \mathbb{R}^n$

Minimização irrestrita diferenciável

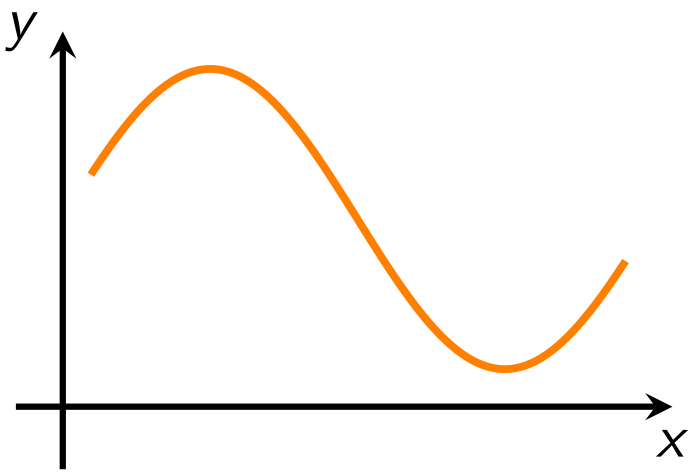
$$\begin{aligned} &\min f(x) \\ &\text{s.a. } x \in \mathbb{R}^n \end{aligned}$$

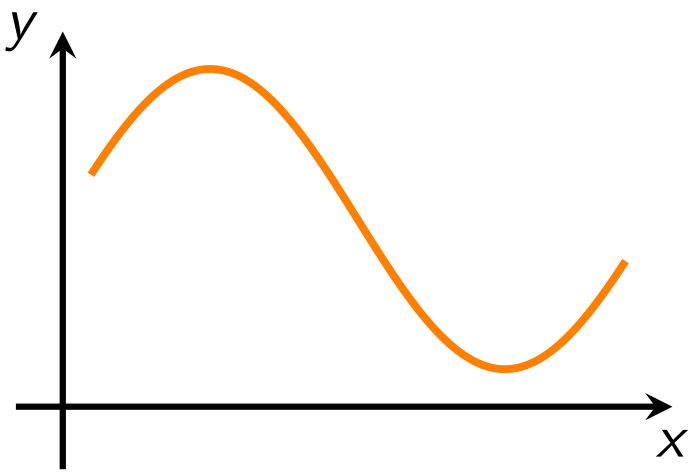
- f é diferenciável no $\mathbb{R}^n$

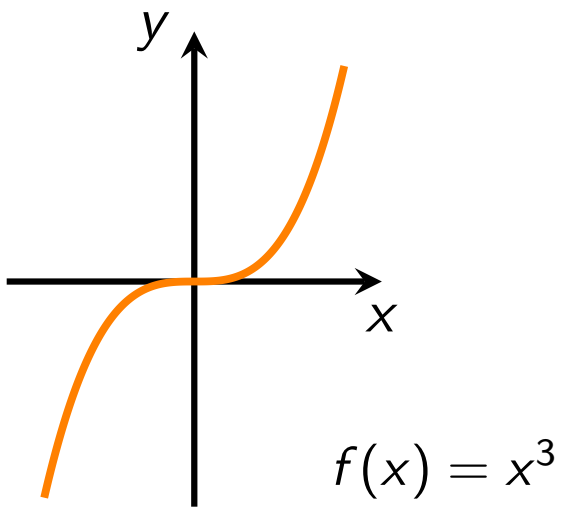
Condição necessária de primeira ordem

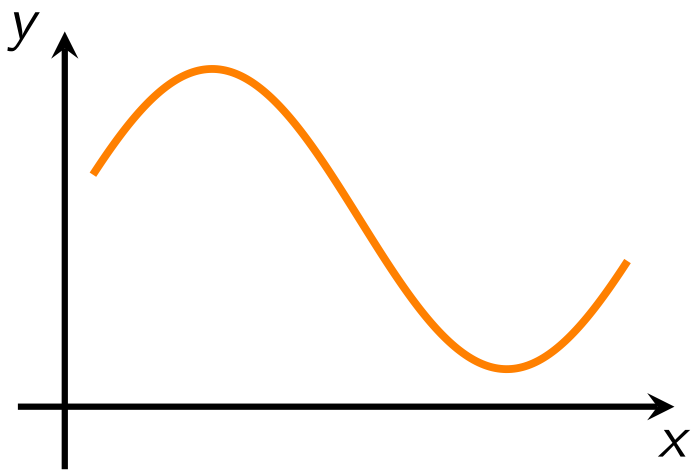
Seja f uma função diferenciável em $\mathbb{R}^n$, se x^* é um mínimo local de f então

