UNIVERSIDADE FEDERAL DO PARANÁ
Adriano Rodrigo Delfino

UM MÉTODO ÓTIMO PARA OTIMIZAÇÃO CONVEXA IRRESTRITA

Curitiba, 2010.

UNIVERSIDADE FEDERAL DO PARANÁ
Adriano Rodrigo Delfino

UM MÉTODO ÓTIMO PARA OTIMIZAÇÃO CONVEXA IRRESTRITA

Dissertação apresentada ao Programa de Pós-Graduação em Matemática Aplicada da Universidade Federal do Paraná, como requisito parcial à obtenção do grau de Mestre em Matemática Aplicada.

Orientador: Profa. Dra. Elizabeth Wegner Karas.

Curitiba, 2010.

A Deus
e a minha família
pelo apoio incondicional.

Agradecimentos

Gostaria de agradecer primeiramente a Deus, por ter me dado vida e ter me conduzido até aqui.

Gostaria de agradecer também a minha orientadora, profa . Dra. Elizabeth Wegner Karas, pela paciência, dedicação e destreza com a qual conduziu esta orientação de mestrado, sempre estando presente e me motivando muito. Gostaria de agradecer aos comentários e sugestões valiosas proposta pela banca composta por prof . Dr . Ademir Alves Ribeiro, prof . Dr Clóvis Caesar Gonzaga e prof . Dr . Paulo José da Silva e Silva. Um especial agradecimento a profa . Dra . Soraya Rosana Torres Kudri, que desde a graduação tem me orientado e motivado em relação aos estudos. Em seguida, agradeço aos professores do Programa de Pós-Graduação em Matemática Aplicada da UFPR e aos demais professores do Departamento de Matemática da UFPR.

Agradeço também aos meus colegas de Pós-Graduação, por todos os que de alguma maneira ou outra contribuiram para esse trabalho. Agradeço também aos amigos, que me apoiaram até o presente momento. Em especial, um agradecimento a três amigos, Alberto Levi, Alessandro Gaio e Helder Geovane pela contribuição valiosa em todos os conhecimentos que adquiri, desde a Graduação e principalmente na Pós-Graduação.

Enfim, registro um agradecimento especial a meus familiares. Minha mãe Sonia, meu pai José, meus irmãos Alessandro e Alisson pelo apoio em toda a minha vida acadêmica. Pontuo um agradecimento a todos os familiares que de alguma maneira confiaram e apoiaram meus estudos, sabendo que o retorno viria a médio e a longo prazo.

"Só há felicidade se não exigirmos nada
do amanhã e aceitarmos do hoje, com gratidão,
o que nos trouxer. A hora mágica chega sempre."

Hermann Hesse

"Uma das trágicas coisas que eu percebo
na natureza humana é que todos nós tendemos
a adiar o viver. Estamos todos sonhando
com um mágico jardim de rosas no horizonte,
ao invés de desfrutar das rosas que estão
florescendo do lado de fora de nossas janelas hoje."

Dale Carnegie

Resumo

Esse trabalho é dedicado ao estudo de complexidade do ponto de vista de Nesterov [Nes04] para métodos de primeira ordem, ou seja, que usam apenas informação de valor de função ou de seu gradiente. Em relação a complexidade, há diversas maneiras de obte-la, seja contando o tempo computacional gasto para resolver o problema, o número de operações aritméticas usado pelo método, entre outros. No nosso caso, será contando o número de iterações gasto pelo método para resolver o problema.

Nos nossos problemas, as funções objetivos pertencem a classe das funções convexas, continuamente diferenciáveis e com constante de Lipschitz L para o gradiente. Estudamos a complexidade ótima desses métodos e provamos a complexidade ótima de um método apresentado em [GK08] para essa classe de funções. Fizemos também alguns testes númericos com alguns métodos ótimos propostos na literatura.

Palavras-chave: *Problemas convexos irrestrito, Complexidade, Método Ótimo.*

Abstract

This work is dedicated to the study of complexity in terms of Nesterov proposed in [Nes04] for methods of first order, ie, using only information of value function or its gradient. For complexity, there are several ways to obtain it, by counting the computational time to solve the problem, the number of arithmetic operations used by the method, between others. In our case, will be counting the number of iterations spent by the method to solve the problem.

In our problems, the objective functions belong to the class of convex functions, continuously differentiable and with constant Lipschitz L for the gradient. We study the optimal complexity of these methods and prove the optimal complexity of a method presented in [GK08] for this class of functions. We also present some numerical tests with some optimal methods proposed in the literature.

Keywords: *Unconstrained convex problems, Complexity, Optimal method.*

Sumário

Capítulo 1

Complexidade algorítmica para a classe de funções convexas

A análise de complexidade clássica da Ciência da Computação está relacionada geralmente com a dimensão do espaço do problema. Nesse trabalho, discutimos o conceito de complexidade, segundo Nesterov [Nes04], que está relacionado com a precisão com que queremos resolver uma classe de problemas.

O objetivo desse capítulo é estabelecer o conceito de complexidade algorítmica de um problema de otimização, bem como, discutir a complexidade algorítmica dos problemas de minimização de uma função convexa ou fortemente convexa usando métodos que utilizam informações pontuais de até primeira ordem de função.

1.1 Complexidade

Considere o problema de minimizar em um conjunto $S \subset \mathbb{R}^n$ uma função $f : \mathbb{R}^n \to \mathbb{R}$ pertencente a uma classe de funções $\mathscr{C}$ e considere f^* seu valor mínimo em S. Resolver o problema significa encontrar uma solução aproximada $\bar{x} \in S$, isto é, dada uma tolerância $\varepsilon > 0$, encontrar $\bar{x}$ tal que $f(\bar{x}) - f^* < \varepsilon$.

Para resolver o problema podemos usar um método[1] baseado no *conceito de caixa preta* [Weg05], ou seja, supomos que não conhecemos necessariamente uma forma fechada da função objetivo f, somente podemos contar com informações de valores pontuais fornecidos por uma caixa preta, ou seja, esses valores são fornecidos de alguma maneira que não estamos interessados em saber como são obtidos.

Quando usamos um determinado método para resolver o problema não podemos contar com uma expressão da função f e sim com valores pontuais solicitados pelo método através

[1] algoritmo

de chamadas de um *oráculo* cujas respostas são fornecidas por uma caixa preta. Se o oráculo é de ordem zero podemos contar apenas com valores pontuais de f. Oráculos de primeira ordem retornam valores pontuais de f e de seu gradiente. Analogamente, um oráculo de segunda ordem pode fornecer informações pontuais de f, de seu gradiente e de sua matriz Hessiana.

O desempenho de um método para resolver um problema pode ser medido pelo número de chamadas do oráculo (complexidade analítica) ou ainda pelo número de operações aritméticas (complexidade aritmética) exigido pelo método para resolver o problema com precisão ε.

Dependendo do algoritmo, é possível estimar o número de chamadas do oráculo por iteração. Nesse caso, a complexidade do algoritmo pode ser analisada a partir do número de iterações.

Ao longo desse trabalho usaremos a seguinte notação.

Notação: Dadas duas funções $g_1, g_2 : X \subset \mathbb{R}^n \to \mathbb{R}_+$, dizemos que $g_1(x) = \mathcal{O}(g_2(x))$ (ou equivalentemente $g_2(x) = \Omega(g_1(x))$) em $\Gamma \subseteq X$ se existe $M > 0$ tal que para todo $x \in \Gamma$, $g_1(x) \leq M g_2(x)$.

Analisamos a seguir a cota de complexidade inferior e superior.

Cota de complexidade inferior

A cota de complexidade inferior está associada a uma classe de funções. Considere uma função $g : \mathbb{R}_+ \to \mathbb{R}_+$. Dizemos que a cota de complexidade inferior de uma classe $\mathscr{C}$ de funções é $\Omega(g(\varepsilon))$ quando para todo método M de uma classe $\mathscr{M}$ de métodos usados para minimizar, com precisão ε, toda função de $\mathscr{C}$, existe uma função $\bar{f} \in \mathscr{C}$ para qual o método gastará $\Omega(g(\varepsilon))$ iterações.

Vemos assim que a complexidade inferior relaciona-se com o pior caso, ou seja, com uma função difícil de ser minimizada que pode diferir de método para método. Note que a complexidade depende da precisão ε com que queremos resolver os problemas.

Cota de complexidade superior

Já a cota superior, por outro lado, está relacionada a um método M usado para minimizar uma classe de funções.

Considere uma função $h : \mathbb{R}_+ \to \mathbb{R}_+$. Fixado um determinado método M, dizemos que a cota de complexidade superior é $\mathcal{O}(h(\varepsilon))$ quando ele minimiza, com precisão ε, toda função da classe $\mathscr{C}$ em $\mathcal{O}(h(\varepsilon))$ iterações.

Método ótimo

Um método M é ótimo para uma classe $\mathscr{C}$ de funções quando a sua cota de complexidade superior é proporcional à cota de complexidade inferior da classe $\mathscr{C}$, em outras palavras, quando as funções g e $h : \mathbb{R}_+ \to \mathbb{R}_+$, consideradas acima, são proporcionais.

1.2 Funções convexas

Nesse trabalho concentraremos nossa atenção nos problemas de minimização de funções $f : \mathbb{R}^n \to \mathbb{R}$ pertencentes à classe das funções diferenciáveis convexas ou fortemente convexas. Nessa seção trabalharemos com funções convexas. As principais referências dessa seção são [Nes04, HUL93a, HUL93b, Roc70].

Definição 1.1. *Uma função continuamente diferenciável $f : \mathbb{R}^n \to \mathbb{R}$ é chamada convexa se para todo $x, y \in \mathbb{R}^n$ temos*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle. \tag{1.1}$$

Se $-f$ é convexa, dizemos que f é côncava.

Assumimos diferenciabilidade na definição acima de convexidade, pois esse é o caso de interesse em nosso trabalho. É possível definir função convexa sem essa hipótese e obter (1.1) como consequência [HUL93a, Teorema 4.1.1].

1.2.1 Propriedades

Essa seção é dedicada ao estudo de algumas propriedades de funções convexas que serão utilizadas ao longo do texto.

Lema 1.2. *Considere $f : \mathbb{R}^n \to \mathbb{R}$ convexa. Se $\nabla f(x^*) = 0$ então x^* é minimizador global de f em $\mathbb{R}^n$.*

Demonstração. Considere $x \in \mathbb{R}^n$ arbitrário. Pela definição de convexidade, temos

$$f(x) \geq f(x^*) + \langle \nabla f(x^*), x - x^* \rangle.$$

Usando a hipótese, concluímos que

$$f(x) \geq f(x^*).$$

Como x foi tomado arbitrário, completamos a demonstração. □

O lema a seguir garante que uma combinação linear de funções convexas nos fornece uma função convexa.

Lema 1.3. *Se f_1, f_2 são funções convexas e $\alpha, \beta \geq 0$, então $f = \alpha f_1 + \beta f_2$ é uma função convexa.*

Demonstração. Considere $x, y \in \mathbb{R}^n$ arbitrários. Pela definição de convexidade

$$f_1(y) \geq f_1(x) + \langle \nabla f_1(x), y - x \rangle,$$

e

$$f_2(y) \geq f_2(x) + \langle \nabla f_2(x), y - x \rangle.$$

Multiplicando a primeira desigualdade por α e a segunda por β obtemos:

$$f(y) = \alpha f_1(y) + \beta f_2(y) \geq \alpha f_1(x) + \beta f_2(x) + \langle \alpha \nabla f_1(x) + \beta \nabla f_2(x), y - x \rangle =$$

$$= f(x) + \langle \nabla f(x), y - x \rangle,$$

que é o resultado desejado. □

Uma maneira alternativa para a caracterização da convexidade de uma função é dada pelo lema abaixo.

Lema 1.4. *Uma função continuamente diferenciável f é convexa se e somente se para todo $x, y \in \mathbb{R}^n$ temos*

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq 0. \tag{1.2}$$

Demonstração. Seja f uma função convexa. Considere x e $y \in \mathbb{R}^n$ arbitrários. Pela convexidade de f temos:

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle,$$

e

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle.$$

Somando as duas desigualdades obtemos (1.2).

Reciprocamente, considere x e $y \in \mathbb{R}^n$ arbitrários e denote $x_\tau = x + \tau(y - x)$.

Então,

$$\begin{aligned}
f(y) &= f(x)+\int_0^1 \langle \nabla f(x+\tau(y-x)), y-x\rangle d\tau \\
&= f(x)+\langle \nabla f(x), y-x\rangle - \langle \nabla f(x), y-x\rangle + \int_0^1 \langle \nabla f(x_\tau), y-x\rangle d\tau \\
&= f(x)+\langle \nabla f(x), y-x\rangle + \int_0^1 \langle \nabla f(x_\tau)-\nabla f(x), y-x\rangle d\tau \\
&= f(x)+\langle \nabla f(x), y-x\rangle + \int_0^1 \langle \nabla f(x_\tau)-\nabla f(x), \frac{\tau}{\tau}(y-x)+x-x\rangle d\tau \\
&= f(x)+\langle \nabla f(x), y-x\rangle + \int_0^1 \frac{1}{\tau}\langle \nabla f(x_\tau)-\nabla f(x), x_\tau - x\rangle d\tau \\
&\geq f(x)+\langle \nabla f(x), y-x\rangle,
\end{aligned}$$

o que mostra que f é convexa. □

Considere matrizes simétricas A e $B \in \mathbb{R}^{n\times n}$. A notação $A \geq B$ indica que a matriz $A-B$ é semi definida positiva. Indicamos a matriz identidade $n \times n$ por I. Uma outra maneira de verificar se uma função f duas vezes diferenciável é convexa é verificando se sua hessiana é semi definida positiva, como mostra o lema abaixo.

Lema 1.5. *Uma função f duas vezes diferenciável é uma função convexa se e somente se para todo $x \in \mathbb{R}^n$ temos*

$$\nabla^2 f(x) \geq 0. \tag{1.3}$$

Demonstração. Considere x e y arbitrários. Usando a aproximação de Taylor de segunda ordem temos que existe $z = \alpha x + (1-\alpha)y$, com $\alpha \in (0,1)$, tal que

$$f(y) = f(x)+\langle \nabla f(x), y-x\rangle + \frac{1}{2}\langle \nabla^2 f(z)(y-x), y-x\rangle.$$

Se vale $\nabla^2 f(x) \geq 0$ para todo $x \in \mathbb{R}^n$, temos que $\frac{1}{2}\langle \nabla^2 f(z)(y-x), y-x\rangle \geq 0$. Logo

$$f(y) \geq f(x)+\langle \nabla f(x), y-x\rangle.$$

Reciprocamente, suponha por absurdo que existem $x \in \mathbb{R}^n$ e $d \in \mathbb{R}^n$ tais que

$$\langle \nabla^2 f(x)d, d\rangle < 0. \tag{1.4}$$

Considere a função $g:\mathbb{R}\to\mathbb{R}$ definida por $g(\lambda)=f(x+\lambda d)$. Pela regra da cadeia,

$$g''(\lambda)=\langle\nabla^2 f(x+\lambda d)d,d\rangle.$$

Usando isto e (1.4), temos que $g''(0)<0$. Pela conservação de sinal, existe $\delta>0$ tal que $g''(t)<0$ para todo $t\in(-\delta,\delta)$. Considere $y=x+\frac{\delta}{2}d$. Por Taylor, existe $z=x+\alpha(y-x)=x+\alpha\frac{\delta}{2}d$, com $\alpha\in(0,1)$ tal que

$$\begin{aligned} f(y) &= f(x)+\langle\nabla f(x),y-x\rangle+\tfrac{1}{2}\langle\nabla^2 f(z)(y-x),y-x\rangle \\ &= f(x)+\langle\nabla f(x),y-x\rangle+\tfrac{\delta^2}{8}\langle\nabla^2 f(x+\alpha\tfrac{\delta}{2}d)d,d\rangle. \end{aligned}$$

Mas $\frac{\alpha\delta}{2}\in(-\delta,\delta)$. Logo $g''(\frac{\alpha\delta}{2})<0$, o que implica

$$f(y)<f(x)+\langle\nabla f(x),y-x\rangle,$$

o que contradiz o fato de f ser convexa, completando a demonstração. □

Estaremos particularmente interessados nas funções diferenciáveis convexas com gradiente Lipschitz, isto é, existe $L>0$ tal que para todo $x,y\in\mathbb{R}^n$, vale

$$\|\nabla f(x)-\nabla f(y)\|\le L\|x-y\|. \tag{1.5}$$

Lema 1.6. *Considere $f:\mathbb{R}^n\to\mathbb{R}$ uma função diferenciável com constante de Lipschitz $L>0$ para o gradiente. Então para todo $x,y\in\mathbb{R}^n$ temos*

$$|f(y)-f(x)-\langle\nabla f(x),y-x\rangle|\le\frac{L}{2}\|y-x\|^2. \tag{1.6}$$

Demonstração. Para todo $x,y\in\mathbb{R}^n$ temos

$$f(y)=f(x)+\int_0^1\langle\nabla f(x+\tau(y-x)),y-x\rangle d\tau. \tag{1.7}$$

Somando e subtraindo $\langle\nabla f(x),y-x\rangle$ em (1.7) obtemos

$$f(y)-f(x)-\langle\nabla f(x),y-x\rangle=\int_0^1\langle\nabla f(x+\tau(y-x))-\nabla f(x),y-x\rangle d\tau.$$

Com isso

$$\begin{aligned}
|f(y)-f(x)-\langle\nabla f(x),y-x\rangle| &= \left|\int_0^1 \langle\nabla f(x+\tau(y-x))-\nabla f(x),y-x\rangle d\tau\right| \\
&\leq \int_0^1 |\langle\nabla f(x+\tau(y-x))-\nabla f(x),y-x\rangle| d\tau \\
&\leq \int_0^1 \|\nabla f(x+\tau(y-x))-\nabla f(x)\|\|y-x\| d\tau \\
&\leq \int_0^1 L\|x+\tau(y-x)-x\|\|y-x\| d\tau \\
&= \int_0^1 \tau L\|y-x\|^2 d\tau = \frac{L}{2}\|y-x\|^2.
\end{aligned}$$

□

Usaremos a letra $\mathscr{F}$ para denotar a classe das funções diferenciáveis convexas que satisfazem (1.5).

Teorema 1.7. *Considere $x, y \in \mathbb{R}^n$ arbitrários. As afirmações abaixo são equivalentes a $f \in \mathscr{F}$:*

$$0 \leq f(y)-f(x)-\langle\nabla f(x),y-x\rangle \leq \frac{L}{2}\|x-y\|^2, \tag{1.8}$$

$$f(x)+\langle\nabla f(x),y-x\rangle+\frac{1}{2L}\|\nabla f(x)-\nabla f(y)\|^2 \leq f(y), \tag{1.9}$$

$$\frac{1}{L}\|\nabla f(x)-\nabla f(y)\|^2 \leq \langle\nabla f(x)-\nabla f(y),x-y\rangle, \tag{1.10}$$

$$0 \leq \langle\nabla f(x)-\nabla f(y),x-y\rangle \leq L\|x-y\|^2. \tag{1.11}$$

Demonstração. A desigualdade (1.8) decorre da definição e do Lema 1.6.

Para provar (1.9) considere $x_0 \in \mathbb{R}^n$ arbitrário e a função $\phi(y) = f(y) - \langle\nabla f(x_0), y\rangle$. Vemos que $\phi \in \mathscr{F}$ pois

$$\|\nabla\phi(x)-\nabla\phi(y)\| = \|\nabla f(x)-\nabla f(y)\| \leq L\|x-y\|$$

e seu ponto ótimo é $y^* = x_0$.

Temos então $\phi(y^*) \leq \phi(z)$ para todo $z \in \mathbb{R}^n$, em particular para $z = y - \frac{1}{L}\nabla\phi(y)$. Logo

usando (1.8) temos

$$\phi\left(y-\frac{1}{L}\nabla\phi(y)\right)-\phi(y)-\left\langle\nabla\phi(y),\frac{-1}{L}\nabla\phi(y)\right\rangle\leq\frac{L}{2}\left\|\frac{1}{L}\nabla\phi(y)\right\|^2.$$

Com isso temos que

$$\phi(y^*)\leq\phi\left(y-\frac{1}{L}\nabla\phi(y)\right)\leq\phi(y)-\frac{1}{L}\left\|\nabla\phi(y)\right\|^2+\frac{1}{2L}\left\|\nabla\phi(y)\right\|^2=\phi(y)-\frac{1}{2L}\left\|\nabla\phi(y)\right\|^2.$$

Donde

$$f(x_0)-\langle\nabla f(x_0),x_0\rangle\leq f(y)-\langle\nabla f(x_0),y\rangle-\frac{1}{2L}\|\nabla f(y)-\nabla f(x_0)\|^2.$$

Como x_0 é arbitrário, temos

$$f(x)+\langle\nabla f(x),y-x\rangle+\frac{1}{2L}\|\nabla f(y)-\nabla f(x)\|^2\leq f(y),$$

demonstrando (1.9). Usando (1.9), temos:

$$f(x)+\langle\nabla f(x),y-x\rangle+\frac{1}{2L}\|\nabla f(x)-\nabla f(y)\|^2\leq f(y),$$

e

$$f(y)+\langle\nabla f(y),x-y\rangle+\frac{1}{2L}\|\nabla f(y)-\nabla f(x)\|^2\leq f(x).$$

Somando essas duas desigualdades obtemos (1.10).

Aplicando a desigualdade de Cauchy-Schwartz em (1.10) temos

$$\frac{1}{L}\|\nabla f(x)-\nabla f(y)\|^2\leq\langle\nabla f(x)-\nabla f(y),x-y\rangle\leq\|\nabla f(x)-\nabla f(y)\|\|x-y\|.$$

Simplificando, obtemos $\|\nabla f(x)-\nabla f(y)\|\leq L\|x-y\|$, demonstrando que $f\in\mathscr{F}$.

Demonstramos acima que

$$f\in\mathscr{F}\Rightarrow(1.8)\Rightarrow(1.9)\Rightarrow(1.10)\Rightarrow f\in\mathscr{F}.$$

Pela desigualdade (1.8), temos:

$$f(y)-f(x)-\langle\nabla f(x),y-x\rangle\leq\frac{L}{2}\|x-y\|^2,$$

e

$$f(x)-f(y)-\langle\nabla f(y),x-y\rangle\leq\frac{L}{2}\|x-y\|^2.$$

Somando essas duas desigualdades obtemos (1.11).

Reciprocamente, temos para todo $x, y \in \mathbb{R}^n$,

$$\begin{aligned} f(y) - f(x) - \langle \nabla f(x), y - x \rangle &= \int_0^1 \langle \nabla f(x + \tau(y-x)) - \nabla f(x), y - x \rangle d\tau \\ &= \int_0^1 \frac{1}{\tau} \langle \nabla f(x + \tau(y-x)) - \nabla f(x), \tau(y-x) \rangle d\tau. \quad (1.12) \end{aligned}$$

Usando (1.12) com (1.11) obtemos

$$\begin{aligned} f(y) - f(x) - \langle \nabla f(x), y - x \rangle &\leq \int_0^1 \frac{1}{\tau} L \|\tau(y-x)\|^2 d\tau \\ &= L\|y-x\|^2 \int_0^1 \tau \, d\tau = \frac{L}{2}\|y-x\|^2, \end{aligned}$$

completando a demonstração.

□

Teorema 1.8. *Uma função contínua duas vezes diferenciável f pertence à classe $\mathscr{F}$ se e somente se para todo $x \in \mathbb{R}^n$ temos*

$$0 \leq \nabla^2 f(x) \leq LI, \quad (1.13)$$

onde I é a matriz identidade $n \times n$.

Demonstração. A primeira desigualdade segue do Lema 1.5 e a segunda segue de (1.11). □

1.2.2 Cota de complexidade inferior

Nesterov mostra em [Nes04] que a cota inferior de complexidade relativa aos problemas com funções objetivo convexas é da ordem de $\frac{1}{\sqrt{\varepsilon}}$ para todo método de primeira ordem. Mostraremos nesta seção como ele faz isso.

Aqui denotaremos as coordenadas do vetor x por $z_1, z_2, \ldots, z_n$, isto é, $x = (z_1, z_2, \ldots, z_n) \in \mathbb{R}^n$ para não confundir com a notação de sequência $x_1, x_2, \ldots$ usada nesse trabalho.

Fixando $L > 0$, considere a seguinte família de funções quadráticas $f_k : \mathbb{R}^n \to \mathbb{R}$ dada por:

$$f_1(x) = \frac{L}{4}\left(z_1^2 - z_1\right),$$

$$f_2(x) = \frac{L}{4}\left(z_1^2 - z_1 z_2 + z_2^2 - z_1\right),$$

e assim por diante,

$$f_k(x) = \frac{L}{4}\left(\frac{1}{2}\left(z_1^2 + \sum_{i=1}^{k-1}(z_i - z_{i+1})^2 + z_k^2\right) - z_1\right), \tag{1.14}$$

para $k = 3, 4, \ldots, n$.

Calculando a derivada dessas funções temos:

$$\nabla f_1(x) = \frac{L}{4}(2z_1 - 1, 0, \ldots, 0),$$

$$\nabla f_2(x) = \frac{L}{4}(2z_1 - z_2 - 1, 2z_2 - z_1, 0, \ldots, 0),$$

ou seja,

$$\nabla f_k(x) = \frac{L}{4}[A_k x - e_1], \tag{1.15}$$

onde

$$A_k = \left(\begin{array}{c|c} \overset{k}{linhas}\left\{\begin{array}{ccccccc} 2 & -1 & 0 & & & & \\ -1 & 2 & -1 & & 0 & & \\ 0 & -1 & 2 & & & & \\ & \cdots & & & \cdots & & \\ & & & -1 & 2 & -1 & \\ & 0 & & 0 & -1 & 2 & \end{array}\right. & 0_{n-k,k} \\ \hline 0_{n-k,k} & 0_{n-k,n-k} \end{array}\right).$$

Note também que $\nabla^2 f_k(x) = \frac{L}{4}A_k$ para $k = 1, \ldots, n$.

Vemos que a hessiana é semi-definida positiva pois:

$$\begin{aligned} \langle \nabla^2 f_k(x)x, x\rangle &= \tfrac{L}{4}\begin{bmatrix} z_1 & z_2 & \ldots & z_n \end{bmatrix} A_k \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} \\ &= \tfrac{L}{4}\left[z_1^2 + \sum_{i=1}^{k-1}(z_i - z_{i+1})^2 + z_k^2\right] \geq 0. \end{aligned}$$

Por outro lado, usando a desigualdade $(a-b)^2 \leq 2(a^2+b^2)$, temos:

$$\begin{aligned}\langle \nabla^2 f_k(x)x, x\rangle &\leq \frac{L}{4}\left[z_1^2+\sum_{i=1}^{k-1}2(z_i^2+z_{i+1}^2)+z_k^2\right]\\ &\leq \frac{L}{4}\left[\sum_{i=1}^{n}z_i^2+2\sum_{i=1}^{n}z_i^2+\sum_{i=1}^{n}z_i^2\right]\\ &= L\sum_{i=1}^{n}z_i^2.\end{aligned}$$

Ou seja, $0 \leq \nabla^2 f_k(x) \leq LI_n$. Portanto pelo Teorema 1.8, $f_k(x) \in \mathscr{F}$, $1 \leq k \leq n$.

Podemos calcular o mínimo das funções f_k.

Começando com $k=1$, temos:

$$\nabla f_1(x) = 0 \Rightarrow (2z_1 - 1, 0, \ldots, 0) = 0,$$

resolvendo obtemos $z_1 = \frac{1}{2}$ e $z_i = 0,\ i = 2, \ldots, n$.

Para $k=2$, temos:

$$\nabla f_2(x) = 0 \Rightarrow (2z_1 - z_2 - 1, 2z_2 - z_1, 0, \ldots, 0) = 0,$$

resolvendo obtemos $z_1 = \frac{2}{3},\ z_2 = \frac{1}{3}$ e $z_i = 0,\ i = 3, \ldots, n$.

E assim por diante, obtemos:

$$\nabla f_k(x) = \frac{L}{4}[A_k x - e_1] = 0. \tag{1.16}$$

donde

$$\bar{x}_{k_i} = \begin{cases} 1 - \frac{i}{k+1}, & i = 1, \ldots, k \\ 0, & k+1 \leq i \leq n. \end{cases} \tag{1.17}$$

Logo o valor ótimo de f_k é dado por

$$f_k^* = f_k(\bar{x}_k) = \frac{L}{4}\left(\frac{1}{2}\left(\bar{z}_1^2 + \sum_{i=1}^{k-1}(\bar{z}_i - \bar{z}_{i+1})^2 + \bar{z}_k^2\right) - \bar{z}_1\right). \tag{1.18}$$

Escrevendo (1.18) em formato de produto interno, obtemos

$$f_k^* = \frac{L}{4}\left(\frac{1}{2}\langle A_k \bar{x}_k, \bar{x}_k\rangle - \langle e_1, \bar{x}_k\rangle\right). \tag{1.19}$$

Resolvendo (1.19) obtemos

$$f_k^* = \frac{L}{8}\left(-1+\frac{1}{k+1}\right). \tag{1.20}$$

Também podemos estimar a norma de $\bar{x}_k$:

$$\|\bar{x}_k\|^2 = \sum_{i=1}^{n}(\bar{x}_{k_i})^2 = \sum_{i=1}^{k}\left(1-\frac{i}{k+1}\right)^2. \tag{1.21}$$

Desenvolvendo o produto notável em (1.21) e simplificando obtemos:

$$\|\bar{x}_k\|^2 \leq k - \frac{2}{k+1}\frac{k(k+1)}{2} + \frac{1}{(k+1)^2}\frac{(k+1)^3}{3} = \frac{1}{3}(k+1). \tag{1.22}$$

Denotando $\mathbb{R}^{k,n} = \{x \in \mathbb{R}^n \mid z_i = 0, k+1 \leq i \leq n\} \subset \mathbb{R}^n$, temos que para todo $x \in \mathbb{R}^{k,n}$, $f_p(x) = f_k(x)$ onde $p = k, \ldots, n$.

Lema 1.9. *Seja $x_0 = 0$. Então para toda sequência satisfazendo a condição*

$$x_k \in \mathscr{L}_k = Lin\{\nabla f_p(x_0), \ldots, \nabla f_p(x_{k-1})\}, \tag{1.23}$$

temos $\mathscr{L}_k \subseteq \mathbb{R}^{k,n}$.

Demonstração. A prova é por indução. Para $k = 1$ temos que $x_1 \in \mathscr{L}_1 = Lin\{\nabla f_p(x_0)\}$. Mas $\nabla f_p(x_0) = -\frac{L}{4}e_1$. Donde $\nabla f_p(x_0) \in \mathbb{R}^{1,n}$. Concluímos então que $\mathscr{L}_1 \equiv \mathbb{R}^{1,n}$.

Supomos então que $\mathscr{L}_k \subseteq \mathbb{R}^{k,n}$ para algum $k < p$. Uma vez que A_p é tridiagonal, para todo $x \in \mathbb{R}^{k,n}$, temos $\nabla f_p(x) \in \mathbb{R}^{k+1,n}$. Logo $\mathscr{L}_{k+1} \subseteq \mathbb{R}^{k+1,n}$. □

Corolário 1.10. *Para toda sequência $\{x_k\}_{k=0}^p$ tal que $x_0 = 0$ e $x_k \in \mathscr{L}_k$, temos que*

$$f_p(x_k) \geq f_k^*. \tag{1.24}$$

Demonstração. Tome $x_k \in \mathscr{L}_k \subseteq \mathbb{R}^{k,n}$. Então $f_p(x_k) = f_k(x_k) \geq f_k^*$. □

O próximo teorema proporciona a cota de complexidade inferior de problemas de minimização de funções da classe $\mathscr{F}$ usando métodos de primeira ordem que satisfazem para $k \geq 1$

$$x_k \in x_0 + Lin\left\{\nabla f(x_0), \ldots, \nabla f(x_{k-1})\right\}, \tag{1.25}$$

com $x_0 \in \mathbb{R}^n$ arbitrário.

A afirmação (1.25) diz que a direção utilizada em cada iteração pelo método é uma combinação linear de gradientes de f em iterados anteriores, como por exemplo, o método do gradiente.

Teorema 1.11. *Para todo $k, 1 \leq k \leq \frac{1}{2}(n-1)$, e todo $x_0 \in \mathbb{R}^n$ existe uma função $f \in \mathscr{F}$ tal que para todo método M de primeira ordem satisfazendo (1.25) ,temos*

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|^2}{32(k+1)^2}, \tag{1.26}$$

$$\|x_k - x^*\|^2 \geq \frac{1}{8}\|x_0 - x^*\|^2, \tag{1.27}$$

onde x^* é o minimizador de f e $f^* = f(x^*)$.

Demonstração. A sequência gerada pelo método para função f começando em x_0, é a mesma gerada pelo método para $\bar{f}(x) = f(x+x_0)$ iniciando na origem. Portanto, sem perda de generalidade, podemos assumir $x_0 = 0$.

Fixado k, aplique o método M para minimizar $f(x) = f_{2k+1}(x)$ definida em (1.14). Então $x^* = \bar{x}_{2k+1}$, $f^* = f^*_{2k+1}$. Pelo Corolário 1.10, temos $f(x_k) = f_{2k+1}(x_k) = f_k(x_k) \geq f^*_k$.

Como $f_k(x_k) \geq \frac{L}{8}\left(-1 + \frac{1}{k+1}\right)$ e $f^*_{2k+1} = \frac{L}{8}\left(-1 + \frac{1}{2k+2}\right)$, temos

$$\begin{array}{rcl} f(x_k) - f^* & \geq & \frac{L}{8}\left(-1 + \frac{1}{k+1}\right) - \left(-1 + \frac{1}{2k+2}\right) \\ & = & \frac{L}{8}\left(\frac{1}{k+1} - \frac{1}{2k+2}\right) \\ & = & \frac{L}{8}\left(\frac{1}{2k+2}\right). \end{array} \tag{1.28}$$

Por outro por (1.21) temos,

$$\begin{array}{rcl} \|x_0 - x^*\|^2 & = & \|x^*\|^2 = \|\bar{x}_{2k+1}\|^2 = \displaystyle\sum_{i=1}^{n} (\bar{x}_{(2k+1)_i})^2 \\ & = & \displaystyle\sum_{i=1}^{2k+1} (\bar{x}_{(2k+1)_i})^2 = \sum_{i=1}^{2k+1} \left(1 - \frac{1}{2k+2}\right)^2 \\ & \leq & 2k+1 - \dfrac{2}{2k+2}\dfrac{(2k+1)(2k+2)}{2} + \dfrac{1}{3}(2k+2)^3 \dfrac{1}{(2k+2)^2} \\ & = & \frac{1}{3}(2k+2). \end{array}$$

Donde concluímos

$$\frac{1}{\|x_0 - x^*\|^2} \geq \frac{1}{\frac{1}{3}(2k+2)}. \tag{1.29}$$

Portanto combinando (1.28) com (1.29) obtemos

$$\frac{f(x_k) - f^*}{\|x_0 - x^*\|^2} \geq \frac{\dfrac{L}{8}\dfrac{1}{2k+2}}{\dfrac{1}{3}(2k+2)} = \frac{3L}{32(k+1)^3},$$

demonstrando (1.26).