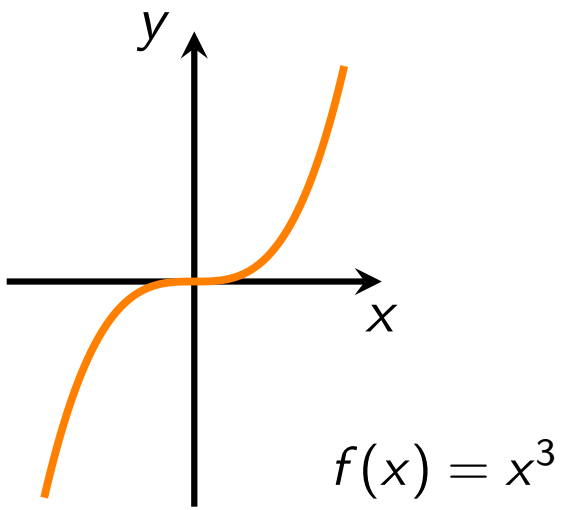
$$\nabla f(x^*) = 0.$$


y
x


y
x

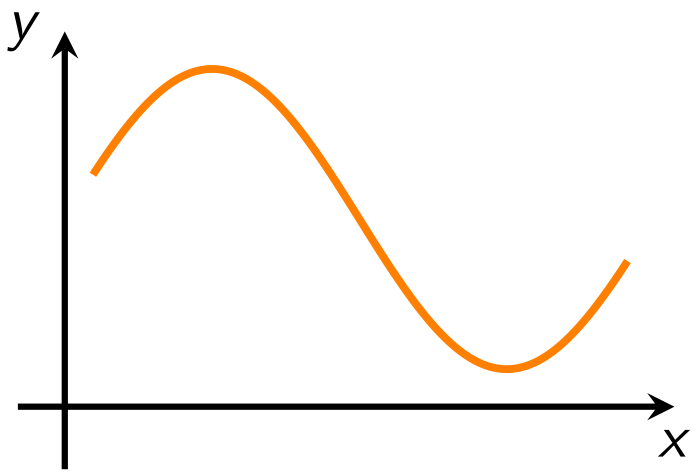

y
x
$f(x) = x^3$

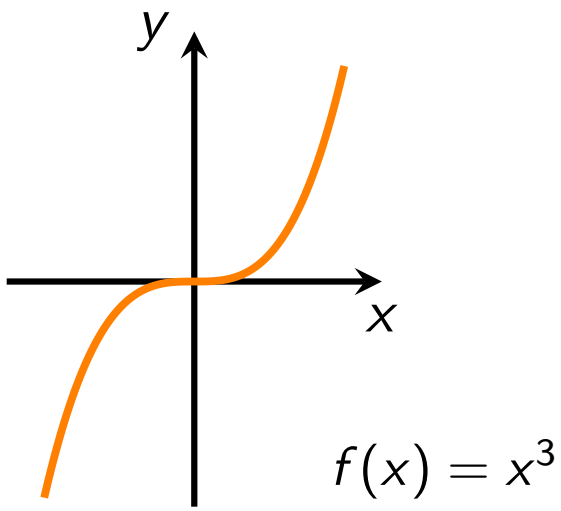




Ponto estacionário

Todo x tal que $\nabla f(x) = 0$ é chamado de **ponto estacionário** do problema.





Ponto estacionário

Todo x tal que $\nabla f(x) = 0$ é chamado de **ponto estacionário** do problema.

- Objetivo: encontrar um mínimo local do problema.

Método do gradiente

Gera uma sequencia $(f(x_k))_{k\in\mathbb{N}}$ tal que $f(x_{k+1}) \leq f(x_k)$

Método do gradiente

Gera uma sequencia $(f(x_k))_{k\in\mathbb{N}}$ tal que $f(x_{k+1}) \leq f(x_k)$

- Escolher um ponto inicial $x_0 \in \mathbb{R}^n$

Método do gradiente

Gera uma sequencia $(f(x_k))_{k\in\mathbb{N}}$ tal que $f(x_{k+1}) \leq f(x_k)$

- Escolher um ponto inicial $x_0 \in \mathbb{R}^n$
- Iterar: $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$, $k = 0, 1, \ldots$

Método do gradiente

Gera uma sequencia $(f(x_k))_{k\in\mathbb{N}}$ tal que $f(x_{k+1}) \leq f(x_k)$

- Escolher um ponto inicial $x_0 \in \mathbb{R}^n$
- Iterar: $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$, $k = 0, 1, \ldots$
- $\alpha_k > 0$ é o **comprimento de passo** tal que $f(x_{k+1}) \leq f(x_k)$.

Método do gradiente

Gera uma sequencia $(f(x_k))_{k\in\mathbb{N}}$ tal que $f(x_{k+1}) \leq f(x_k)$

- Escolher um ponto inicial $x_0 \in \mathbb{R}^n$
- Iterar: $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$, $k = 0, 1, \ldots$
- $\alpha_k > 0$ é o **comprimento de passo** tal que $f(x_{k+1}) \leq f(x_k)$.
  - $\alpha_k = \alpha$ constante para todo k
  - busca linear de Armijo:

$$f(x_k - \alpha_k \nabla f(x_k)) \leq f(x_k) - \frac{\alpha_k}{2} \|\nabla f(x_k)\|^2$$

Convergência a um ponto estacionário [IzS12]

Se ∇f é L-Lipschitz contínuo em $\mathbb{R}^n$, f é limitada inferiormente em $\mathbb{R}^n$ e $\alpha_k = \alpha < 2/L$ para todo k, então

$$\|\nabla f(x_k)\| \to 0, \quad k \to \infty$$

Convergência a um ponto estacionário [IzS12]

Se ∇f é L-Lipschitz contínuo em $\mathbb{R}^n$, f é limitada inferiormente em $\mathbb{R}^n$ e $\alpha_k = \alpha < 2/L$ para todo k, então

$$\|\nabla f(x_k)\| \to 0, \quad k \to \infty$$

Portanto todo **ponto de acumulação** da sequência $(x_k)_{k\in\mathbb{N}}$ é um **ponto estacionário do problema**.

Convergência a um ponto estacionário [IzS12]

Se ∇f é L-Lipschitz contínuo em $\mathbb{R}^n$, f é limitada inferiormente em $\mathbb{R}^n$ e $\alpha_k = \alpha < 2/L$ para todo k, então

$$\|\nabla f(x_k)\| \to 0, \quad k \to \infty$$

Portanto todo **ponto de acumulação** da sequência $(x_k)_{k\in\mathbb{N}}$ é um **ponto estacionário do problema**.

ϵ-solução

Encontrar $\overline{x}$ tal que $\|\nabla f(\overline{x})\| \leq \epsilon$.

Convergência a um ponto estacionário [IzS12]

Se ∇f é L-Lipschitz contínuo em $\mathbb{R}^n$, f é limitada inferiormente em $\mathbb{R}^n$ e $\alpha_k = \alpha < 2/L$ para todo k, então

$$\|\nabla f(x_k)\| \to 0, \quad k \to \infty$$

Portanto todo **ponto de acumulação** da sequência $(x_k)_{k\in\mathbb{N}}$ é um **ponto estacionário do problema**.

ϵ-solução

Encontrar $\overline{x}$ tal que $\|\nabla f(\overline{x})\| \leq \epsilon$.

- O número de iterações para obter $\|\nabla f(\overline{x})\| \leq \epsilon$ é $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$.