Para demonstrar (1.27) temos as seguintes relações:

$$\sum_{i=k+1}^{2k+1} 1 = k+1, \tag{1.30}$$

$$\sum_{i=k+1}^{2k+1} i^2 = \frac{1}{6}\Bigg((k+1)(2k+1)(7k+6)\Bigg), \tag{1.31}$$

$$\sum_{i=k+1}^{2k+1} i = \frac{(k+1)(3k+2)}{2}. \tag{1.32}$$

Usando que $x^* = \bar{x}_{2k+1}$ e as relações (1.30), (1.31) e (1.32) obtemos

$$\begin{array}{rclcl}
\|x_k - x^*\|^2 & = & \displaystyle\sum_{i=1}^{n}((x_k - x^*)_i)^2 & \geq & \displaystyle\sum_{i=k+1}^{2k+1}((x_k - x^*)_i)^2 \\
& = & \displaystyle\sum_{i=k+1}^{2k+1}\left(\bar{x}_{(2k+1)_i}\right)^2 & = & \displaystyle\sum_{i=k+1}^{2k+1}\left(1 - \frac{i}{2k+2}\right)^2 \\
& = & \dfrac{2k^2+7k+6}{24(k+1)} & = & \dfrac{2k^2+7k+6}{16(k+1)^2}\dfrac{1}{3}(2k+2) \\
& \geq & \dfrac{2k^2+7k+6}{16(k+1)^2}\|x_0 - \bar{x}_{2k+1}\|^2 & = & \dfrac{2k^2+7k+6}{16(k+1)^2}\|x_0 - x^*\|^2 \\
& \geq & \dfrac{1}{8}\|x_0 - x^*\|^2. & &
\end{array}$$

□

Com este teorema podemos obter uma estimativa da cota de complexidade inferior para a classe $\mathscr{F}$.

Igualando $\dfrac{3L\|x_0 - x^*\|^2}{32(k+1)^2}$ a ε e isolando k obtemos $\ell = \sqrt{\dfrac{3L\|x_0 - x^*\|^2}{32\varepsilon}} - 1$, o que significa que $\ell = \Omega\left(\dfrac{1}{\sqrt{\varepsilon}}\right)$.

Isso significa dizer que para todo método M de primeira ordem satisfazendo (1.25), existe uma função $\bar{f} \in \mathscr{F}$ que o método gastará $\Omega\left(\frac{1}{\sqrt{\varepsilon}}\right)$ iterações para minimizá-la com precisão ε. Pela definição na Seção 1.1, a cota de complexidade inferior de $\mathscr{F}$ é $\Omega\left(\frac{1}{\sqrt{\varepsilon}}\right)$. O Teorema 1.11 explicita a função $\bar{f} \in \mathscr{F}$ independente do método, nesse caso.

1.3 Funções fortemente convexas

Essa seção é análoga à seção anterior para funções fortemente convexas conforme definição abaixo.

Definição 1.12. *Uma função continuamente diferenciável f é fortemente convexa em $\mathbb{R}^n$ se existe uma constante $\mu > 0$ tal que para todo $x, y \in \mathbb{R}^n$ temos*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}\mu \|y - x\|^2. \tag{1.33}$$

A constante $\mu > 0$ é chamada parâmetro de convexidade de f.

Note que se f é fortemente convexa, então f é convexa e portanto as propriedades vistas na seção anterior valem.

Denotaremos por $\mathscr{F}_\mu$ a classe das funções diferenciáveis fortemente convexas com parâmetro de convexidade μ e com constante de Lipschitz L para o gradiente. O valor $Q_f = \frac{L}{\mu}$ é chamado de *número de condição* da função f.

Por (1.8) temos para todo $x, y \in \mathbb{R}^n$

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|x - y\|^2.$$

Combinando com (1.33), obtemos $L \geq \mu$.

1.3.1 Propriedades

Lema 1.13. *Se $f \in \mathscr{F}_\mu$ e $\nabla f(x^*) = 0$ então*

$$f(x) \geq f(x^*) + \frac{1}{2}\mu \|x - x^*\|^2, \tag{1.34}$$

para todo $x \in \mathbb{R}^n$.

Demonstração. Segue de (1.33) com $y = x$ e $x = x^*$. □

Lema 1.14. *Se $f_1 \in \mathscr{F}_{\mu_1}, f_2 \in \mathscr{F}_{\mu_2}$ e $\alpha, \beta \geq 0$, então*

$$f = \alpha f_1 + \beta f_2 \in \mathscr{F}_{\alpha\mu_1 + \beta\mu_2}. \tag{1.35}$$

Demonstração. Para todo $x, y \in \mathbb{R}^n$ temos

$$f_1(y) \geq f_1(x) + \langle \nabla f_1(x), y - x \rangle + \frac{1}{2}\mu_1 \|y - x\|^2, \tag{1.36}$$

$$f_2(y) \geq f_2(x) + \langle \nabla f_2(x), y - x \rangle + \frac{1}{2}\mu_2 \|y - x\|^2. \tag{1.37}$$

Multiplicando (1.36) e (1.37) por α e β respectivamente e somando as, obtemos (1.35). □

Lema 1.15. *Uma função continuamente diferenciável f pertence a classe $\mathscr{F}_\mu$ se e somente se para todo $x, y \in \mathbb{R}^n$ temos*

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \mu \|x - y\|^2. \tag{1.38}$$

Demonstração. Se $f \in \mathscr{F}_\mu$ temos,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}\mu \|y - x\|^2,$$

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{1}{2}\mu \|y - x\|^2.$$

Somando essas duas equações obtemos (1.38).

Reciprocamente, denote $x_\tau = x + \tau(y - x)$.Então,

$$\begin{aligned}
f(y) &= f(x) + \int_0^1 \langle \nabla f(x + \tau(y - x)), y - x \rangle d\tau \\
&= f(x) + \langle \nabla f(x), y - x \rangle - \langle \nabla f(x), y - x \rangle + \int_0^1 \langle \nabla f(x_\tau), y - x \rangle d\tau \\
&= f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \langle \nabla f(x_\tau) - \nabla f(x), y - x \rangle d\tau \\
&= f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \langle \nabla f(x_\tau) - \nabla f(x), \frac{\tau}{\tau}(y - x) + x - x \rangle d\tau \\
&= f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \frac{1}{\tau} \langle \nabla f(x_\tau) - \nabla f(x), x_\tau - x \rangle d\tau \\
&\geq f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \frac{1}{\tau} \mu \|x_\tau - x\|^2 d\tau \\
&= f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|x - y\|^2.
\end{aligned}$$

□

Lema 1.16. *Se $f \in \mathscr{F}_\mu$, então para todo $x, y \in \mathbb{R}^n$ temos*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2\mu} \|\nabla f(x) - \nabla f(y)\|^2, \tag{1.39}$$

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq \frac{1}{\mu} \|\nabla f(x) - \nabla f(y)\|^2. \tag{1.40}$$

Demonstração. Fixe $x \in \mathbb{R}^n$ e considere a função $\phi(y) = f(y) - \langle \nabla f(x), y \rangle$. Note que $\nabla\phi(x) = \nabla f(x) - \nabla f(x) = 0$. Logo ϕ assume o valor mínimo em x. Temos também que

$$\begin{aligned}
\phi(y) &= f(y) - \langle \nabla f(x), y \rangle \\
&\geq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{1}{2}\mu\|y - x\|^2 - \langle \nabla f(x), y \rangle + \langle \nabla f(x), x \rangle - \langle \nabla f(x), x \rangle \\
&= \phi(x) + \langle \nabla f(x) - \nabla f(x), y - x \rangle + \tfrac{1}{2}\mu\|y - x\|^2 \\
&= \phi(x) + \langle \nabla\phi(x), y - x \rangle + \tfrac{1}{2}\mu\|y - x\|^2.
\end{aligned}$$

Isso mostra que $\phi \in \mathscr{F}_\mu$. Usando a definição de fortemente convexa temos

$$\phi(x) = \min_v \phi(v) \geq \min_v \left[\phi(y) + \langle \nabla\phi(y), v - y \rangle + \frac{1}{2}\mu\|v - y\|^2. \right. \tag{1.41}$$

Diferenciando (1.41) em relação a v temos $\nabla\phi(y) + \mu(v - y) = 0$, donde $\bar{v} = y - \frac{1}{\mu}\nabla\phi(y)$.

Substituindo $\bar{v}$ na desigualdade (1.41) obtemos

$$\phi(x) \geq \phi(y) - \left\langle \nabla\phi(y), \frac{1}{\mu}\nabla\phi(y) \right\rangle + \frac{1}{2}\mu \left\| \frac{1}{\mu}\nabla\phi(y) \right\|^2 = \phi(y) - \frac{1}{2\mu}\left\|\nabla\phi(y)\right\|^2.$$

Recuperando a forma original temos

$$f(x) - \langle \nabla f(x), x \rangle \geq f(y) - \langle \nabla f(x), y \rangle - \frac{1}{2\mu}\|\nabla f(y) - \nabla f(x)\|^2,$$

que é (1.39).

Trocando x por y e somando as obtemos (1.40). □

Note que μ é sempre menor ou igual a L.

Lema 1.17. *Se $f \in \mathscr{F}_\mu$, então para todo $x, y \in \mathbb{R}^n$ temos*

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{\mu L}{\mu + L}\|x - y\|^2 + \frac{1}{\mu + L}\|\nabla f(x) - \nabla f(y)\|^2. \tag{1.42}$$

Demonstração. Considere $\phi(x) = f(x) - \frac{1}{2}\mu\|x\|^2$. Então $\nabla\phi(x) = \nabla f(x) - \mu x$. Donde

$$\begin{aligned}
\langle \nabla\phi(x) - \nabla\phi(y), x - y \rangle &= \langle \nabla f(x) - \nabla f(y) - \mu(x - y), x - y \rangle \\
&= \langle \nabla f(x) - \nabla f(y), x - y \rangle - \mu\|x - y\|^2 \\
&\leq L\|x - y\|^2 - \mu\|x - y\|^2 = (L - \mu)\|x - y\|^2.
\end{aligned}$$

Isso mostra que ϕ é convexa com a constante de Lipschtiz do gradiente igual a $L - \mu$. Se

$\mu = L$ temos de (1.10) e (1.38) que

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L} \|\nabla f(x) - \nabla f(y)\|^2,$$

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \mu \|x - y\|^2.$$

Somando essas duas desigualdades obtemos

$$2\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L} \|\nabla f(x) - \nabla f(y)\|^2 + L\|x - y\|^2,$$

que é a desigualdade (1.42). Se $\mu < L$, por (1.10) temos

$$\frac{1}{L-\mu} \|\nabla \phi(x) - \nabla \phi(y)\|^2 \leq \langle \nabla \phi(x) - \nabla \phi(y), x - y \rangle.$$

Recuperando a forma original,

$$\|\nabla f(x) - \nabla f(y)\|^2 - 2\mu\langle \nabla f(x) - \nabla f(y), x - y \rangle + \mu^2 \|x - y\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle - (L - \mu)\mu \|x - y\|^2.$$

Juntando os termos semelhantes , obtemos (1.42). □

1.3.2 Cota de complexidade inferior

Nesterov [Nes04] cria uma função em um espaço de dimensão infinita difícil de ser minimizada por todo método de primeira ordem e mostra que a cota de complexidade relativa à classe das funções fortemente convexas com gradiente Lipschitz é da ordem de $\ln \frac{1}{\varepsilon}$. Segundo ele, construir essa função em um espaço de dimensão finita é mais complicado.

Considere $\mathbb{R}^\infty = \left\{ x = (z_i)_{i=1}^\infty \mid \|x\|^2 = \sum_{i=1}^\infty z_i^2 < \infty \right\}$. Esse espaço também é denotado por ℓ_2.

Escolhendo parâmetros $\mu > 0$ e $Q_f > 1$ definimos a função $f_{\mu,Q_f} : \mathbb{R}^\infty \to \mathbb{R}$ por

$$f_{\mu,Q_f}(x) = \frac{\mu(Q_f - 1)}{8} \left(z_1^2 + \sum_{i=1}^\infty (z_i - z_{i+1})^2 - 2z_1 \right) + \frac{\mu}{2} \|x\|^2. \tag{1.43}$$

Derivando f_{μ,Q_f} temos:

$$\nabla f_{\mu,Q_f}(x) = \frac{\mu(Q_f - 1)}{4} (2z_1 - 1 - z_2, 2z_2 - z_1 - z_3, \ldots) + \mu(z_1, z_2, \ldots).$$

Denotando

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & \ddots \\ 0 & 0 & \ddots & \ddots \end{pmatrix},$$

temos

$$\nabla f_{\mu,Q_f}(x) = \left(\frac{\mu(Q_f-1)}{4}A + \mu I\right)x - \frac{\mu(Q_f-1)}{4}e_1. \tag{1.44}$$

E calculando a hessiana, temos

$$\nabla^2 f_{\mu,Q_f}(x) = \frac{\mu(Q_f-1)}{4}A + \mu I. \tag{1.45}$$

De $0 \leq A \leq 4I$, pois $4I - A$ dada por

$$4I - A = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & \ddots \\ 0 & 0 & \ddots & \ddots \end{pmatrix}$$

é semi definida positiva, temos

$$\begin{array}{rcccl} 0 & \leq & \frac{\mu(Q_f-1)}{4}A & \leq & \mu(Q_f-1)I \\ \mu I & \leq & \frac{\mu(Q_f-1)}{4}A + \mu I & \leq & \mu(Q_f-1)I + \mu I \\ \mu I & \leq & \nabla^2 f_{\mu,Q_f}(x) & \leq & \mu Q_f I. \end{array}$$

Isso significa pelo Teorema 1.8, que $f_{\mu,Q_f} \in \mathscr{F}_\mu(\mathbb{R}^\infty)$ com constante de Lipschitz para o gradiente igual a μQ_f.

Note que o número de condição de f_{μ,Q_f} é

$$Q_{f_{\mu,Q_f}} = \frac{\mu Q_f}{\mu} = Q_f.$$

A condição de otimalidade de primeira ordem

$$\nabla f_{\mu,\mu Q_f}(x) = \left(\frac{\mu(Q_f-1)}{4}A + \mu I\right)x - \frac{\mu(Q_f-1)}{4}e_1 = 0,$$

resulta em

$$\left(\frac{\mu(Q_f-1)}{4}A + \mu I\right)x = \frac{\mu(Q_f-1)}{4}e_1$$

que pode ser reescrito como

$$Ax + \frac{4}{Q_f - 1}x = e_1. \tag{1.46}$$

Separando as coordenadas temos

$$\begin{cases} 2z_1 - z_2 + \frac{4}{Q_f-1}z_1 = 1 \\ -z_1 + 2z_2 - z_3 + \frac{4}{Q_f-1}z_2 = 0 \\ -z_2 + 2z_3 - z_4 + \frac{4}{Q_f-1}z_3 = 0 \end{cases}$$

e assim por diante. Esse sistema pode ser escrito como

$$\begin{cases} \left(2\frac{Q_f+1}{Q_f-1}\right)z_1 - z_2 = 1 \\ z_{k+1} - 2\frac{Q_f+1}{Q_f-1}z_k + z_{k-1} = 0, \quad k = 2, \ldots \end{cases} \tag{1.47}$$

Seja q a menor raiz da equação $q^2 - 2\frac{Q_f+1}{Q_f-1}q + 1 = 0$, isto é, $q = \dfrac{\sqrt{Q_f}-1}{\sqrt{Q_f}+1}$.

Note que q de fato é uma raiz da equação. A conta abaixo mostra isso.

$$\frac{\left(\sqrt{Q_f}-1\right)^2}{\left(\sqrt{Q_f}+1\right)^2} - 2\frac{\left(Q_f+1\right)}{\left(Q_f-1\right)}\frac{\left(\sqrt{Q_f}-1\right)}{\left(\sqrt{Q_f}+1\right)} + 1 =$$

$$= \frac{\left(Q_f-1\right)\left(\sqrt{Q_f}-1\right)^2 - 2\left(Q_f+1\right)\left(\sqrt{Q_f}-1\right)\left(\sqrt{Q_f}+1\right)}{\left(\sqrt{Q_f}+1\right)^2\left(Q_f-1\right)} + 1$$

$$= \frac{\left(\sqrt{Q_f}-1\right)^2 - 2\left(Q_f+1\right)}{\left(\sqrt{Q_f}+1\right)^2} + 1$$

$$= -\frac{\left(\sqrt{Q_f}+1\right)^2}{\left(\sqrt{Q_f}+1\right)^2} + 1 = 0.$$

Então $z_k = q^k, k = 1, 2, \ldots$ satisfaz o sistema (1.47), ou seja, x^* cuja k-ésima coordenada $z_k^* = q^k$ anula o gradiente da função f_{μ,Q_f} e é portanto seu minimizador.

Teorema 1.18. *Para todo $x_0 \in \mathbb{R}^\infty$ e todas constantes $\mu > 0, Q_f > 1$, existe uma função $f \in \mathscr{F}_{\mu,}(\mathbb{R}^\infty)$ tal que para todo método M de primeira ordem satisfazendo (1.25) temos*

$$\|x_k - x^*\|^2 \geq \left(\frac{\sqrt{Q_f}-1}{\sqrt{Q_f}+1}\right)^{2k} \|x_0 - x^*\|^2, \tag{1.48}$$

$$f(x_k)-f^* \geq \frac{\mu}{2}\left(\frac{\sqrt{Q_f}-1}{\sqrt{Q_f}+1}\right)^{2k}\|x_0-x^*\|^2, \tag{1.49}$$

onde x^* é o minimizador de f e $f^*=f(x^*)$.

Demonstração. Sem perda de generalidade, assumimos que $x_0=0$. Escolhemos $f(x)=f_{\mu,\mu Q_f}(x)$. Então

$$\|x_0-x^*\|^2=\sum_{i=1}^{\infty}[(x^*)_i]^2=\sum_{i=1}^{\infty}q^{2i}=\frac{q^2}{1-q^2}.$$

Uma vez que $\nabla^2 f_{\mu,\mu Q_f}(x)$ é um operador tri-diagonal, e $\nabla f_{\mu,\mu Q_f}(0)=-\frac{\mu(Q_f-1)}{4}e_1$, nós concluímos que $x_k\in\mathbb{R}^{k,\infty}$.

Então

$$\|x_k-x^*\|^2 \geq \sum_{i=k+1}^{\infty}[(x^*)_i]^2\sum_{i=k+1}^{\infty}q^{2i}=\frac{q^{2(k+1)}}{1-q}=q^{2k}\|x_0-x^*\|^2.$$

Isso demonstra (1.48).

Pelo Lema 1.13, temos

$$f(x_k)-f^* \geq \frac{1}{2}\mu\|x_k-x^*\|^2 \geq \frac{1}{2}\mu q^{2k}\|x_0-x^*\|^2.$$

Isso prova (1.49). □

O lema a seguir nos auxiliará a obter uma estimativa da cota de complexidade inferior para a classe $\mathscr{F}_\mu$.