Convergência a um ponto estacionário [IzS12]

Se ∇f é L-Lipschitz contínuo em $\mathbb{R}^n$, f é limitada inferiormente em $\mathbb{R}^n$ e $\alpha_k = \alpha < 2/L$ para todo k, então

$$\|\nabla f(x_k)\| \to 0, \quad k \to \infty$$

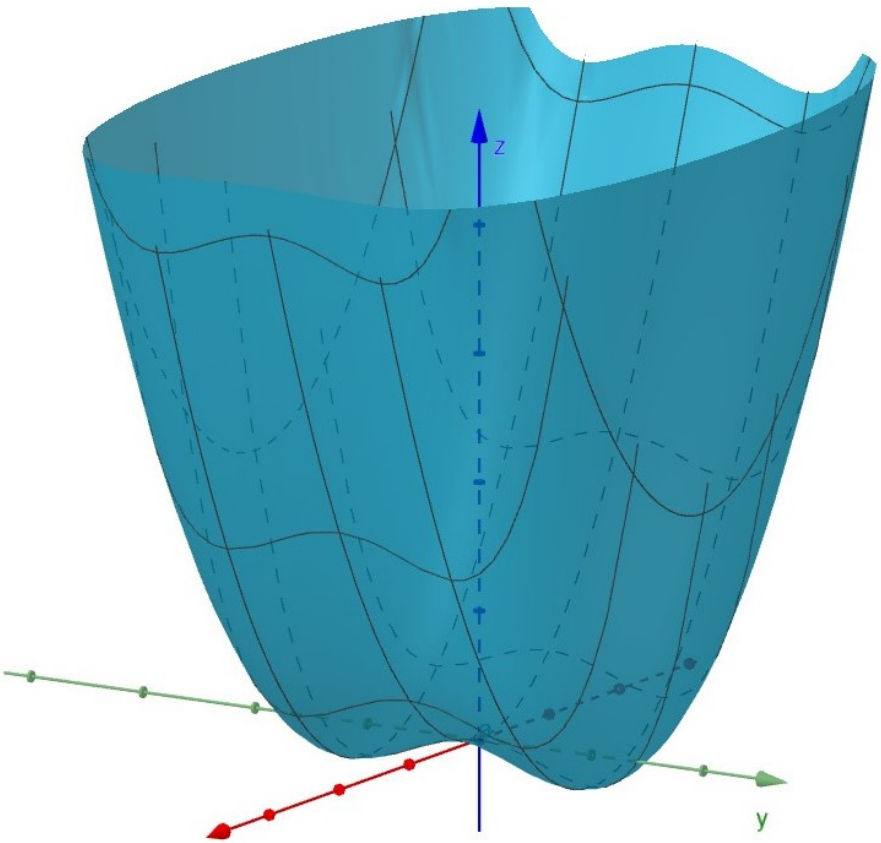
Portanto todo **ponto de acumulação** da sequência $(x_k)_{k\in\mathbb{N}}$ é um **ponto estacionário do problema**.

ϵ-solução

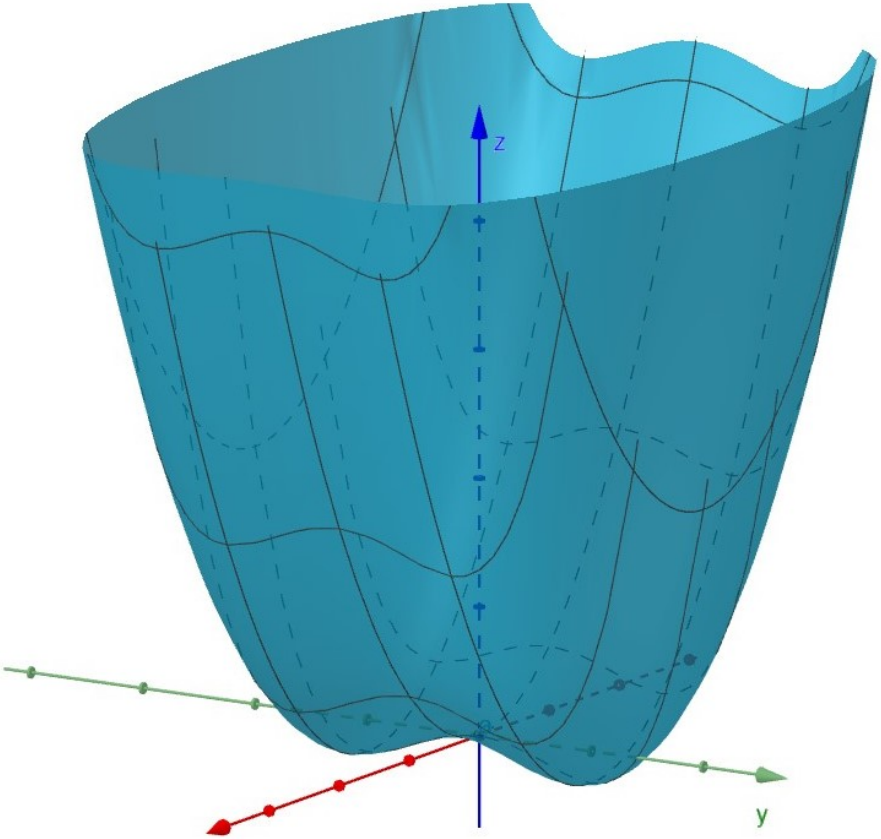
Encontrar $\overline{x}$ tal que $\|\nabla f(\overline{x})\| \leq \epsilon$.

- ♣ O número de iterações para obter $\|\nabla f(\overline{x})\| \leq \epsilon$ é $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$.

O nosso objetivo pode não ser alcançado pelo método do gradiente. Apenas podemos garantir convergência a um ponto estacionário.