Lema 1.19. *Considere $a>1$. Então para todo $k\geq 0$ temos*

$$\left(\frac{\sqrt{a}-1}{\sqrt{a}+1}\right)^{2k} \geq e^{\frac{-4k}{\sqrt{a}-1}}.$$

Demonstração. Pelo teorema do valor médio existe $c\in(\sqrt{a}-1,\sqrt{a}+1)$ tal que

$$\ln\left(\frac{\sqrt{a}+1}{\sqrt{a}-1}\right)=\ln(\sqrt{a}+1)-\ln(\sqrt{a}-1)=\frac{1}{c}(\sqrt{a}+1-\sqrt{a}+1)=\frac{2}{c}<\frac{2}{\sqrt{a}-1}.$$

Como a função exponencial é crescente, temos que

$$\left(\frac{\sqrt{a}+1}{\sqrt{a}-1}\right)<e^{\frac{2}{\sqrt{a}-1}},$$

donde segue que

$$\left(\frac{\sqrt{a}-1}{\sqrt{a}+1}\right)^{2k} \geq e^{\frac{-4k}{\sqrt{a}-1}}.$$

□

Pelo Teorema 1.18 e o Lema 1.19 temos

$$f(x_k)-f^* \geq \frac{\mu}{2}\left(\frac{\sqrt{Q_f}-1}{\sqrt{Q_f}+1}\right)^{2k}\|x_0-x^*\|^2 \geq \frac{\mu}{2}e^{\frac{-4k}{\sqrt{Q_f}-1}}\|x_0-x^*\|^2. \tag{1.50}$$

Igualando $\frac{\mu}{2}e^{\frac{-4k}{\sqrt{Q_f}-1}}\|x_0-x^*\|^2$ a ε e isolando k obtemos $\ell = \frac{\sqrt{Q_f}-1}{4}\left(\ln\frac{1}{\varepsilon}+\ln\frac{\mu}{2}+2\ln\|x_0-x^*\|\right)$. A menos de constante isso significa que $\ell = \Omega\left(\ln\frac{1}{\varepsilon}\right)$.

Isso significa dizer que para todo método M de primeira ordem satisfazendo (1.25), existe uma função $\bar{f}\in\mathscr{F}_\mu$ que o método gastará $\Omega\left(\ln\frac{1}{\varepsilon}\right)$ iterações para minimizá-la com precisão ε. Pela definição na Seção 1.1, a cota de complexidade inferior de $\mathscr{F}_\mu$ é $\Omega\left(\ln\frac{1}{\varepsilon}\right)$.

1.4 Método do Gradiente

O método mais conhecido para minimizar uma função em $\mathbb{R}^n$ é o método de máxima descida, devido a Cauchy, no século XIX, que toma a cada iteração a direção oposta ao gradiente. O objetivo dessa seção é mostrar que o método do gradiente não é ótimo no sentido discutido na Seção 1.1. Apresentaremos abaixo o algoritmo básico com algumas opções para o cálculo do comprimento do passo ao longo de cada direção considerada.

Algoritmo 1.20. *Método do Gradiente*

Dado $x_0\in\mathbb{R}^n$

k=0

Repita

$d_k = -\nabla f(x_k)$.

Calcule o comprimento do passo $\lambda > 0$.

$x_{k+1} = x_k + \lambda d_k$.

$k = k+1$.

Fim

O comprimento do passo λ pode ser calculado de diferentes [IS07, NW99] maneiras. Satisfazendo, por exemplo, uma das condições abaixo:

- Passo pré definido: $0<\lambda<\frac{2}{L}$.
- Busca exata: $\lambda = \underset{t\geq 0}{argmin}\, f(x_k - t\nabla f(x_k))$.

- Goldstein-Armijo. Dados α, β com $0 < \alpha < \beta < 1$, determine λ tal que

$$\alpha\langle\nabla f(x_k), x_k - x_{k+1}\rangle \leq f(x_k) - f(x_{k+1}), \tag{1.51}$$

$$\beta\langle\nabla f(x_k), x_k - x_{k+1}\rangle \geq f(x_k) - f(x_{k+1}), \tag{1.52}$$

  onde $x_{k+1} = x_k - \lambda\nabla f(x_k)$.

- Armijo. Dado $\alpha \in (0,1)$. Determine λ que satisfaça a condição:

$$f(x_k - \lambda\nabla f(x_k)) \leq f(x_k) - \alpha\lambda\|\nabla f(x_k)\|^2. \tag{1.53}$$

Nosso objetivo é discutir como o método do gradiente se comporta quando aplicado para a minimização de uma função $f \in \mathscr{F}$.

Teorema 1.21. *Seja $f \in \mathscr{F}$ e $0 < \lambda < \frac{2}{L}$. Então o método do gradiente gera uma sequência $\{x_k\}$ que converge como segue:*

$$f(x_k) - f^* \leq \frac{2(f(x_0) - f^*)\|x_0 - x^*\|^2}{2\|x_0 - x^*\|^2 + k\lambda(2 - L\lambda)(f(x_0) - f^*)}. \tag{1.54}$$

Demonstração. Denote $r_k = \|x_k - x^*\|^2$. Então,

$$\begin{aligned} r_{k+1}^2 &= \|x_{k+1} - x^*\|^2 = \|x_k - \lambda\nabla f(x_k) - x^*\|^2 \\ &= r_k^2 - 2\lambda\langle\nabla f(x_k), x_k - x^*\rangle + \lambda^2\|\nabla f(x_k)\|^2 \\ &\leq r_k^2 - \tfrac{1}{L}2\lambda\|\nabla f(x_k)\|^2 + \lambda^2\|\nabla f(x_k)\|^2 \\ &= r_k^2 - \lambda(\tfrac{2}{L} - \lambda)\|\nabla f(x_k)\|^2, \end{aligned}$$

onde na passagem para desigualdade usamos (1.10) e $\nabla f(x^*) = 0$. Disto concluímos que $r_k \leq r_0$.

Por (1.8) temos,

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \langle\nabla f(x_k), x_{k+1} - x_k\rangle + \tfrac{L}{2}\|x_{k+1} - x_k\|^2 \\ &= f(x_k) + \langle\nabla f(x_k), -\lambda\nabla f(x_k)\rangle + \tfrac{L}{2}\|-\lambda\nabla f(x_k)\|^2 \\ &= f(x_k) - \lambda\|\nabla f(x_k)\|^2 + \lambda^2\tfrac{L}{2}\|\nabla f(x_k)\|^2. \end{aligned}$$

Donde

$$f(x_{k+1}) \leq f(x_k) - w\|\nabla f(x_k)\|^2, \tag{1.55}$$

onde $w = \lambda(1 - \frac{L}{2}\lambda)$.

Denote $\Delta_k = f(x_k) - f^*$. Então

$$\Delta_k \leq \langle\nabla f(x_k), x_k - x^*\rangle \leq \|\nabla f(x_k)\| r_k \leq r_0\|\nabla f(x_k)\|. \tag{1.56}$$

Da desigualdade (1.55), temos

$$\begin{aligned} f(x_{k+1}) - f^* &\leq f(x_k) - f^* - w\|\nabla f(x_k)\|^2 \\ \Delta_{k+1} &\leq \Delta_k - w\|\nabla f(x_k)\|^2. \end{aligned}$$

E usando (1.56), obtemos

$$\Delta_{k+1} \leq \Delta_k - \frac{w}{r_0^2}\Delta_k^2. \tag{1.57}$$

Multiplicando (1.57) por $\frac{1}{\Delta_k \Delta k+1}$ temos

$$\frac{1}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{w}{r_k^2}\frac{\Delta_k}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{w}{r_0^2}. \tag{1.58}$$

Portanto usando (1.58) k vezes,

$$\frac{1}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{w}{r_0^2} \geq \frac{1}{\Delta_{k-1}} + 2\frac{w}{r_0^2} \geq \cdots \geq \frac{1}{\Delta_0} + \frac{w}{r_0^2}(k+1).$$

Obtemos,

$$\frac{1}{\Delta_{k+1}} \geq \frac{1}{\Delta_0} + \frac{w}{r_0^2}(k+1). \tag{1.59}$$

De (1.59), obtemos

$$\Delta_k \leq \frac{\Delta_0 r_0^2}{r_0^2 + \Delta_0 w(k+1)}. \tag{1.60}$$

Substituindo w em (1.60) obtemos

$$\Delta_k \leq \frac{2\Delta_0 r_0^2}{2r_0^2 + k\lambda(2 - L\lambda)\Delta_0}.$$

□

Se escolhermos $\lambda^* = \frac{1}{L}$, temos a seguinte estimativa para o método do gradiente:

$$f(x_k) - f^* \leq \frac{2L(f(x_0) - f^*)\|x_0 - x^*\|^2}{2L\|x_0 - x^*\|^2 + k(f(x_0) - f^*)}. \tag{1.61}$$

Corolário 1.22. *Se $\lambda = \frac{1}{L}$ e $f \in \mathscr{F}$, então*

$$f(x_k) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{k+4}. \tag{1.62}$$

Demonstração. Pela desigualdade (1.8) temos,

$$f(x_0) \leq f^* + \langle \nabla f(x^*), x_0 - x^* \rangle + \frac{L}{2}\|x_0 - x^*\|^2.$$

Donde, obtemos

$$f(x_0) - f^* \leq \frac{L}{2} r_0^2. \tag{1.63}$$

Substituindo (1.63) em (1.61) temos

$$\begin{aligned} f(x_k) - f^* &\leq \frac{2L(f(x_0) - f^*) r_0^2}{2Lr_0^2 + k(f(x_0) - f^*)} \\ &\leq \frac{2L\frac{L}{2} r_0^2 r_0^2}{2Lr_0^2 + k\frac{L}{2} r_0^2} = \frac{2Lr_0^2}{k+4}. \end{aligned}$$

□

Com esses resultados acima, notamos que a cota de complexidade superior do Método do Gradiente relativo a classe de funções $\mathscr{F}$ é da ordem de $\frac{1}{\varepsilon}$, mostrando que o método do gradiente não é ótimo para essa classe. Uma conta análoga pode ser feita para a classe $\mathscr{F}_\mu$.

O capítulo a seguir é dedicado à apresentação de um método ótimo para a classe de funções $\mathscr{F}$ e $\mathscr{F}_\mu$.

Capítulo 2

Um Método Ótimo

No capítulo anterior foi mostrado que a cota de complexidade inferior das classes de funções convexas e fortemente convexas são da ordem de $\frac{1}{\sqrt{\varepsilon}}$ e $\ln\frac{1}{\varepsilon}$ respectivamente. Mostramos também que o método do gradiente não é ótimo. Agora exibiremos um algoritmo cuja cota de complexidade superior é da ordem da complexidade inferior dessa classe de funções, portanto ótimo.

Nesterov em [Nes04] exibe um algoritmo ótimo para a minimização irrestrita de funções fortementes convexas. Gonzaga e Karas em [GK08] rediscutem esse algoritmo procurando enfatizar uma interpretação geométrica dos seus fundamentos e apresentam uma classe de algoritmos ótimos. Neste capítulo nos focamos num destes métodos que é um ajuste fino do método proposto em [Nes04]. Procuramos enfatizar suas propriedades geométricas e provamos sua complexidade ótima. Em seguida, apresentamos uma modificação do algoritmo sugerido em [GK08] para a minimização de funções fortementes convexas quando o parâmetro de convexidade não é conhecido.

Lembramos que L denota a constante de Lipschitz para o gradiente e μ o parâmetro de convexidade da função f.

2.1 Algoritmo

Nosso objetivo é exibir um algoritmo ótimo para o problema de minimização de uma função da classe $\mathscr{F}_\mu$.

Algoritmo 1. *Gonzaga-Karas*

Dado $x_0 \in \mathbb{R}^n$, $v_0 = x_0$, $\gamma_0 > \mu$.

k=0

Repita

$d_k = v_k - x_k$.

Se $f(v_k) \leq f(x_k)$ *então* $y_k = v_k$, *senão*

Escolha $\theta \in [0,1)$ *e* $y_k = x_k + \theta d_k$ *sujeito a* $f(y_k) \leq f(x_k)$ *e* $\langle \nabla f(y_k), d_k \rangle \geq 0$.

Calcule $x_{k+1} = y_k - \lambda \nabla f(y_k)$, *satisfazendo*

$f(x_{k+1}) \leq f(y_k) - \frac{1}{4L}\|\nabla f(y_k)\|^2$.

$G = \frac{\mu}{2}\|y_k - v_k\|^2 + \langle \nabla f(y_k), v_k - y_k \rangle$,

$A = -\frac{1}{2}\|\nabla f(y_k)\|^2 + (\gamma_k - \mu)(f(x_k) - f(y_k)) - \gamma_k G$,

$B = (\gamma_k - \mu)(f(x_{k+1}) - f(x_k)) + \gamma_k(f(y_k) - f(x_k)) + \gamma_k G$,

$C = \gamma_k(f(x_k) - f(x_{k+1}))$.

Compute $\alpha_k \in (0,1)$ *solução de* $A\alpha^2 + B\alpha + C = 0$.

$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k \mu$.

$v_{k+1} = \frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu y_k - \nabla f(y_k)))$.

$k = k+1$.

Fim

Note que a sequência $\{f(x_k)\}$ é monótona decrescente.

2.1.1 Boa definição

Nesta seção mostraremos que o algoritmo está bem definido, ou seja, que todos os passos do algoritmo são possíveis.

Lema 2.1. *Considere* $x_k, v_k \in \mathbb{R}^n$ *e* $d_k = v_k - x_k$. *Se* $f(v_k) > f(x_k)$, *existe* $\theta \in [0,1)$ *tal que*

$$f(y_k) \leq f(x_k) \quad e \quad \langle \nabla f(y_k), d_k \rangle \geq 0, \tag{2.1}$$

onde $y_k = x_k + \theta d_k$.

Demonstração. Defina $h : [0,\infty) \to \mathbb{R}$ por $h(\theta) = f(x_k + \theta d_k)$. Note que $h(0) = f(x_k)$ e $h(1) = f(v_k)$ e por hipótese $h(0) < h(1)$. Observe que $h'(\theta) = \langle \nabla f(x_k + \theta d_k), d_k \rangle$. Se $h'(0) \geq 0$, então $\theta = 0$ satisfaz (2.1). Neste caso $y_k = x_k$. Veja Figura 2.1.

Caso contrário, $h'(0) \leq 0$ e portanto existe $\bar{\theta} \in (0,1)$ tal que $h(\bar{\theta}) < h(0)$. Pelo Teorema do valor intermediário existe $\tilde{\theta} \in (\bar{\theta}, 1)$ tal que

$$h(\tilde{\theta}) = h(0). \tag{2.2}$$

Além disso, pela convexidade de h,

$$h(0) \geq h(\tilde{\theta}) + h'(\tilde{\theta})(-\tilde{\theta}),$$

o que comparando com (2.2) garante que $h'(\tilde{\theta}) \geq 0$, o que prova a existência de θ satisfazendo (2.1). Veja Figura 2.2.

Vamos mostrar que neste caso θ não é único. Como $h'(0) < 0$ e $h'(\tilde{\theta}) \geq 0$, pelo teorema do valor intermediário existe $\theta^* \in (0, \tilde{\theta}]$ tal que

$$h'(\theta^*) = 0.$$

Como h é convexa, θ^* é um minimizador de h e $h(\theta^*) < h(0)$. Portanto pela convexidade de h todo $\theta \in [\theta^*, \tilde{\theta}]$ satisfaz (2.2). □

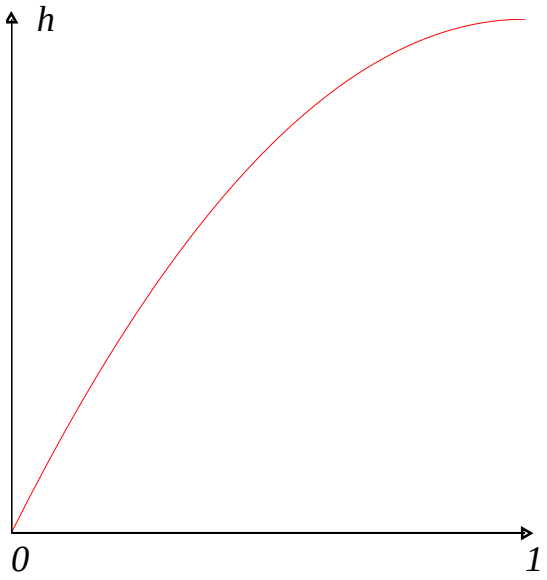

Figura 2.1: $\theta = 0$

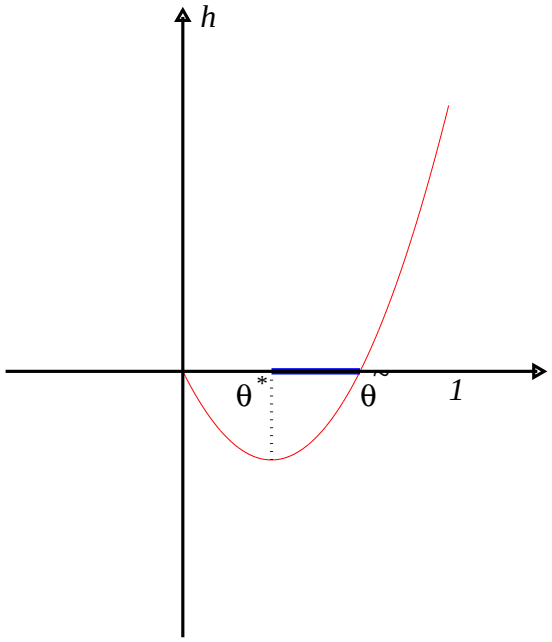

Figura 2.2: $\theta \in [\theta^*, \tilde{\theta}]$

Lema 2.2. *A equação*

$$A\alpha^2 + B\alpha + C = 0, \tag{2.3}$$

onde

$$G = \frac{\mu}{2}\|y_k - v_k\|^2 + \langle \nabla f(y_k), v_k - y_k \rangle, \tag{2.4}$$

$$A = -\frac{1}{2}\|\nabla f(y_k)\|^2 + (\gamma_k - \mu)(f(x_k) - f(y_k)) - \gamma_k G, \tag{2.5}$$

$$B = (\gamma_k - \mu)(f(x_k) - f(x_{k+1}) + \gamma_k(f(y_k) - f(x_k)) + \gamma_k G, \tag{2.6}$$

$$C = \gamma_k(f(x_k) - f(x_{k+1})), \tag{2.7}$$

tem uma solução em $(0, 1]$.