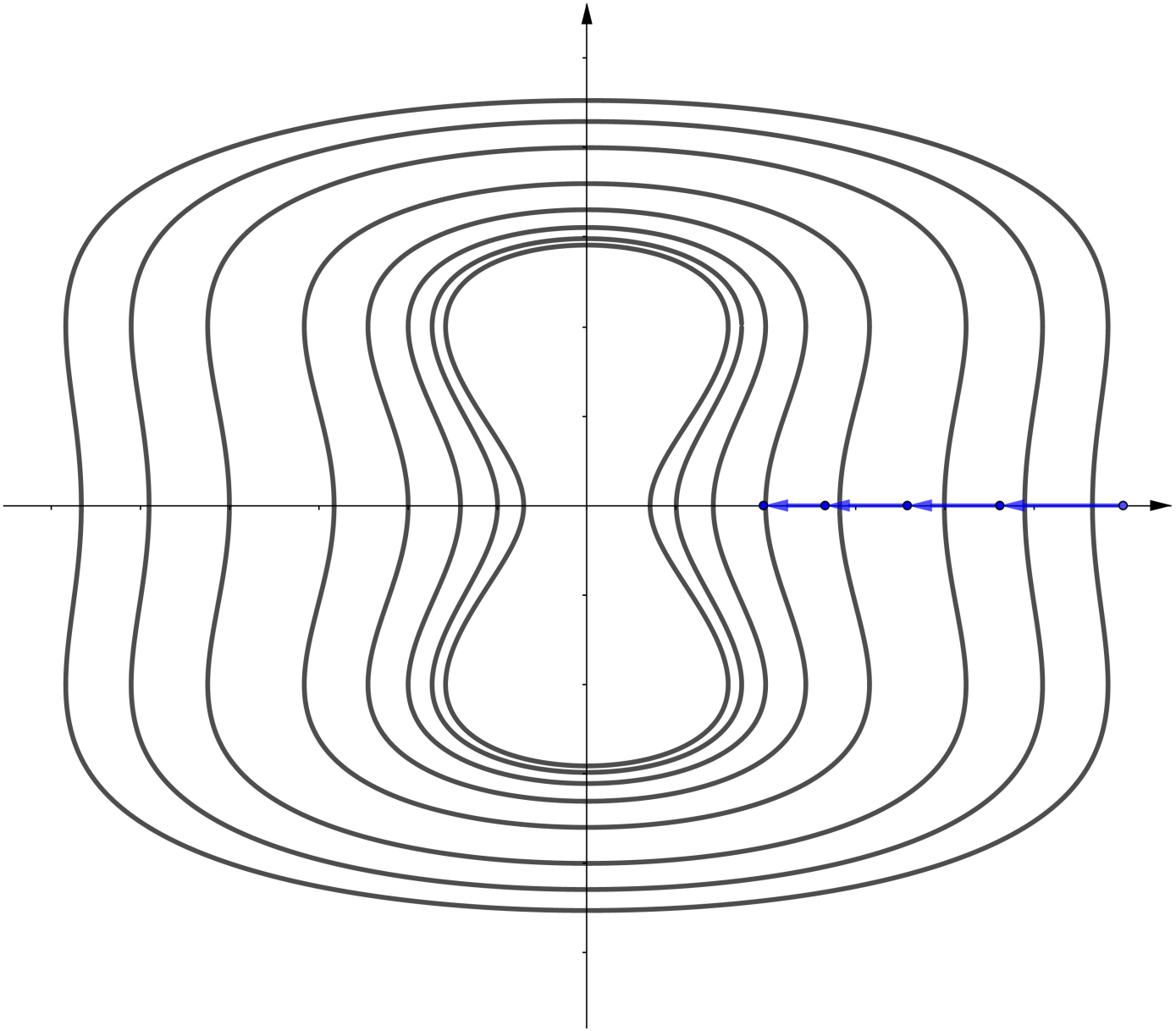


$$f(x, y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$$



$$f(x,y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$$

- $(0,1)$ e $(0,-1)$ mínimos locais
- $(0,0)$ ponto sela



Minimização convexa

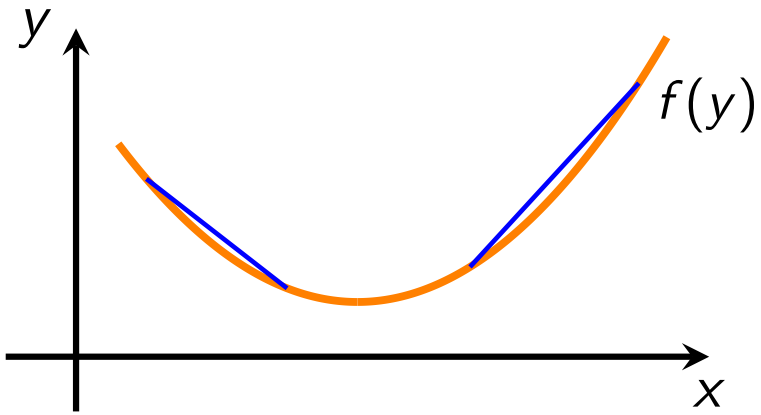
Em (P) consideramos $D = \mathbb{R}^n$ e f uma função **convexa**.

Função convexa

$f : \mathbb{R}^n \to \mathbb{R}$ é convexa se

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$

para todo $t \in [0,1]$ e $x, y \in \mathbb{R}^n$.



- Todo mínimo local é global

- Todo mínimo local é global
- Se a função é diferenciável, a condição necessária de primeira ordem é suficiente.

- Todo mínimo local é global
- Se a função é diferenciável, a condição necessária de primeira ordem é suficiente.

♣ Se $(x_k)_{k\in\mathbb{N}}$ é gerada pelo método do gradiente com passo constante $\alpha_k \equiv 1/L$, então a sequência converge a uma solução global.

- Todo mínimo local é global
- Se a função é diferenciável, a condição necessária de primeira ordem é suficiente.
- ♣ Se $(x_k)_{k\in\mathbb{N}}$ é gerada pelo método do gradiente com passo constante $\alpha_k \equiv 1/L$, então a sequência converge a uma solução global.

ϵ-solução

Dado $\epsilon > 0$, $\overline{x}$ é uma ϵ-solução se $f(\overline{x}) - f^* \leq \epsilon$

- Todo mínimo local é global
- Se a função é diferenciável, a condição necessária de primeira ordem é suficiente.

♣ Se $(x_k)_{k\in\mathbb{N}}$ é gerada pelo método do gradiente com passo constante $\alpha_k \equiv 1/L$, então a sequência converge a uma solução global.

ϵ-solução

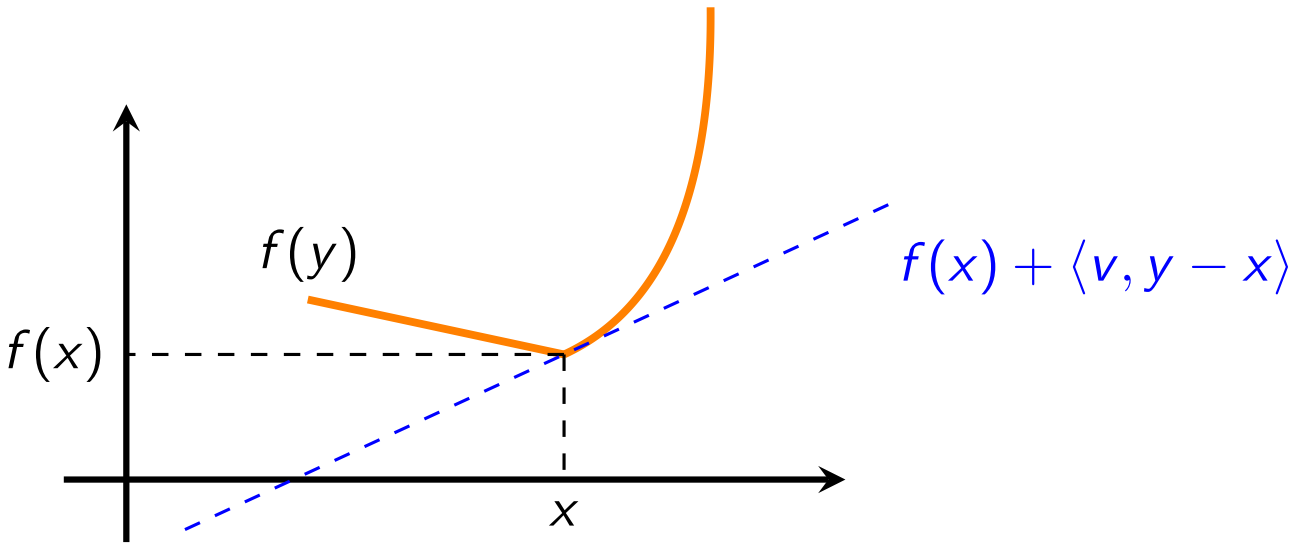
Dado $\epsilon > 0$, $\overline{x}$ é uma ϵ-solução se $f(\overline{x}) - f^* \leq \epsilon$

♣ O número de iterações para obter uma ϵ-**solução** é $\mathcal{O}(1/\epsilon)$.

- Generalização da noção de derivada

  Se $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa, dizemos que $v \in \mathbb{R}^n$ é um **subgradiente** de f no ponto $x \in \mathbb{R}^n$ se

$$f(y) \geq f(x) + \langle v, y - x \rangle, \qquad \forall y \in \mathbb{R}^n.$$

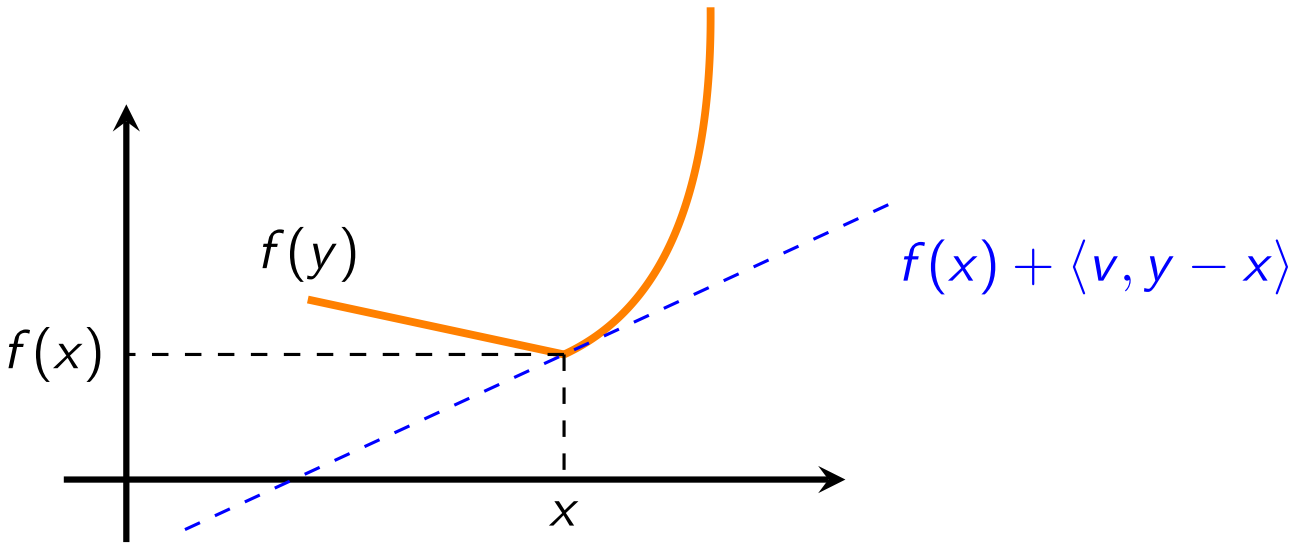


- O subgradiente sempre existe

- Generalização da noção de derivada

  Se $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa, dizemos que $v \in \mathbb{R}^n$ é um **subgradiente** de f no ponto $x \in \mathbb{R}^n$ se

$$f(y) \geq f(x) + \langle v, y - x \rangle, \qquad \forall y \in \mathbb{R}^n.$$



- O subgradiente sempre existe
- x^* é um mínimo de f se e somente se 0 é um subgradiente de f em x^*.

Aprendizado de máquina

Aprendizado de máquina

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{m} f_i(x) + \lambda \mathcal{R}(x)$$

- $f_1, \ldots, f_m$, $\mathcal{R}$ são funções convexas e $\lambda \geq 0$.

Aprendizado de máquina

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{m} f_i(x) + \lambda \mathcal{R}(x)$$

- $f_1, \ldots, f_m$, $\mathcal{R}$ são funções convexas e $\lambda \geq 0$.
- $f_i(x)$ representa o custo de usar x no i-ésimo elemento de algum conjunto de dados.
- $\mathcal{R}(x)$ é um termo de regularização.

Conjunto de dados

$$D = \{(w_i, y_i) \in \mathbb{R}^n \times \mathcal{Y},\ i = 1, \ldots, m\}$$

Conjunto de dados

$$D = \{(w_i, y_i) \in \mathbb{R}^n \times \mathcal{Y},\ i = 1, \ldots, m\}$$

- Classificação: $\mathcal{Y} = \{1, -1\}$
  - SVM: $f_i(x) = \max(0, 1 - y_i x^\top w_i)$ e $\mathcal{R}(x) = \|x\|_2^2$
  - Regressão logística regularizada:
    $f_i(x) = \log(1 + \exp(-y_i x^\top w_i))$ e $\mathcal{R}(x) = \|x\|_2^2$

Conjunto de dados

$$D = \{(w_i, y_i) \in \mathbb{R}^n \times \mathcal{Y},\ i = 1, \ldots, m\}$$

- Classificação: $\mathcal{Y} = \{1, -1\}$
  - SVM: $f_i(x) = \max(0, 1 - y_i x^\top w_i)$ e $\mathcal{R}(x) = \|x\|_2^2$
  - Regressão logística regularizada: $f_i(x) = \log(1 + \exp(-y_i x^\top w_i))$ e $\mathcal{R}(x) = \|x\|_2^2$
- Regressão: $\mathcal{Y} = \mathbb{R}$
  - Regressão linear regularizada: $f_i(x) = (x^\top w_i - y_i)^2$ e $\mathcal{R}(x) = \|x\|_2^2$
  - LASSO: $f_i(x) = (x^\top w_i - y_i)^2$ e $\mathcal{R}(x) = \|x\|_1$