Demonstração. Considere $g:[0,\infty)\to\mathbb{R}$ definida por $g(\alpha)=A\alpha^2+B\alpha+C$. Temos que

$$g(0)=\gamma_k(f(x_k)-f(x_{k+1}))>0$$

pois $\gamma_k>0$ e $f(x_k)>f(x_{k+1})$. Temos também que

$$g(1)=A+B+C=-\frac{1}{2}\|\nabla f(y_k)\|^2+\mu(f(y_k)-f(x_{k+1})). \tag{2.8}$$

Por (1.33) temos

$$f(y_k)-f(x_{k+1})\leq-\frac{\mu}{2}\|x_{k+1}-y_k\|^2-\langle\nabla f(y_k),x_{k+1}-y_k\rangle.$$

Comparando com (2.8) temos que

$$\begin{aligned} g(1) &\leq -\frac{1}{2}\left(\|\nabla f(y_k)\|^2+\mu^2\|x_{k+1}-y_k\|^2+2\mu\langle\nabla f(y_k),x_{k+1}-y_k\rangle\right)\\ &= -\frac{1}{2}\left(\|\nabla f(y_k)+\mu(x_{k+1}-y_k)\|^2\right)\leq 0. \end{aligned}$$

Portanto pelo teorema do valor intermedário existe $\alpha\in(0,1]$ tal que $g(\alpha)=0$. □

No algoritmo exige-se que $x_{k+1}=y_k-\lambda\nabla f(y_k)$, satisfaça uma condição de decréscimo:

$$f(x_{k+1})\leq f(y_k)-\frac{1}{4L}\|\nabla f(y_k)\|^2. \tag{2.9}$$

É necessário saber se isso sempre é possível. De fato, é possível como mostra o lema abaixo.

Lema 2.3. *Seja $x_{k+1}=y_k-\lambda\nabla f(y_k)$. Então o passo $\lambda=\frac{1}{L}$ satisfaz*

$$f(x_{k+1})\leq f(y_k)-\frac{1}{4L}\|\nabla f(y_k)\|^2.$$

Demonstração. Pela desigualdade (1.8) com $x=y_k$ e $y=x_{k+1}$ temos

$$\begin{aligned} f(x_{k+1})-f(y_k)-\langle\nabla f(y_k),-\frac{1}{L}\nabla f(y_k)\rangle &\leq \frac{L}{2}\|\frac{1}{L}\nabla f(y_k)\|^2\\ f(x_{k+1})-f(y_k) &\leq \left(\frac{1}{2L}-\frac{1}{L}\right)\|\nabla f(y_k)\|^2 \end{aligned}$$

Multiplicando por -1 essa desigualdade temos

$$f(y_k)-f(x_{k+1})\geq\frac{1}{2L}\|\nabla f(y_k)\|^2\geq\frac{1}{4L}\|\nabla f(y_k)\|^2.$$

□

2.1.2 Interpretação geométrica

O algoritmo gera sequências $\{x_k\}$,$\{y_k\}$ e $\{v_k\}$ em $\mathbb{R}^n$, $\{\gamma_k\}$, $\{\alpha_k\}$ em $\mathbb{R}$. O objetivo dessa seção é discutir algumas propriedades dessas sequências, inclusive uma interpretação geométrica referente à construção das mesmas.

A chave da interpretação geométrica reside na construção de uma sequência de funções ϕ_k definida da seguinte maneira. Dado $x_0 \in \mathbb{R}^n, \gamma_0 \leq L$ e $v_0 = x_0$, defina

$$\phi_0 = f(x_0) + \frac{\gamma_0}{2}\|x - v_0\|^2 \tag{2.10}$$

e ϕ_{k+1} como uma combinação convexa de ϕ_k com uma aproximação quadrática de f em torno de y_k, ou seja,

$$\phi_{k+1}(x) = (1-\alpha_k)\phi_k(x) + \alpha_k l_k(x), \tag{2.11}$$

com

$$l_k(x) = f(y_k) + \langle \nabla f(y_k), x - y_k\rangle + \frac{\mu}{2}\|x - y_k\|^2. \tag{2.12}$$

Tendo em vista (1.33), temos que para todo k, l_k é uma aproximação inferior de f.

Lema 2.4. *A sequência de funções ϕ_k satisfaz*

$$\phi_k(x) = f(x_k) + \frac{\gamma_k}{2}\|x - v_k\|^2, \tag{2.13}$$

com $\gamma_0 \geq L$ e $v_0 = x_0$ dados,

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k \mu, \tag{2.14}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu y_k - \nabla f(y_k))\right), \tag{2.15}$$

e α_k dado como raiz da equação (2.3). Além disso, $\alpha_k \in (0,1]$ é solução de (2.3) é equivalente a $\phi_{k+1}(v_{k+1}) = f(x_{k+1})$ e

$$\begin{aligned}\phi_{k+1}^* = \phi_{k+1}(v_{k+1}) &= (1-\alpha_k)f(x_k) + \alpha_k f(y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}}\|\nabla f(y_k)\|^2 \\ &+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left(\frac{\mu}{2}\|v_k - y_k\|^2 + \langle \nabla f(y_k), v_k - y_k\rangle\right).\end{aligned} \tag{2.16}$$

Demonstração. Note que $\nabla\phi_0(x) = \gamma_0(x - v_0)$ e $\nabla^2\phi_0(x) = \gamma_0 I$.

Inicialmente, vamos mostrar por indução que $\nabla^2\phi_k = \gamma_k I$ para todo $k \geq 0$. Suponha que isso é válido para algum $k \geq 0$. Por (2.11) temos que

$$\phi_{k+1}(x) = (1-\alpha_k)\phi_k(x) + \alpha_k\left(f(y_k) + \langle \nabla f(y_k), x - y_k\rangle + \frac{\mu}{2}\|x - y_k\|^2\right). \tag{2.17}$$

Derivando duas vezes em relação a x temos

$$\nabla^2\phi_{k+1}(x) = (1-\alpha_k)\nabla^2\phi_k(x) + \alpha_k\mu I.$$

Usando a hipótese de indução e a definição de γ_{k+1}, temos que

$$\nabla^2\phi_{k+1}(x) = \left((1-\alpha_k)\gamma_k + \alpha_k\mu\right) I = \gamma_{k+1} I.$$

Isso justifica a forma de ϕ_k dada em (2.13).

Suponha agora que v_k é o minimizador de ϕ_k com valor mínimo $f(x_k)$. Note por (2.10) que isso vale para $k=0$. Suponha que vale para algum $k \geq 0$, ou seja, que

$$\phi_k(x) = f(x_k) + \frac{\gamma_k}{2}\|x - v_k\|^2.$$

Substituindo isso em (2.17) temos

$$\phi_{k+1}(x) = (1-\alpha_k)\left(f(x_k) + \frac{\gamma_k}{2}\|x-v_k\|^2\right) + \alpha_k\left(f(y_k) + \langle\nabla f(y_k), x - y_k\rangle + \frac{\mu}{2}\|x-y_k\|^2\right).$$

Derivando e igualando a zero obtemos

$$(1-\alpha_k)\gamma_k(x-v_k) + \alpha_k\nabla f(y_k) + \alpha_k\mu(x-y_k) = 0.$$

Isolando x obtemos a expressão para o minimizador de ϕ_{k+1} :

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu y_k - \nabla f(y_k))\right).$$

Resta mostrar que $\phi^*_{k+1} := \phi_{k+1}(v_{k+1}) = f(x_{k+1})$. Temos que

$$\phi_{k+1}(v_{k+1}) = (1-\alpha_k)\phi_k(v_{k+1}) + \alpha_k\left(f(y_k) + \langle\nabla f(y_k), v_{k+1} - y_k\rangle + \frac{\mu}{2}\|v_{k+1}-y_k\|^2\right). \tag{2.18}$$

Subtraindo y_k do minimizador v_{k+1} obtemos

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k(v_k - y_k) - \alpha_k\nabla f(y_k)\right), \tag{2.19}$$

Subtraindo agora v_k do minimizador v_{k+1} obtemos

$$\begin{aligned} & \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu y_k - \nabla f(y_k))\right) - \frac{v_k\gamma_{k+1}}{\gamma_{k+1}} \\ = \; & \frac{1}{\gamma_{k+1}}\left(\left((1-\alpha_k)\gamma_k - \gamma_{k+1}\right)v_k + \alpha_k(\mu y_k - \nabla f(y_k))\right). \end{aligned} \tag{2.20}$$

Usando que $\gamma_{k+1} = (1-\alpha_k)+\alpha_k\mu$ e substituindo em (2.20), obtemos

$$\begin{aligned} v_{k+1}-v_k &= \frac{1}{\gamma_{k+1}}\left(-\alpha_k\mu v_k+\alpha_k(\mu y_k-\nabla f(y_k))\right) \\ &= \frac{1}{\gamma_{k+1}}\left(\alpha_k\mu(y_k-v_k)-\alpha_k\nabla f(y_k)\right) \\ &= \frac{1}{\gamma_{k+1}}\left(\alpha_k\left(\mu(y_k-v_k)-\nabla f(y_k)\right)\right). \end{aligned} \tag{2.21}$$

De (2.21) obtemos

$$\|v_{k+1}-v_k\|^2 = \frac{\alpha_k^2}{\gamma_{k+1}^2}\left(\mu^2\|v_k-y_k\|^2+2\mu\langle v_k-y_k,\nabla f(y_k)\rangle+\|\nabla f(y_k)\|^2\right), \tag{2.22}$$

Usando (2.19) temos

$$\begin{aligned} \|v_{k+1}-y_k\|^2 &= \|\frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k(v_k-y_k)-\alpha_k\nabla f(y_k)\right)\|^2 \\ &= \frac{1}{\gamma_{k+1}^2}(1-\alpha_k)^2\gamma_k^2\|v_k-y_k\|^2-2\frac{1}{\gamma_{k+1}}\langle(1-\alpha_k)\gamma_k(v_k-y_k),\alpha_k\nabla f(y_k)\rangle+\alpha_k^2\|\nabla f(y_k)\|^2 \\ &= \frac{1}{\gamma_{k+1}^2}\left((1-\alpha_k)^2\gamma_k^2\|v_k-y_k\|^2-2(1-\alpha_k)\alpha_k\gamma_k\langle\nabla f(y_k),v_k-y_k\rangle\right) \\ &+ \frac{1}{\gamma_{k+1}^2}\left(\alpha_k^2\|\nabla f(y_k)\|^2\right). \end{aligned} \tag{2.23}$$

De (2.18) obtemos

$$\begin{aligned} \phi_{k+1}^* &= (1-\alpha_k)\left(f(x_k)+\frac{\gamma_k}{2}\|v_{k+1}-v_k\|^2\right)+\alpha_k\left(f(y_k)+\langle\nabla f(y_k),v_{k+1}-y_k\rangle+\frac{\mu}{2}\|v_{k+1}-y_k\|^2\right) \\ &= (1-\alpha_k)f(x_k)+\alpha_k f(y_k)+(1-\alpha_k)\frac{\gamma_k}{2}\|v_{k+1}-v_k\|^2 \\ &+ \alpha_k\langle\nabla f(y_k),v_{k+1}-y_k\rangle+\alpha_k\frac{\mu}{2}\|v_{k+1}-y_k\|^2. \end{aligned} \tag{2.24}$$

Substituindo (2.19),(2.22),(2.23) em (2.24) obtemos

$$\begin{aligned} \phi_{k+1}^* &= (1-\alpha_k)f(x_k)+\alpha_k f(y_k)+(1-\alpha_k)\frac{\gamma_k}{2}\left(\frac{\alpha_k^2}{\gamma_{k+1}^2}\|\mu(v_k-y_k)+\nabla f(y_k)\|^2\right) \\ &+ \alpha_k\langle\nabla f(y_k),\frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k(v_k-y_k)-\alpha_k\nabla f(y_k)\right)\rangle \\ &+ \alpha_k\frac{\mu}{2}\frac{1}{\gamma_{k+1}^2}\|(1-\alpha_k)\gamma_k(v_k-y_k)-\alpha_k\nabla f(y_k)\|^2. \end{aligned}$$

Desenvolvendo os produtos internos, obtemos

$$\begin{aligned}\phi^*_{k+1} &= (1-\alpha_k)f(x_k)+\alpha_k f(y_k)+\frac{(1-\alpha_k)\alpha_k^2\gamma_k}{2\gamma_{k+1}^2}\left(\mu^2\|v_k-y_k\|^2+\|\nabla f(y_k)\|^2\right)-\frac{\alpha_k^2}{\gamma_{k+1}}\|\nabla f(y_k)\|^2\\ &+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left(\langle\nabla f(y_k),v_k-y_k\rangle+\frac{(1-\alpha_k)\gamma_k\mu}{2\gamma_{k+1}}\|v_k-y_k\|^2\right)+\frac{\alpha_k^3\mu}{2\gamma_{k+1}^2}\|\nabla f(y_k)\|^2. \quad (2.25)\end{aligned}$$

Simplificando (2.25) temos

$$\begin{aligned}\phi^*_{k+1} &= (1-\alpha_k)f(x_k)+\left(\frac{(1-\alpha_k)\alpha_k^2\gamma_k}{2\gamma_{k+1}^2}-\frac{\alpha_k^2}{\gamma_{k+1}}+\frac{\alpha_k^3\mu}{2\gamma_{k+1}^2}\right)\|\nabla f(y_k)\|^2\\ &+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left(\frac{\mu}{2}\|v_k-y_k\|^2+\langle\nabla f(y_k),v_k-y_k\rangle\right). \quad (2.26)\end{aligned}$$

Simplificando (2.26) obtemos

$$\begin{aligned}\phi^*_{k+1} &= (1-\alpha_k)f(x_k)+\alpha_k f(y_k)-\frac{\alpha_k^2}{2\gamma_{k+1}}\|\nabla f(y_k)\|^2\\ &+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left(\frac{\mu}{2}\|v_k-y_k\|^2+\langle\nabla f(y_k),v_k-y_k\rangle\right). \quad (2.27)\end{aligned}$$

Denotando

$$G=\frac{\mu}{2}\|v_k-y_k\|^2+\langle\nabla f(y_k),v_k-y_k\rangle,$$

e multiplicando (2.27) por γ_{k+1} temos

$$\gamma_{k+1}\phi^*_{k+1}=\gamma_{k+1}(1-\alpha_k)f(x_k)+\gamma_{k+1}\alpha_k f(y_k)-\frac{\alpha_k^2}{2}\|\nabla f(y_k)\|^2+\alpha_k(1-\alpha_k)\gamma_k G. \quad (2.28)$$

Usando a definição de γ_{k+1} e reduzindo os termos semelhantes obtemos

$$\bar{A}\alpha_k^2+\bar{B}\alpha_k+\bar{C}=0,$$

onde

$$\begin{aligned}\bar{A}&=-\frac{1}{2}\|\nabla f(y_k)\|^2+(\gamma_k-\mu)(f(x_k)-f(y_k))-\gamma_k G,\\ \bar{B}&=(\gamma_k-\mu)(f(x_k)-\phi^*_{k+1})+\gamma_k(f(y_k)-f(x_k))+\gamma_k G,\\ \bar{C}&=\gamma_k(f(x_k)-\phi^*_{k+1}).\end{aligned}$$

Note que $\bar{A}=A$. Se $\phi^*_{k+1}:=f(x_{k+1})$, então $\bar{B}=B$, $\bar{C}=C$ e α_k é solução de (2.3).

Reciprocamente, se $\alpha_k\in(0,1)$ é solução da equação (2.3), temos que

$$(B-\bar{B})\alpha_k-(C-\bar{C})=0.$$

Substituindo $B, \bar{B}, C, \bar{C}$ na igualdade acima e simplificando obtemos

$$(f(x_{k+1}) - \phi^*_{k+1})((\gamma_k - \mu)\alpha_k - \gamma_k) = 0.$$

Se $(\gamma_k - \mu)\alpha_k - \gamma_k = 0$, temos que $\alpha_k = \frac{\gamma_k}{\gamma_k - \mu} \geq 1$, o que contradiz o fato de $\alpha_k \in (0,1]$. Logo $f(x_{k+1}) = \phi^*_{k+1}$, completando a demonstração. □

Como consequência do lema anterior vemos que o parâmetro α_k da combinação convexa das funções ϕ_k e l_k é calculado de modo que $\phi^*_{k+1} = f(x_{k+1})$. Assim

$$\phi_{k+1}(x) \geq \phi^*_{k+1} = f(x_{k+1}) \geq f(x^*). \tag{2.29}$$

A figura abaixo ilustra o que ocorre. Note que ϕ_k e l_k é dada, sendo ϕ_k uma aproximação quadrática superior de f e l_k uma aproximação quadrática inferior de f. Note também que ϕ_{k+1} é dada como combinação convexa de ϕ_k e l_k de maneira que o valor mínimo de ϕ_{k+1} seja igual a $f(x_{k+1})$.

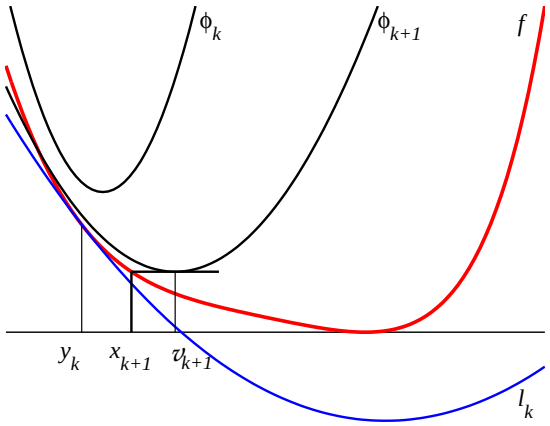

Figura 2.3: ϕ_{k+1} é uma combinação convexa de ϕ_k e l_k de modo que $\phi^*_k = f(x_{k+1})$.