Método do subgradiente

- $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa não diferenciável

Método do subgradiente

- $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa não diferenciável

O método é similar ao método do gradiente, mas substituindo gradientes por subgradientes: escolha $x_0 \in \mathbb{R}^n$ e itere

$$x_{k+1} = x_k - \alpha_k v_k, \quad k = 0, 1, 2, \ldots,$$

Método do subgradiente

- $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa não diferenciável

O método é similar ao método do gradiente, mas substituindo gradientes por subgradientes: escolha $x_0 \in \mathbb{R}^n$ e itere

$$x_{k+1} = x_k - \alpha_k v_k, \quad k = 0, 1, 2, \ldots,$$

- v_k é subgradiente de f em x_k
- $\alpha_k > 0$

Método do subgradiente

- $f : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa não diferenciável

O método é similar ao método do gradiente, mas substituindo gradientes por subgradientes: escolha $x_0 \in \mathbb{R}^n$ e itere

$$x_{k+1} = x_k - \alpha_k v_k, \quad k = 0, 1, 2, \ldots,$$

- v_k é subgradiente de f em x_k
- $\alpha_k > 0$

Diferença com o método do gradiente: as escolhas do passo são prefixadas, não calculadas em cada iteração.

- Constante: $\alpha_k = \alpha$ para todo $k \in \mathbb{N}$.
- Quadrado somável, mas não somável:

$$\alpha_k > 0, \qquad \sum_{k=0}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=0}^{\infty} \alpha_k = \infty.$$

Complexidade

O método do subgradiente não é necessariamente um método de decida, logo guardamos em cada iteração o melhor iterado

$$f(x_k^{best}) = \min_{i=1,\ldots,k} f(x_k)$$

Complexidade

O método do subgradiente não é necessariamente um método de decida, logo guardamos em cada iteração o melhor iterado

$$f(x_k^{best}) = \min_{i=1,\ldots,k} f(x_k)$$

- Para obter $f(x_k^{best}) - f^* \leq \epsilon$ é necessário $\mathcal{O}(1/\epsilon^2)$ iterações.

Complexidade

O método do subgradiente não é necessariamente um método de decida, logo guardamos em cada iteração o melhor iterado

$$f(x_k^{best}) = \min_{i=1,\ldots,k} f(x_k)$$

♣ Para obter $f(x_k^{best}) - f^* \leq \epsilon$ é necessário $\mathcal{O}(1/\epsilon^2)$ iterações.

Método do gradiente: para obter $f(x_k) - f^* \leq 1/100 \Rightarrow$ 100 iterações.

Método do subgradiente: para obter $f(x_k^{best}) - f^* \leq 1/100 \Rightarrow$ 10000 iterações.

Vantagens:

- Ampla aplicação: se podemos computar subgradientes, então podemos minimizar (quase) toda função convexa.
- Fácil implementação.

Vantagens:

- Ampla aplicação: se podemos computar subgradientes, então podemos minimizar (quase) toda função convexa.
- Fácil implementação.

Desvantagens:

- A taxa de convergência $\mathcal{O}(1/\epsilon^2)$ é lenta.

Vantagens:

- Ampla aplicação: se podemos computar subgradientes, então podemos minimizar (quase) toda função convexa.
- Fácil implementação.

Desvantagens:

- A taxa de convergência $\mathcal{O}(1/\epsilon^2)$ é lenta.

Pode-se melhorar?

Vantagens:

- Ampla aplicação: se podemos computar subgradientes, então podemos minimizar (quase) toda função convexa.
- Fácil implementação.

Desvantagens:

- A taxa de convergência $\mathcal{O}(1/\epsilon^2)$ é lenta.

Pode-se melhorar? Em geral não! [Nes04]

Mas considerando funções da forma

$$f(x) = h(x) + g(x)$$

onde h é convexa e diferenciável e g é convexa, a complexidade $\mathcal{O}(1/\epsilon)$ do método do gradiente pode ser recuperada com um algoritmo simples.

Minimização da soma de duas funções convexas

$$\min_{x \in \mathbb{R}^n} h(x) + g(x)$$

- $h, g : \mathbb{R}^n \to \mathbb{R}$ são funções convexas e h é diferenciável.

Minimização da soma de duas funções convexas

$$\min_{x \in \mathbb{R}^n} h(x) + g(x)$$

- $h, g : \mathbb{R}^n \to \mathbb{R}$ são funções convexas e h é diferenciável.

Operador proximal

Dado $\alpha > 0$, $\mathbf{prox}_{\alpha g} : \mathbb{R}^n \to \mathbb{R}^n$

$$\mathbf{prox}_{\alpha g}(z) := \arg\min_{x \in \mathbb{R}^n} \alpha g(x) + \frac{1}{2} \|x - z\|^2$$

Minimização da soma de duas funções convexas

$$\min_{x \in \mathbb{R}^n} h(x) + g(x)$$

- $h, g : \mathbb{R}^n \to \mathbb{R}$ são funções convexas e h é diferenciável.

Operador proximal

Dado $\alpha > 0$, $\mathbf{prox}_{\alpha g} : \mathbb{R}^n \to \mathbb{R}^n$

$$\mathbf{prox}_{\alpha g}(z) := \arg\min_{x \in \mathbb{R}^n} \alpha g(x) + \frac{1}{2} \|x - z\|^2$$

- Se C é um conjunto convexo e fechado e $g = \delta_C$, então $\mathbf{prox}_g(z) = P_C(z)$

Método do gradiente proximal (MGP)

- ∇h é Lipschitz contínuo com $L > 0$
- $\mathbf{prox}_{\alpha g}$ pode ser avaliado

Método do gradiente proximal (MGP)

- ∇h é Lipschitz contínuo com $L > 0$
- $\mathbf{prox}_{\alpha g}$ pode ser avaliado

MGP gera uma sequência (x_k) a partir de um ponto inicial $x_0 \in \mathbb{R}^n$

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k)), \quad k = 0, 1, \ldots$$

- $\alpha_k > 0$

Método do gradiente proximal (MGP)

- ∇h é Lipschitz contínuo com $L > 0$
- $\mathbf{prox}_{\alpha g}$ pode ser avaliado

MGP gera uma sequência (x_k) a partir de um ponto inicial $x_0 \in \mathbb{R}^n$

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k)), \quad k = 0, 1, \ldots$$

- $\alpha_k > 0$

- Se $g \equiv 0$ o MGP se reduz ao método do gradiente
- Se $g = \delta_C$ o MGP se reduz ao método do gradiente projetado
- Se $h \equiv 0$ o MGP se reduz ao método de ponto proximal

- Usando passo constante $\alpha \leq 1/L$, o método do gradiente proximal tem complexidade $\mathcal{O}(1/\epsilon)$ ([BeT09], [MoS10]).

- Usando passo constante $\alpha \leq 1/L$, o método do gradiente proximal tem complexidade $\mathcal{O}(1/\epsilon)$ ([BeT09], [MoS10]).
- É recuperada a complexidade do método do gradiente, mas tem o custo da avaliação do $\mathbf{prox}_{\alpha g}$.

- Usando passo constante $\alpha \leq 1/L$, o método do gradiente proximal tem complexidade $\mathcal{O}(1/\epsilon)$ ([BeT09], [MoS10]).
- É recuperada a complexidade do método do gradiente, mas tem o custo da avaliação do $\mathbf{prox}_{\alpha g}$.

Para muitas funções g, importantes na prática, o $\mathbf{prox}_{\alpha g}$ tem forma fechada,

- Usando passo constante $\alpha \leq 1/L$, o método do gradiente proximal tem complexidade $\mathcal{O}(1/\epsilon)$ ([BeT09], [MoS10]).
- É recuperada a complexidade do método do gradiente, mas tem o custo da avaliação do $\mathbf{prox}_{\alpha g}$.

Para muitas funções g, importantes na prática, o $\mathbf{prox}_{\alpha g}$ tem forma fechada, mas e se não podemos avaliar **prox**?

Avaliar $\mathbf{prox}_{\alpha g}(z) = \arg\min_{x \in \mathbb{R}^n} \alpha g(x) + 1/2 \left\| x - z \right\|^2$ é equivalente a encontrar x, $w \in \mathbb{R}^n$ tais que w é um subgradiente de g em x e

$$\alpha w + x - z = 0$$

Avaliar $\mathbf{prox}_{\alpha g}(z) = \arg\min_{x \in \mathbb{R}^n} \alpha g(x) + 1/2 \|x - z\|^2$ é equivalente a encontrar x, $w \in \mathbb{R}^n$ tais que w é um subgradiente de g em x e

$$\alpha w + x - z = 0$$

Soluções aproximadas:

- Dado $r \geq 0$, um ponto (x, w) é uma r-**solução aproximada** de $\mathbf{prox}_{\alpha g}(z)$ se w é subgradiente de g em x e

$$\|\alpha w + x - z\| \leq r.$$

- Dado $\sigma \in [0, 1)$, o par (x, w) é uma σ-**solução aproximada** de $\mathbf{prox}_{\alpha g}(z)$ se w é subgradiente de g em x e

$$\|\alpha w + x - z\| \leq \sigma \|x - z\|.$$

[SoS99]

Método do gradiente proximal inexato

Similar ao método do gradiente proximal mas avaliando $\mathbf{prox}_{\alpha g}$ de forma inexata.

Método do gradiente proximal inexato

Similar ao método do gradiente proximal mas avaliando $\mathbf{prox}_{\alpha g}$ de forma inexata.

Escolha $x_0 \in \mathbb{R}^n$ e itera para $k = 0, 1, 2, \ldots$

- $y_k = x_k - \alpha_k \nabla h(x_k)$;
- calcule $(\overline{x}_k, \overline{w}_k)$ solução aproximada de $\mathbf{prox}_{\alpha_k g}(y_k)$;
- $x_{k+1} = y_k - \alpha_k \overline{w}_k$

Método do gradiente proximal inexato

Similar ao método do gradiente proximal mas avaliando $\mathbf{prox}_{\alpha g}$ de forma inexata.

Escolha $x_0 \in \mathbb{R}^n$ e itera para $k = 0, 1, 2, \ldots$

- $y_k = x_k - \alpha_k \nabla h(x_k)$;
- calcule $(\overline{x}_k, \overline{w}_k)$ solução aproximada de $\mathbf{prox}_{\alpha_k g}(y_k)$;
- $x_{k+1} = y_k - \alpha_k \overline{w}_k$

- MGPI-A: $(\overline{x}_k, \overline{w}_k)$ é uma r_k-solução aproximada para todo $k = 1, 2, \ldots$. Neste caso a sequência $\{r_k\}$ deve ser fixada com antecedência.
- MGPI-R: $(\overline{x}_k, \overline{w}_k)$ é uma σ-solução aproximada para todo $k = 1, 2, \ldots$, com $\sigma \in (0, 1)$ fixo.

[MiM19]

Generalizações do MGP

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k))$$

Generalizações do MGP

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k))$$

- $\nabla h(x_k) \to v_k$ subgradiente [Ber15], [Bel16]

Generalizações do MGP

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k))$$

- $\nabla h(x_k) \to v_k$ subgradiente [Ber15], [Bel16]
- $\nabla h(x_k) \to v_k$ ϵ_k-subgradiente [MiM19]

Generalizações do MGP

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k))$$

- $\nabla h(x_k) \to v_k$ subgradiente [Ber15], [Bel16]
- $\nabla h(x_k) \to v_k$ ϵ_k-subgradiente [MiM19]
- Avaliação aproximada do operador proximal
  - $\nabla h(x_k) \to \nabla h(x_k) + e_k$ e erro absoluto [Sch11]

Generalizações do MGP

$$x_{k+1} = \mathbf{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k))$$

- $\nabla h(x_k) \to v_k$ subgradiente [Ber15], [Bel16]
- $\nabla h(x_k) \to v_k$ ϵ_k-subgradiente [MiM19]
- Avaliação aproximada do operador proximal
  - $\nabla h(x_k) \to \nabla h(x_k) + e_k$ e erro absoluto [Sch11]
  - $\nabla h(x_k) \to v_k$ ϵ_k-subgradiente, erro absoluto e erro relativo [MiM19]

Métodos acelerados

Os métodos do gradiente e do gradiente proximal podem ser acelerados para obter complexidade $\mathcal{O}(1/\sqrt{\epsilon})$.