Lema 2.5. *Considere as sequências geradas pelo algoritmo. Então, para todo $k \geq 0$,*

$$\alpha_k \geq \sqrt{\frac{\gamma_{k+1}}{2L}}. \tag{2.30}$$

Demonstração. Somando e subtraindo $(1-\alpha_k)f(y_k)$ em (2.16) temos

$$\phi^*_{k+1} = f(y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}}\|\nabla f(y_k)\|^2 + (1-\alpha_k)\left((f(x_k) - f(y_k)) + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}G\right), \tag{2.31}$$

onde $G = \frac{\mu}{2}\|v_k - y_k\|^2 + \langle \nabla f(y_k), v_k - y_k\rangle$.

Mas pelo algoritmo temos $f(x_k) - f(y_k) \geq 0$ e

$$\langle \nabla f(y_k), v_k - y_k \rangle = (1-\theta)\langle \nabla f(y_k), d_k \rangle \geq 0,$$

mostrando que $G \geq 0$. Logo

$$\phi^*_{k+1} \geq f(y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}} \|\nabla f(y_k)\|^2. \tag{2.32}$$

Usando a condição de decréscimo suficiente (2.9)

$$f(x_{k+1}) \leq f(y_k) - \frac{1}{4L} \|\nabla f(y_k)\|^2,$$

temos que

$$\phi^*_{k+1} \geq f(x_{k+1}) + \left(\frac{1}{4L} - \frac{\alpha_k^2}{2\gamma_{k+1}} \right) \|\nabla f(y_k)\|^2. \tag{2.33}$$

Por outro lado $\phi^*_{k+1} = f(x_{k+1})$, portanto

$$\frac{1}{4L} - \frac{\alpha_k^2}{2\gamma_{k+1}} \leq 0,$$

ou seja,

$$\alpha_k \geq \sqrt{\frac{\gamma_{k+1}}{2L}},$$

completando a prova. □

2.1.3 Prova de complexidade ótima

Nessa seção mostraremos que o Algoritmo 1 é ótimo. Iniciaremos com dois lemas técnicos, o segundo provado em [Nes04, GK08].

Lema 2.6. *Considere a e b dois números reais com $a \geq b > 0$. Então $a\sqrt{b} \geq b\sqrt{a}$.*

Demonstração. Multiplicando a desigualdade $a \geq b$ por (ab), temos $a^2 b \geq b^2 a$. Extraindo a raiz quadrada , temos o resultado. □

Lema 2.7. *Considere a sequência positiva $\{\lambda_k\}$. Assuma que existe $M > 0$ tal que $\lambda_{k+1} \leq \left(1 - M\sqrt{\lambda_{k+1}}\right)\lambda_k$. Então*

$$\lambda_k \leq \frac{4}{M^2} \frac{1}{k^2}. \tag{2.34}$$

Demonstração. Denote $a_k = \frac{1}{\sqrt{\lambda_k}}$. Como $\{\lambda_k\}$ é uma sequência decrescente temos que

$$a_{k+1} - a_k = \frac{\sqrt{\lambda_k} - \sqrt{\lambda_{k+1}}}{\sqrt{\lambda_k \lambda_{k+1}}} = \frac{\lambda_k - \lambda_{k+1}}{\lambda_k \sqrt{\lambda_{k+1}} + \lambda_{k+1}\sqrt{\lambda_k}}.$$

Usando o lema anterior com $x = \lambda_k$ e $b = \lambda_{k+1}$, temos

$$a_{k+1} - a_k \geq \frac{\lambda_k - \lambda_{k+1}}{2\lambda_k \sqrt{\lambda_{k+1}}}.$$

Aplicando a hipótese

$$a_{k+1} - a_k \geq \frac{\lambda_k - \left(1 - M\sqrt{\lambda_{k+1}}\right)\lambda_k}{2\lambda_k \sqrt{\lambda_{k+1}}} = \frac{M}{2}.$$

Logo

$$a_k \geq a_{k-1} + \frac{M}{2} \cdots \geq a_0 + \frac{Mk}{2} \geq \frac{Mk}{2}.$$

Portanto $\lambda_k \leq \frac{4}{M^2}\frac{1}{k^2}$, completando a demonstração. □

Lema 2.8. *Considere $\gamma_0 > \mu$. Seja x^* o minimizador de f. Então, em todas as iterações do Algoritmo 1,*

$$\phi_k(x^*) - f(x^*) \leq \frac{\gamma_0 + L}{\gamma_0 - \mu}\frac{\|x^* - x_0\|^2}{2}\left(\gamma_k - \mu\right). \tag{2.35}$$

Demonstração. A prova será feita por indução. Considere $k = 0$. Então temos

$$\begin{aligned} \phi_0(x^*) - f(x^*) &= \phi_0^* + \frac{\gamma_0}{2}\|x^* - v_0\|^2 - f(x^*) \\ &= f(x_0) - f(x^*) + \frac{\gamma_0}{2}\|x^* - x_0\|^2. \end{aligned}$$

Por (1.6) temos

$$f(x_0) - f(x^*) \leq \frac{L}{2}\|x^* - x_0\|^2.$$

Usando na desigualdade acima obtemos

$$\phi_0(x^*) - f(x^*) \leq \frac{L + \gamma_0}{2}\|x^* - x_0\|^2.$$

Agora suponha que vale para k, vamos mostrar que vale para $k+1$. Então,

$$\phi_{k+1}(x^*) - f(x^*) = (1-\alpha_k)\phi_k(x^*) + \alpha_k\left(f(y_k) + \langle \nabla f(y_k), x^* - y_k\rangle + \frac{\mu}{2}\|x^* - y_k\|^2\right) - f(x^*). \tag{2.36}$$

Pela convexidade de f a parte direita de (2.36) é menor ou igual a

$$(1-\alpha_k)\phi_k(x^*)+\alpha_k f(x^*)-f(x^*).$$

Donde

$$\phi_{k+1}(x^*)-f(x^*)\leq(1-\alpha_k)\left(\phi_k(x^*)-f(x^*)\right). \tag{2.37}$$

Usando a hipótese temos que

$$\begin{aligned}\phi_{k+1}(x^*)-f(x^*) &\leq (1-\alpha_k)\left(\phi_k(x^*)-f(x^*)\right)\\ &\leq (1-\alpha_k)\left(\frac{\gamma_0+L}{\gamma_0-\mu}\frac{\|x^*-x_0\|^2}{2}(\gamma_k-\mu)\right).\end{aligned} \tag{2.38}$$

Simplificando (2.38) obtemos

$$\phi_{k+1}(x^*)-f(x^*)\leq\frac{\gamma_0+L}{\gamma_0-\mu}\frac{\|x^*-x_0\|^2}{2}\left(\gamma_{k+1}-\mu\right),$$

completando a demonstração.

□

Lema 2.9. *Considere $\gamma_0>\mu$. Então para todo $k>0$ temos*

$$\gamma_k-\mu\leq\min\left\{\left(1-\sqrt{\frac{\mu}{2L}}\right)^k(\gamma_0-\mu),8L\frac{1}{k^2}\right\} \tag{2.39}$$

Demonstração. Pelo Lema 2.5, $\alpha_k\geq\sqrt{\frac{\gamma_{k+1}}{2L}}$. Em particular temos

$$\alpha_k\geq\sqrt{\frac{\gamma_{k+1}}{2L}}\geq\sqrt{\frac{\gamma_{k+1}-\mu}{2L}}, \tag{2.40}$$

e

$$\alpha_k\geq\sqrt{\frac{\gamma_{k+1}}{2L}}\geq\sqrt{\frac{\mu}{2L}}. \tag{2.41}$$

Pela definição de γ_{k+1} temos que

$$\gamma_{k+1}-\mu=(1-\alpha_k)(\gamma_k-\mu).$$

Usando (2.41) temos

$$\begin{aligned}\gamma_{k+1}-\mu &\leq \left(1-\sqrt{\frac{\mu}{2L}}\right)(\gamma_k-\mu)\\ &\leq \left(1-\sqrt{\frac{\mu}{2L}}\right)^2(\gamma_{k-1}-\mu)\leq\cdots\leq\left(1-\sqrt{\frac{\mu}{2L}}\right)^{k+1}(\gamma_0-\mu).\end{aligned}$$

Isso mostra a primeira desigualdade. Agora usando (2.40) temos

$$\begin{aligned}\gamma_{k+1}-\mu &= (1-\alpha_k)(\gamma_k-\mu)\\ &\leq \left(1-\sqrt{\frac{\gamma_{k+1}-\mu}{2L}}\right)(\gamma_k-\mu)\\ &= \left(1-\frac{1}{\sqrt{2L}}\sqrt{\gamma_{k+1}-\mu}\right)(\gamma_k-\mu).\end{aligned}$$

Tomando $\lambda_k=\gamma_k-\mu$ e $M=\frac{1}{\sqrt{2L}}$, pelo Lema 2.7 temos que

$$\gamma_k-\mu\leq\frac{4}{\left(\frac{1}{\sqrt{2L}}\right)^2}\frac{1}{k^2}=8L\frac{1}{k^2},$$

completando a demonstração. □

Teorema 2.10. *Considere $\gamma_0>\mu\geq 0$. Então o Algoritmo 1 gera uma sequência $\{x_k\}$ tal que para todo $k>0$,*

$$f(x_k)-f(x^*)\leq\min\left\{\left(1-\sqrt{\frac{\mu}{2L}}\right)^k(L+\gamma_0)\frac{\|x^*-x_0\|^2}{2},\frac{4L\|x^*-x_0\|^2(\gamma_0+L)}{\gamma_0-\mu}\frac{1}{k^2}\right\}. \tag{2.42}$$

Demonstração. Por construção, $f(x_k)\leq\phi_k(x)$ para todo $x\in\mathbb{R}^n$, em particular para x^*. Usando o Lema 2.8 e 2.9 temos

$$\begin{aligned}f(x_k)-f(x^*) &\leq \phi_k(x^*)-f(x^*)\\ &\overset{L\,2.8}{\leq} \frac{\gamma_0+L}{\gamma_0-\mu}\frac{\|x^*-x_0\|^2}{2}(\gamma_k-\mu)\\ &\overset{L\,2.9}{\leq} \frac{\gamma_0+L}{\gamma_0-\mu}\frac{\|x^*-x_0\|^2}{2}\min\left\{\left(1-\sqrt{\frac{\mu}{2L}}\right)^k(\gamma_0-\mu),8L\frac{1}{k^2}\right\}\\ &= \min\left\{\left(1-\sqrt{\frac{\mu}{2L}}\right)^k(L+\gamma_0)\frac{\|x^*-x_0\|^2}{2},\frac{4L\|x^*-x_0\|^2(\gamma_0+L)}{\gamma_0-\mu}\frac{1}{k^2}\right\},\end{aligned}$$

completando a demonstração. □

Esse teorema mostrou que o Algoritmo 1 é ótimo para as classes $\mathscr{F}$ e $\mathscr{F}_\mu$, pois a menos de constantes, a cota de complexidade superior das classes $\mathscr{F}$ e $\mathscr{F}_\mu$ é da ordem de $\frac{1}{\sqrt{\varepsilon}}$ e $\ln\frac{1}{\varepsilon}$ respectivamente , sendo proporcional as cotas de complexidade inferior de $\mathscr{F}$ e $\mathscr{F}_\mu$ mostradas no capítulo 1.

Para ver isso, note que se queremos que ocorra $f(x_k) - f^* < \varepsilon$ devemos ter

$$\min\left\{\left(1-\sqrt{\frac{\mu}{2L}}\right)^k (L+\gamma_0)\frac{\|x^*-x_0\|^2}{2}, \frac{4L\|x^*-x_0\|^2(\gamma_0+L)}{\gamma_0-\mu}\frac{1}{k^2}\right\} < \varepsilon.$$

Isolando k e desconsiderando as constantes, obtemos $k = \mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$ se $\mu = 0$ e $k = \mathcal{O}\left(\ln\frac{1}{\varepsilon}\right)$ se $\mu \neq 0$.

2.2 Algoritmo ótimo com parâmetro de convexidade desconhecido

Nem sempre o parâmetro de convexidade é conhecido, então é necessário adaptar o Algoritmo 1 para que ele seja utilizado em caso isso ocorra. O algoritmo é essencialmente o mesmo com uma pequena modificação proposta por Gonzaga e Karas em [GK08].

Denotando por $\bar{\mu}$ (possivelmente desconhecido) o parâmetro de convexidade de f, o algoritmo constrói uma sequência μ_k convergente para $\bar{\mu}$ de modo que em cada iteração tenhamos $\mu_k \leq \tilde{\mu}$ com

$$\tilde{\mu} = \frac{\|\nabla f(y_k)\|^2}{2(f(y_k) - f(x_{k+1}))}, \tag{2.43}$$

como sugerido em [GK08] e $\mu_k \leq \gamma_k$.

Esta última condição vem do fato de que, por (2.12) l_k é uma aproximação quadrática inferior de f com parâmetro de convexidade $\mu = \mu_k$ e pelo Lema 2.4 temos que ϕ_k é uma aproximação quadrática superior de f com parâmetro de convexidade γ_k. Assim, devemos ter sempre $\mu_k \leq \gamma_k$.

O algoritmo parte de $\mu_0 > 0$ dado e gera uma sequência μ_k de forma que $\mu_k \to \bar{\mu}$. Para tanto, se na iteração k, o valor corrente μ_k verifica $\mu_k \geq \gamma_k$ ou $\mu_k \geq \tilde{\mu}$ é sinal que está grande e devemos diminuí-lo. Em suma temos o algoritmo.

Algoritmo 2. *Gonzaga-Karas*

Dado $x_0 \in \mathbb{R}^n$, $v_0 = x_0$, $\gamma_0 > \mu, \beta > 1, \bar{\mu} = \mu, \mu_0 \in [\bar{\mu}, \gamma_0)$.

k=0

Repita

$d_k = v_k - x_k$.

Se $f(v_k) \leq f(x_k)$ *então* $y_k = v_k$, *senão*

Escolha $\theta \in [0,1)$ *e* $y_k = x_k + \theta d_k$ *sujeito a* $f(y_k) \leq f(x_k)$ *e* $\langle \nabla f(y_k), d_k \rangle \geq 0$.

Calcule $x_{k+1} = y_k - \lambda \nabla f(y_k)$, *satisfazendo*

$f(x_{k+1}) \leq f(y_k) - \frac{1}{4L}\|\nabla f(y_k)\|^2$.

Se $(\gamma_k - \bar{\mu}) < \beta(\mu_k - \bar{\mu})$, *então escolha* $\mu_k \in \left[\bar{\mu}, \frac{\gamma_k}{\beta}\right]$.
Calcule $\tilde{\mu} = \frac{\|\nabla f(y_k)\|^2}{2(f(y_k) - f(x_{k+1}))}$.
Se $\mu_k > \tilde{\mu}$, *então* $\mu_k = \max\left\{\bar{\mu}, \frac{\tilde{\mu}}{10}\right\}$.
$G = \frac{\mu_k}{2}\|y_k - v_k\|^2 + \langle \nabla f(y_k), v_k - y_k\rangle$,
$A = -\frac{1}{2}\|\nabla f(y_k)\|^2 + (\gamma_k - \mu_k)(f(x_k) - f(y_k)) - \gamma_k G$,
$B = (\gamma_k - \mu_k)(f(x_{k+1}) - f(x_k)) + \gamma_k(f(y_k) - f(x_k)) + \gamma_k G$,
$C = \gamma_k(f(x_k) - f(x_{k+1}))$.
Compute $\alpha_k \in (0,1)$ *solução de* $A\alpha^2 + B\alpha + C = 0$.
$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu_k$.
$v_{k+1} = \frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu_k y_k - \nabla f(y_k)))$.
$k = k+1$.

Fim

O algoritmo está bem definido, pois é o mesmo da seção anterior, com modificação do cálculo de μ_k que é sempre possível de ser feita. A figura 2.4 ilustra o que ocorre. Na cor azul, a curva dada pela equação (2.12) com $\tilde{\mu}$ no lugar de μ. A curva verde é a equação (2.12) com $\mu_k > \tilde{\mu}$, o que foge da estrutura do algoritmo. Já a curva em amarelo, é a equação (2.12) com μ_k aceitável. Lembrando que (2.12) é a aproximação quadrática em torno de y_k. No algoritmo deseja-se que ela seja uma aproximação inferior de f, o que ocorre com a curva em amarelo. Quando conheçemos μ, o algoritmo torna-se o mesmo que o anterior, se não conhecermos μ, basta colocar $\bar{\mu} = 0$.

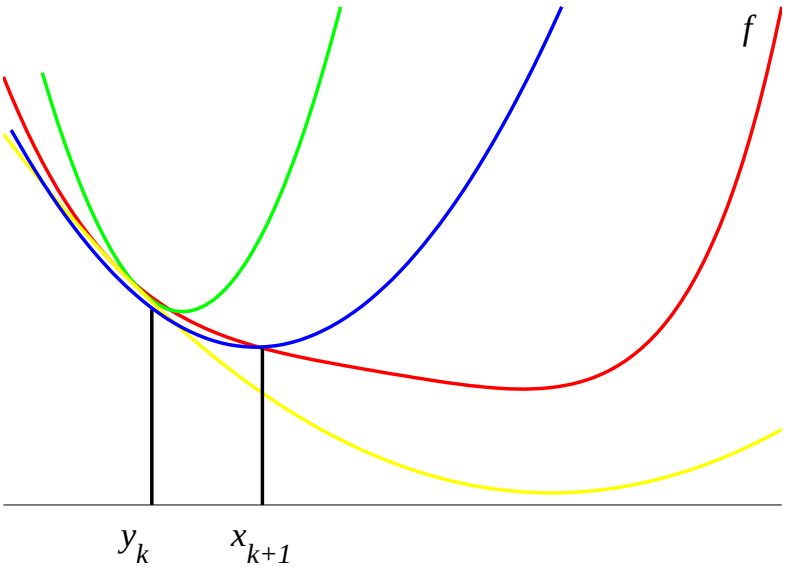

Figura 2.4: Situação em que μ é desconhecido

A prova de complexidade é essencialmente a mesma da seção anterior. No Lema 2.8, μ torna- se $\bar{\mu}$. O Lema 2.9 tem uma pequena modificação. Então enunciaremos novamente.