Métodos acelerados

Os métodos do gradiente e do gradiente proximal podem ser acelerados para obter complexidade $\mathcal{O}(1/\sqrt{\epsilon})$.

Método do gradiente: para obter $f(x_k) - f^* \leq 1/100 \Rightarrow$ 100 iterações.

Método do subgradiente: para obter $f(x_k^{best}) - f^* \leq 1/100 \Rightarrow$ 10000 iterações.

Método do gradiente acelerado: para obter $f(x_k) - f^* \leq 1/100 \Rightarrow$ 10 iterações

- Método do gradiente acelerado
  - [Nes83], [Nes04]

- Método do gradiente acelerado
  - [Nes83], [Nes04]
- Método do gradiente proximal acelerado
  - [Nes13], [BeT09] (FISTA)

- Método do gradiente acelerado
  - [Nes83], [Nes04]
- Método do gradiente proximal acelerado
  - [Nes13], [BeT09] (FISTA)
  - [MoS13]

- Método do gradiente acelerado
  - [Nes83], [Nes04]
- Método do gradiente proximal acelerado
  - [Nes13], [BeT09] (FISTA)
  - [MoS13]
- Método do gradiente proximal acelerado com cálculo aproximado do operador proximal
  - erro absoluto e aproximação do gradiente [Sch11]
  - erro absoluto [Vil13]
  - erro relativo [MiM19], [Bel20]

AMERICAN MATHEMATICAL SOCIETY
MATHSCINET
MATHEMATICAL REVIEWS

Home Resources Reviewers Free Tools Help Pages

Universidade Federal da Bahia

Publications Authors Journals Series Search MSC

Show Classic Interface

A method for solving the convex programming problem with convergence rate O(1/k2)

Show Search History

Show All Fields

Search Results

Filters Newest Export

20 First Prev 1 2 3 4 ... Next

837 results

MR4634221 - On the strong convergence of continuous Newton-like inertial dynamics with Tikhonov regularization for monotone inclusions
Boţ, Radu Ioan; Csetnek, Ernö Robert; László, Szilárd Csaba
J. Math. Anal. Appl. **530** (2024), no. 2, Paper No. 127689, 30 pp.
Pending
MSC 49J53
Article

MR4646887 - Zero-norm regularized problems: equivalent surrogates, proximal MM method and statistical error bound
Zhang, Dongdong; Pan, Shaohua; Bi, Shujun; Sun, Defeng
Comput. Optim. Appl. **86** (2023), no. 2, 627–667.
Pending
MSC 90C27
Article

MR4646881 - An accelerated proximal gradient method for multiobjective optimization
Tanabe, Hiroki; Fukuda, Ellen H.; Yamashita, Nobuo
Pending
1 citations

AMERICAN MATHEMATICAL SOCIETY
MATHSCINET
MATHEMATICAL REVIEWS

Home Resources Reviewers Free Tools Help Pages

Universidade Federal da Bahia

Publications Authors Journals Series Search MSC

Show Classic Interface

A fast iterative shrinkage-thresholding algorithm for linear inverse problems

Show Search History

Show All Fields

Search Results

Filters Newest Export

20 First Prev 1 2 3 4 ... Next

1.406 results

MR4623936 - Multi-step inertial strictly contractive PRSM algorithms for convex programming problems with applications
Deng, Zhao; Han, Deren
J. Comput. Appl. Math. **437** (2024), Paper No. 115469, 17 pp.
Pending
MSC 65M12
Article

MR4646884 - Doubly majorized algorithm for sparsity-inducing optimization problems with regularizer-compatible constraints
Liu, Tianxiang; Pong, Ting Kei; Takeda, Akiko
Comput. Optim. Appl. **86** (2023), no. 2, 521–553.
Pending
MSC 90C26
Article

MR4646881 - An accelerated proximal gradient method for multiobjective optimization
Tanabe, Hiroki; Fukuda, Ellen H.; Yamashita, Nobuo
Pending
1 citations

Obrigada!

Referências:

[BeT09] Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci.2(1), 183–202 (2009)

[Bel16] Bello Cruz, J. Y.: On proximal subgradient splitting method for minimizing the sum of two nonsmooth convex functions. Set-Valued and Variational Analysis (2016)

[Bel20] Bello Cruz, Y.; Gonçalves, M.L.N.; Krislock, N.: On inexact accelerated proximal gradient methods with relative error rules, (2020)

[Ber09] Bertsekas, D. P.: Convex Optimization Theory. Athena Scientific, 2009

[Ber15] Bertsekas, D. P.: Incremental gradient, subgradient, and proximal methods for convex optimization: A survey. CoRR abs/1507.01030 (2015)

[IzS12] A. Izmailov, M. Solodov. Otimização, Volume 2: Métodos Computacionais. Rio de Janeiro, Brazil, Segunda Edição (2012).

[MiM19] Millán, R.D. and Machado, M.P.: Inexact proximal ϵ-subgradient methods for composite convex optimization problems. J Glob Optim (2019). https://doi.org/10.1007/s10898-019-00808-8

[MoS10] Monteiro, R.D.C.; Svaiter, B.F.: Convergence rate of inexact proximal point methods with relative error criteria for convex optimization. Optimization Online (2010)

[MoS13] Monteiro, R.D.C., Svaiter, B.F.: An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. SIAM J. Optim.23(2), 1092–1125 (2013)

[Nes83] Nesterov, Y.: A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Sov.Math. Dokl.27, 372–376 (1983)

[Nes04] Nesterov, Y.: Introductory Lectures on Convex Optimization. Kluwer, Boston, 2004

[Nes13] Nesterov, Y.: Gradient methods for minimizing composite functions. Math. Program. Ser. B140(1),125–161 (2013)

[Sch11] Schmidt, M., Le Roux, N., Bach, F.: Convergence rates of inexact proximal-gradient methods for convex optimization. In: NIPS'11—25th Annual Conference on Neural Information Processing Systems (Grenada, Spain, Dec. 2011)

[SoS99] Solodov, M.V., Svaiter, B.F.: A hybrid approximate extragradient-proximal point algorithm using theenlargement of a maximal monotone operator. Set-Valued Anal.7(4), 323–345 (1999)

[Vil13] Villa, S., Salzo, S., Baldassarre, L., Verri, A.: Accelerated and inexact forward–backward algorithms. JSIAM J. Optim.23(3), 1607–1633 (2013)