Lema 2.11. *Para todo $k \geq 0$, $(\gamma_k - \bar{\mu}) \geq \beta(\mu_k - \bar{\mu})$.*

Demonstração. Se no começo da iteração k, temos $(\gamma_k - \bar{\mu}) > \beta(\mu_k - \bar{\mu})$, escolhemos $\mu_k \in \left[\bar{\mu}, \frac{\gamma_k}{\beta}\right]$ com $\beta > 1$. Então

$$\beta(\mu_k - \bar{\mu}) \leq \beta\left(\frac{\gamma_k}{\beta} - \bar{\mu}\right) = \gamma_k - \beta\bar{\mu} \leq \gamma_k - \bar{\mu}.$$

□

Lema 2.12. *Considere $\gamma_0 > \bar{\mu}$. Então, para todo $k > 0$,*

$$\gamma_k - \bar{\mu} \leq \min\left\{\left(1 - \frac{\beta-1}{\beta}\sqrt{\frac{\bar{\mu}}{2L}}\right)^k(\gamma_0 - \bar{\mu}), \frac{8\beta^2 L}{(\beta-1)^2}\frac{1}{k^2}\right\}. \tag{2.44}$$

Demonstração. Note que agora temos $\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu_k$. Somando e subtraindo $\alpha_k\bar{\mu}$ temos

$$\gamma_{k+1} - \bar{\mu} = (1-\alpha_k)(\gamma_k - \bar{\mu}) + \alpha_k(\mu_k - \bar{\mu}). \tag{2.45}$$

Logo usando o lema anterior na igualdade (2.45) temos

$$\begin{aligned} \gamma_{k+1} - \bar{\mu} &\leq (1-\alpha_k)(\gamma_k - \bar{\mu}) + \frac{\alpha_k}{\beta}(\gamma_k - \bar{\mu}) \\ &= \left(1 - \frac{\beta-1}{\beta}\alpha_k\right)(\gamma_k - \bar{\mu}). \end{aligned} \tag{2.46}$$

Novamente, usando o Lema 2.5 e o fato que $\gamma_{k+1} \geq \bar{\mu}$, temos

$$\alpha_k \geq \sqrt{\frac{\bar{\mu}}{2L}}, \tag{2.47}$$

e

$$\alpha_k \geq \sqrt{\frac{\gamma_{k+1} - \bar{\mu}}{2L}}. \tag{2.48}$$

Então usando (2.46) com (2.47) junto com recorrência temos

$$\gamma_{k+1} - \bar{\mu} \leq \left(1 - \frac{\beta-1}{\beta}\sqrt{\frac{\bar{\mu}}{2L}}\right)(\gamma_k - \bar{\mu}) \leq \left(1 - \frac{\beta-1}{\beta}\sqrt{\frac{\bar{\mu}}{2L}}\right)^{k+1}(\gamma_0 - \bar{\mu}),$$

demonstrando a primeira desigualdade.

Por outro lado, por (2.48) temos

$$\gamma_{k+1} - \bar{\mu} \leq \left(1 - \frac{\beta - 1}{\beta\sqrt{2L}}\sqrt{\gamma_{k+1} - \bar{\mu}}\right)(\gamma_k - \bar{\mu}). \tag{2.49}$$

Usando o Lema 2.7 em (2.49) com $\lambda_k = \gamma_k - \bar{\mu}$ e $M = \frac{\beta-1}{\beta\sqrt{2L}}$ temos

$$\gamma_k - \bar{\mu} \leq \frac{4}{\left(\frac{\beta-1}{\beta\sqrt{2L}}\right)^2}\frac{1}{k^2} = \frac{8\beta^2 L}{(\beta-1)^2}\frac{1}{k^2},$$

completando a demonstração. □

E por fim, temos o seguinte teorema.

Teorema 2.13. *Considere $\gamma_0 > \bar{\mu} \geq 0$. Então o Algoritmo 2 gera uma sequência $\{x_k\}$ tal que para todo $k > 0$,*

$$f(x_k) - f(x^*) \leq \min\left\{\left(1 - \frac{\beta-1}{\beta}\sqrt{\frac{\bar{\mu}}{2L}}\right)^k \|x^* - x_0\|^2 \frac{L+\gamma_0}{2}, \frac{4\beta^2 L\|x^* - x_0\|^2(L+\gamma_0)}{(\beta-1)^2(\gamma_0 - \bar{\mu})}\frac{1}{k^2}\right\}. \tag{2.50}$$

A prova desse teorema, essencialmente é a mesma do Teorema 2.10, portanto omitida aqui. Os resultados obtidos também são análogos da Seção 2.1.

2.3 Convergência global

Os algoritmos estudados produzem sequências $\{x_k\}$ e $\{y_k\}$ tais que $f(x_{k+1}) \leq f(y_k) \leq f(x_k)$, e cada x_{k+1} é obtido como um passo de Cauchy a partir de y_k. Então, se considerarmos a sequência $x_1, y_1, x_2, y_2, \ldots$, temos um algoritmo de descida, com passos em direções de máxima descida. É conhecido na literatura que algoritmos com essa propriedade são globalmente convergentes, no sentido de que todo ponto de acumulação da sequência gerada pelo algoritmo é estacionário, ou seja, satisfaz a condição necessária de otimalidade. Nesta seção mostraremos a convergência global dos algoritmos como feito em [GK08].

Teorema 2.14. *Sejam $\{x_k\}$ e $\{y_k\}$ sequências geradas pelos algoritmos, e assuma que para todo $k \in N$,*

i) $f(y_k) \leq f(x_k)$,

ii) $f(x_{k+1}) \leq f(y_k) - \frac{1}{4L}\|\nabla f(y_k)\|^2$.

Então todo ponto de acumulação $\bar{y}$ de $\{y_k\}$ é estacionário.

Demonstração. Considere uma subsequência $\{y_{k_i}\}$ da sequência $\{y_k\}$ tal que $y_{k_i} \to \bar{y}$. Por construção $f(y_{k+1}) \leq f(y_k)$ para todo k, $f(y_{k_i}) \to f(\bar{y})$ e $f(x_{k_i}) \to f(\bar{y})$. Então por (ii),

$$\|\nabla f(y_{k_i})\|^2 \leq 4L(f(y_{k_i}) - f(x_{k_i+1})).$$

Passando o limite e usando a continuidade do gradiente temos $\|\nabla f(\bar{y})\| = 0$, completando a demonstração. □

Note que o resultado de convergência global acima foi obtido para funções não necessariamente convexas, mas com constante de Lipschitz para o gradiente.

Capítulo 3

Testes Computacionais

Esse capítulo é dedicado a testes computacionais de alguns métodos conhecidos na literatura para problemas de minimização de funções convexas e fortemente convexas. Comparamos o desempenho dos modelos ótimos do capítulo anterior com o modelo de Nesterov proposto em [Nes04] e em [Nes07] para funções convexas e fortemente convexas. Em especial, para funções quadráticas, como citado em [Ber99] e mostrado em [Pol87, pág 170] o método de gradiente conjugado é ótimo, então faremos uma comparação também com esse método.

3.1 Algoritmos testados

Nessa seção apresentamos alguns métodos ótimos conhecidos na literatura que serão comparados com os métodos discutidos no capítulo anterior.

O Algoritmo 1, discutido no capítulo anterior, é uma das variantes apresentadas em [GK08] que por sua vez faz um ajuste fino no método ótimo proposto por Nesterov em [Nes04] e apresentado abaixo.

Algoritmo 3 ([Nes04])**.** *Nesterov*

Dado $x_0 \in \mathbb{R}^n$, $v_0 = x_0$, $\gamma_0 > 0$.

k=0

Repita

$d_k = v_k - x_k$.

Calcule $\alpha_k \in (0,1)$ *solução positiva de*

$L\alpha^2 = (1-\alpha)\gamma_k + \alpha\mu$.

Calcule $\theta = \dfrac{\gamma_k}{\gamma_k + \alpha_k \mu}\alpha_k$.

$y_k = x_k + \theta d_k$.

Calcule $x_{k+1} = y_k - \lambda \nabla f(y_k)$, *satisfazendo*

$f(x_{k+1}) \leq f(y_k) - \frac{1}{2L}\|\nabla f(y_k)\|^2$.
$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu$.
$v_{k+1} = \frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k v_k + \alpha_k(\mu y_k - \nabla f(y_k)))$.
$k = k+1$.

Fim

Ao compararmos os Algoritmos 1 e 3, notamos que no Algoritmo 1 o parâmetro α_k é calculado como uma solução de uma equação do segundo grau que envolve informações do novo iterado x_{k+1}. Isso é feito para garantir que

$$\phi^*_{k+1} = f(x_{k+1}) \tag{3.1}$$

onde por (2.11), ϕ_{k+1} é dada pela combinação convexa com parâmetro α_k das funções ϕ_k e l_k. Por outro lado, no Algoritmo 3 o parâmetro α_k é calculado no ínicio da iteração enfraquecendo a condição (3.1) para

$$\phi^*_{k+1} \geq f(x_{k+1}),$$

mas que garante de qualquer forma a condição

$$\phi_{k+1}(x) \geq f(x^*)$$

expressa em (2.29) para todo $x \in \mathbb{R}^n$ e de fundamental importância na discussção da complexidade ótima desses algoritmos.

A mudança discutida acima no cálculo do parâmetro α_k implica uma mudança no cálculo do parâmetro θ que fornece o comprimento do passo a partir de x_k na direção $d_k = v_k - x_k$. Enquanto no Algoritmo 3, o parâmetro θ é dado por uma fórmula fechada com informação explícita de α_k, no Algoritmo 1 o parâmetro θ é dado por uma busca linear da direção d_k o que encarece o algoritmo.

No entanto, o preço dessa busca é compensado pela não necessidade do conhecimento da constante de Lipschitz L do gradiente de f. Note que o cálculo de α_k, no Algoritmo 3, depende explicitamente da constante L e μ, bem como o desempenho do algoritmo como veremos nas próximas seções.

Nesterov em [Nes07] propõe um algoritmo ótimo para minimização de funções convexas que independe do conhecimento da constante L. O algoritmo é descrito abaixo.

Algoritmo 4 ([Nes07])**.** *Nesterov*

Dado $x_0 \in \mathbb{R}^n$, $v_0 = x_0$, $N_0 > 0$, $A_0 = 0$, $\gamma_\mu > 1$, $\gamma_d \geq 1$.

k=0

Repita

$N = N_k$.

Repita

Calcule $\bar{a}$ solução de

$$Na^2 - 2a - 2A_k = 0.$$

$$y = \frac{A_k x_k + \bar{a} v_k}{A_k + \bar{a}}.$$

$$T = y - \frac{1}{N}\nabla f(y).$$

Se $\langle \nabla f(T), \nabla f(y)\rangle < \|\nabla f(T)\|^2$, *então* $N = N\gamma_\mu$.

Senão

Fim do repita.

$y_k = y$, $x_{k+1} = T$.

$a_{k+1} = \bar{a}$, $A_{k+1} = A_k + a_{k+1}$.

$N_{k+1} = \frac{N}{\gamma_d}$.

$v_{k+1} = v_k + a_{k+1}\nabla f(x_{k+1})$.

$k = k+1$.

Fim

Note que o algoritmo acima tem dois enlaces. O enlace interno fornece uma estimativa N para a constante de Lipschitz L do gradiente.

Os algoritmos de gradiente conjugado são clássicos para minimização irrestrita e são amplamente discutidos na literatura. Veja, por exemplo, [NW99]. Segundo [Ber99, Pol87] os métodos de gradiente conjugado são ótimos para minimização de funções quadráticas e portanto,farão parte dos nossos testes computacionais. Abaixo apresentamos a versão do algoritmo de gradiente conjugado implementado, discutida em [NW99, pág 120] e proposta por Polak-Ribière em [PR69].

Algoritmo 5 ([NW99])**.** *Polak-Ribière*

Dado $x_0 \in \mathbb{R}^n$, $p_0 = -\nabla f(x_0)$.

k=0

Repita

λ_k *obtida por busca linear a partir de* x_k *na direção* p_k.

$$x_{k+1} = x_k + \lambda_k p_k.$$

$$\beta_{k+1} = \frac{\langle \nabla f(x_{k+1}), \nabla f(x_{k+1}) - \nabla f(x_k)\rangle}{\|\nabla f(x_k)\|^2}.$$

$$p_{k+1} = -\nabla f(x_{k+1}) + \beta_{k+1} p_k.$$

$k = k+1$.

Fim

Nos testes computacionais, para funções quadráticas, calculamos λ_k como o resultado de uma busca linear exata a partir de x_k na direção p_k, ou seja,

$$\lambda_k = -\frac{\langle x_k, \nabla f(p_k)\rangle}{\langle p_k, \nabla f(p_k)\rangle}.$$

Enquanto que para funções não quadráticas, implementamos λ_k como resultado de uma buscar linear inexata de Armijo (ver (1.53)).

Os testes computacionais foram feitos em Matlab, com ponto inicial tomado aleatoriamente, mas sempre o mesmo para todos os algoritmos. O critério de parada utilizado em cada algoritmo está associado à condição de otimalidade de primeira ordem, ou seja, dado $\varepsilon > 0$, o algoritmo para quando obtém x_k tal que $\|\nabla f(x_k)\| < \varepsilon$. Isso ocorre, pois na prática nem sempre o valor ótimo é conhecido. Caso conhecemos o valor ótimo, podemos usar o critério de parada citado na página 3. Em nossos testes tomamos $\varepsilon = 10^{-5}$.

3.2 Testes númericos com funções quadráticas

A função quadrática $f : \mathbb{R}^n \to \mathbb{R}$ é dada por

$$f(x) = \frac{1}{2}\langle Qx, x\rangle, \tag{3.2}$$

onde Q é uma matriz definida positiva gerada aleatoriamente, a partir da dimensão dada. Aqui conhecemos μ e L que são o menor e o maior autovalor de Q respectivamente. Como μ e L são conhecidos, é possível rodar os algoritmos com os dados originais e depois simulando μ e L desconhecidos para comparar o desempenho de ambos. Nessa seção, comparamos os 5 algoritmos discutidos nessa dissertação. É conhecido e provado na literatura [IS07, NW99] que o método do gradiente conjugado para função quadrática com a matriz hessiana definida positiva tem um comportamento muito bom, converge em no máximo n iterações, onde n é a dimensão do espaço em questão. Esse algoritmo é ótimo para funções quadráticas como mostrado em [Ber99, Pol87].

Exemplo 1. Simulamos agora um problema quadrático para a dimensão 5. A princípio, fornecemos aos algoritmos os dados originais do problema, ou seja, μ como sendo o menor autovalor de Q e L sendo o maior autovalor de Q. Depois, simulamos o caso de μ e L ser desconhecido para os algoritmos. Comparamos os algoritmos, com a mesma matriz Q e o mesmo ponto inicial para ambos os casos.

Caso μ e L conhecidos

Aqui fornecemos aos algoritmos, os dados originais do problema. As Figuras 3.1 e 3.2 ilustram o comportamento dos algoritmos. A figura da esquerda exibe a variação do valor da função ao longo das iterações, enquanto a figura da direita mostra em escala logarítmica, a variação da norma do gradiente ao longo das iterações de cada um dos algoritmos.

Como μ e L são conhecidos, os Algoritmos 1 e 2 se coincidem, portanto é mostrado só um deles nas figuras. Como era de se esperar, o Algoritmo 5 resolve em 5 iterações, o Algoritmo 2

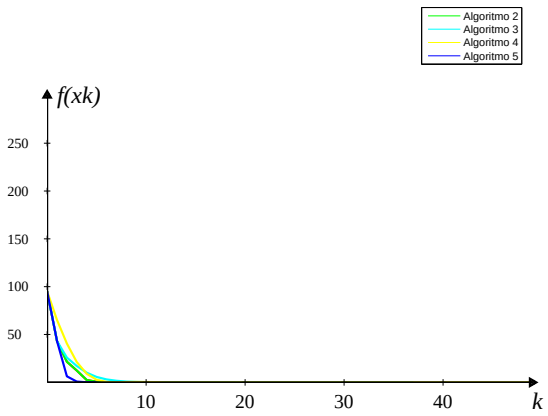

Figura 3.1: Caso em que $n = 5$

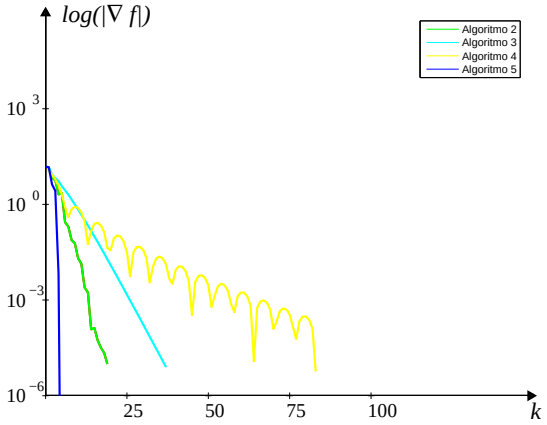

Figura 3.2: Caso em que $n = 5$

em 19, o Algoritmo 3 tem um bom desempenho com 37 iterações, pois faz o uso da constante de Lipschitz L conhecida, e o Algoritmo 4 gasta 83 iterações.

Caso μ e L desconhecidos

Fornecemos aos algoritmos $\mu = 0$ e $L = 10000$. Os resultados são mostrados nas Figuras 3.3 e 3.4.

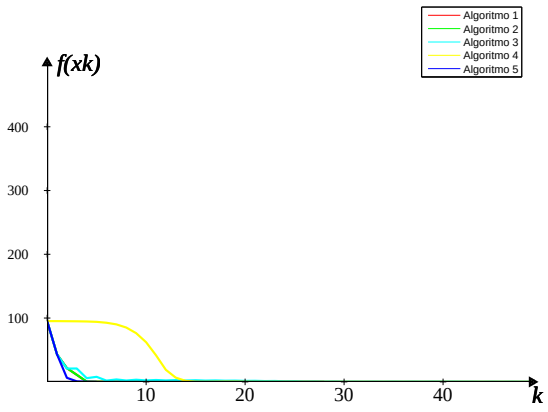

Figura 3.3: Caso em que $n = 5$

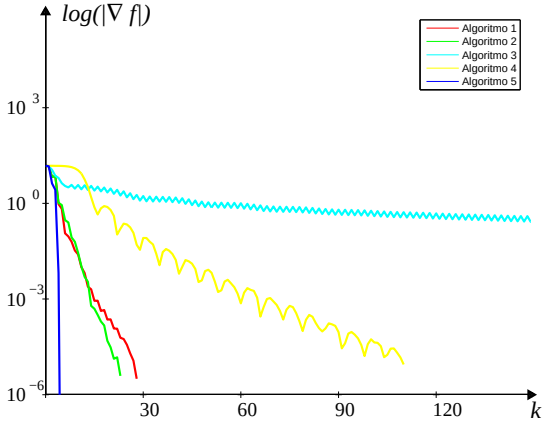

Figura 3.4: Caso em que $n = 5$

Como esperado, o Algoritmo 5 resolve o problema em 5 iterações. Por outro lado, o Algoritmo 2 gasta 23 iterações, o Algoritmo 1 gasta 28 iterações e o Algoritmo 4 gasta 110 iterações. Em relação ao caso anterior, não há um acréscimo muito significativo de iterações. Porém o Algoritmo 3 gasta 11857 iterações cerca de 320 vezes a mais iterações do que no caso de conhecer L. Vemos que o conhecimento de L é uma grande vantagem para o Algoritmo 3.

Exemplo 2. Simulamos agora um problema quadrático para a dimensão 1000. Nesse caso o cálculo da matriz Q exige mais memória. Comparamos os algoritmos, com a mesma matriz Q e o mesmo ponto inicial para ambos os casos.

Caso μ e L conhecidos

Aqui fornecemos aos algoritmos, os dados originais do problema. A Figura 3.5 e 3.6 ilustram o que ocorrem.

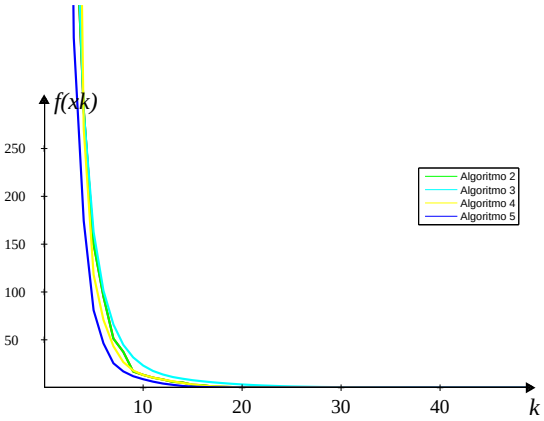

Figura 3.5: Caso em que $n = 1000$

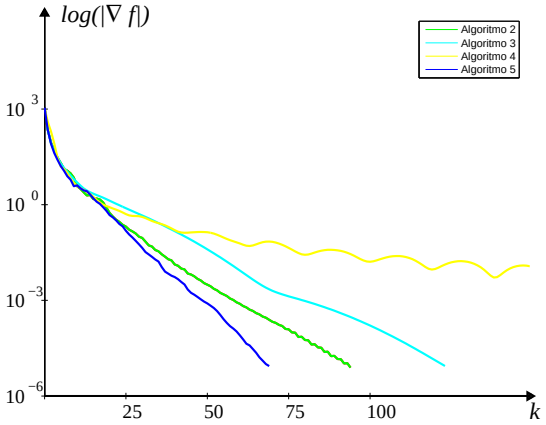

Figura 3.6: Caso em que $n = 1000$

O Algoritmo 5 resolve o problema em 69 iterações. O Algoritmo 2 gasta 94 iterações e o Algoritmo 3 gasta 123 iterações. O Algoritmo 4 tem o desempenho ruim, gasta 1528 iterações.

Caso μ e L desconhecidos

Fornecemos aos algoritmos $\mu = 0$ e $L = 10000$. Os resultados são mostrados nas Figuras 3.7 e 3.8.

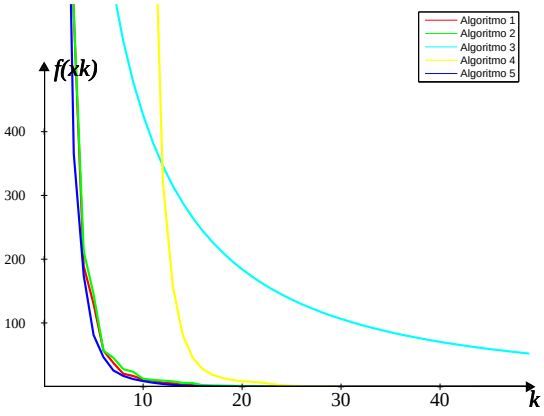

Figura 3.7: Caso em que $n = 1000$

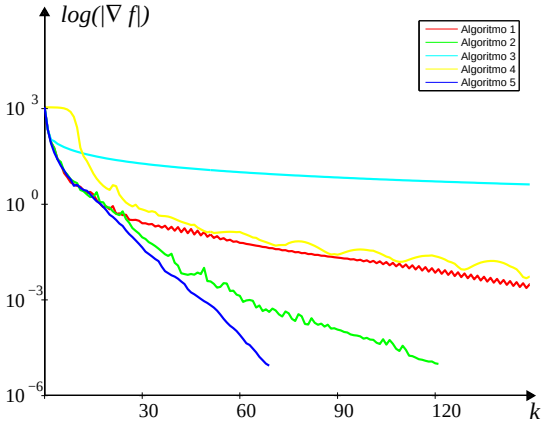

Figura 3.8: Caso em que $n = 1000$

Nesse caso, o Algoritmo 5 mantém seu desempenho, continua resolvendo o problema em 69 iterações. Isso ocorre, pois ele não usa as informações de μ e L. O Algoritmo 2, que faz uma estimativa para o parâmetro de convexidade, tem um desempenho bom. Resolve em 121 iterações. O Algoritmo 1 gasta 320 iterações, enquanto que o Algoritmo 4 gasta 720 iterações. Já o Algoritmo 3 tem o pior desempenho, resolve o problema em 12709 iterações.

Perfil de desempenho

Nessa seção apresentamos um perfil de desempenho dos algoritmos em função do número de iterações, segundo Dolan e Moré [DM02]. Para tanto, resolvemos problemas quadráticos em diferentes dimensões escolhidas aleatoriamente. Rodamos os algoritmos para o caso de μ e L conhecido e depois desconhecidos. Rodamos 5000 problemas para as funções quadráticas com a dimensão aleatória variando entre 1 e 2095 para o caso de μ e L conhecidos. Isso é mostrado na Figura 3.9. Para o caso de μ e L desconhecidos, rodamos 5046 problemas com dimensão aleatória variando entre 1 e 1178. A Figura 3.10 mostra o resultado. Note que as dimensões usadas não são tão grandes, isso ocorre pelo fato do cálculo da matriz Q exigir muita memória.

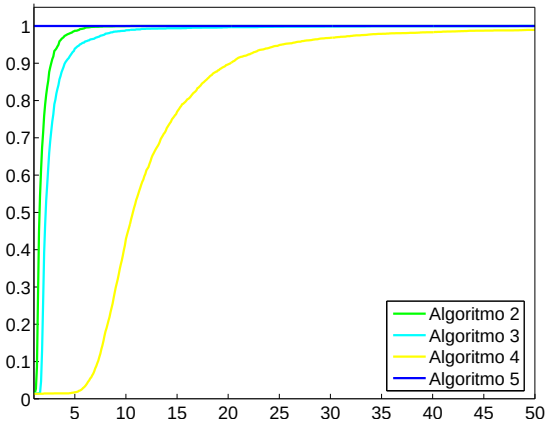

Figura 3.9: μ e L conhecidos

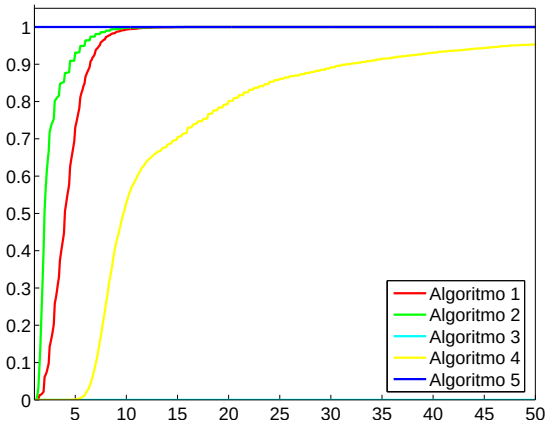

Figura 3.10: μ e L desconhecidos

A Figura 3.9 exibe o perfil de desempenho dos algoritmos em função do número de iterações no caso em que μ e L são conhecidos. Note que não aparece informação do Algoritmo 1 pois, nesse caso, é análogo ao Algoritmo 2 e ambos possuem o mesmo desempenho. Pela figura vemos que o Algoritmo 5 gasta menos iterações que os demais em 100% dos problemas. Por outro lado, o Algoritmo 2 resolve 100% dos problemas gastando não mais que 6,6 vezes iterações que o melhor algoritmo. O Algoritmo 3 pode gastar até 23 vezes iterações que o melhor algoritmo. O Algoritmo 4 perde em 100% dos problemas. Para resolver 50% dos problemas, o Algoritmo 4 gasta até 11 vezes o número de iterações usado pelo melhor algoritmo.

Já a Figura 3.10 exibe o perfil de desempenho dos algoritmos no caso em que μ e L são desconhecidos. Vemos também que o Algoritmo 5 ganha em 100% dos problemas. Por outro lado, o Algoritmo 2 resolve 100% dos problemas gastando não mais que 9 vezes o número de iterações que o Algoritmo 5. O Algoritmo 1 pode gastar até 12 vezes. Em 70% dos problemas o Algoritmo 4, que teve um desempenho inferior no caso de μ e L conhecidos, gasta até 14 vezes o número de iterações usado pelo melhor algoritmo. No entanto, o Algoritmo 3 que teve um desempenho razoável no caso de μ e L conhecido, agora perde em 100% dos

problemas.

3.3 Testes númericos com funções convexas diversas

Essa seção é dedicada a testes numéricos com os quatro algoritmos ótimos para funções convexas discutidos nessa dissertação: Algoritmo 1, Algoritmo 2, Algoritmo 3, Algoritmo 4 e o método de gradientes conjugados denotado por Algoritmo 5. Apresentamos algumas funções convexas e para cada uma delas discutimos o comportamento desses algoritmos para minimizá-las em diferentes dimensões. Ao final da seção apresentamos a análise de desempenho segundo Dolan e Moré [DM02].

Exemplo 3. A primeira função a ser testada é a função proposta por Nesterov em [Nes04, pág 67] para dimensão infinita, aqui adaptada para dimensão finita. Dados $\mu > 0$ e $L > \mu$, seja $Q = \frac{L}{\mu}$. Considere a função $f : \mathbb{R}^n \to \mathbb{R}$ dada por

$$f(x) = \frac{\mu(Q-1)}{8}\left(x_1^2 + \sum_{i=1}^{n-1}(x_i - x_{i+1})^2\right) + \frac{\mu}{2}\|x\|^2. \tag{3.3}$$

Nos testes, tomamos $\mu = 1$ e $L = 10$. Simularemos o caso de μ e L conhecidos e depois desconhecidos. Para ambos os casos, simulamos um problema de dimensão pequena e outro de dimensão maior.

Caso μ e L conhecidos

Simularemos primeiramente um problema com dimensão 5 e posteriormente um com dimensão 500. Nesse caso, fornecemos aos algoritmos o valor $\mu = 1$ e $L = 10$ que são o parâmetro de convexidade e a constante de Lipschitz para o gradiente.

Como μ e L são conhecidos, o Algoritmo 1 e o Algoritmo 2 são os mesmos. Então nesse caso, só comparamos quatro algoritmos: Algoritmo 2, Algoritmo 3, Algoritmo 4 e Algoritmo 5.

Para resolver o problema na dimensão 5, o Algoritmo 2 gastou 25 iterações, o Algoritmo 3 gastou 36 iterações , o Algoritmo 4 gastou 99 iterações e o Algoritmo 5 gastou 22 iterações. As Figuras 3.11 e 3.12 mostram as 75 primeiras iterações.

Já se aumentarmos a dimensão, os Algoritmos 2 e 3 tem quase o mesmo desempenho em relação a dimensão 5. O Algoritmo 2 gastou 31 iterações enquanto que o Algoritmo 3 gastou 50 iterações para resolver o problema como mostrado nas Figuras 3.13 e 3.14. Mas o Algoritmo 4 gastou 1312 iterações para resolver o problema. O Algoritmo 5 resolveu em 30 iterações.

Em ambas as dimensões 5 e 500, o comportamento do Algoritmo 4 teve um desempenho inferior aos demais justificado provavelmente em função de que o algoritmo não faz uso do

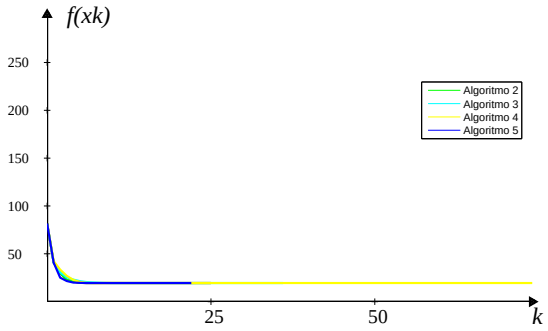

Figura 3.11: Caso em que $n = 5$

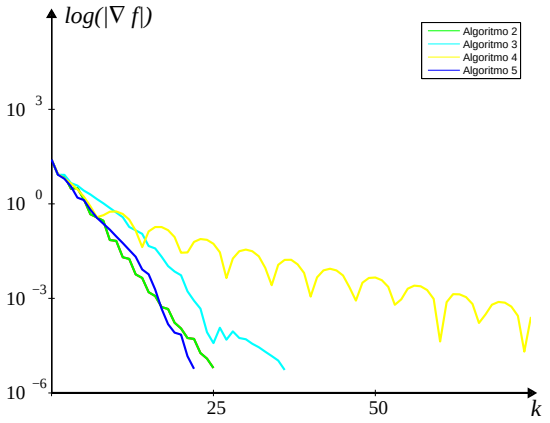

Figura 3.12: Caso em que $n = 5$

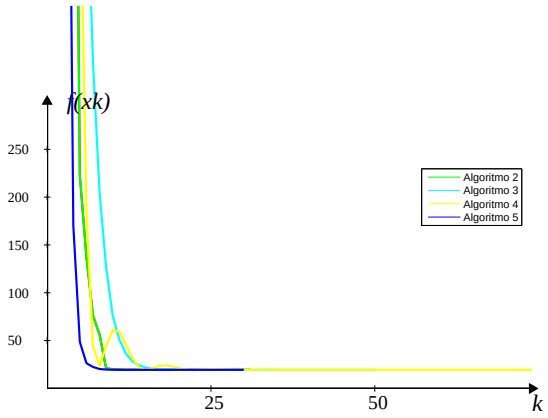

Figura 3.13: Caso em que $n = 500$

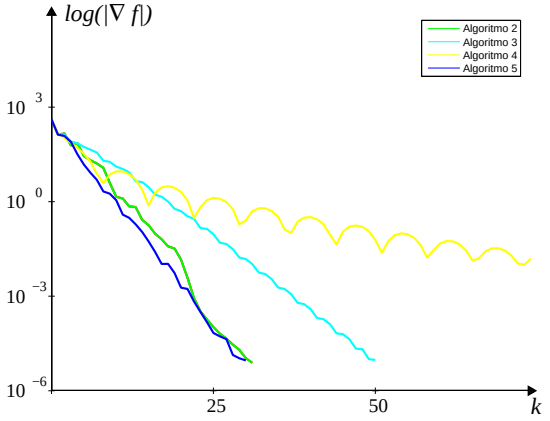

Figura 3.14: Caso em que $n = 500$

conhecimento do parâmetro de convexidade μ.

Caso μ e L desconhecidos

Simularemos primeiramente um problema com dimensão 5 e posteriormente um com dimensão 1000. Como na função, $\mu = 1$ e $L = 10$, fornecemos aos algoritmos $\mu = 0$ e $L = 100000$. É importante fornercer valores grandes para L, pela própria construção dos algoritmos, pois $\gamma_0 \geq L$ e $\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu$.

Quando μ e L são desconhecidos, o Algoritmo 1 e 2 tem desempenho diferentes. Para resolver o problema com dimensão 5, o Algoritmo 1 gastou 29 iterações, o Algoritmo 2 foi um pouco melhor com 25 iterações, o Algoritmo 4 gastou 116 iterações e o Algoritmo 3 gastou 58597 iterações para resolver o problema. Como era de se esperar, pois f é quadrática, o Algoritmo 5 resolveu o problema em 20 iterações. As Figuras 3.15 e 3.16 mostram as 75 primeiras iterações. Note que o desempenho do Algoritmo 3 foi muito inferior aos demais, isso ocorre pelo fato dele depender da constante de Lipschitz L do gradiente para o cálculo de α_k já no início do algoritmo.

Já para a dimensão 1000, o desempenho dos Algoritmos 1,2, 4 e 5 são semelhantes ao

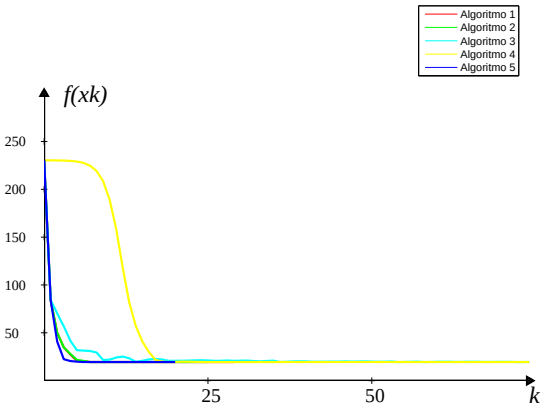

Figura 3.15: Caso em que $n = 5$

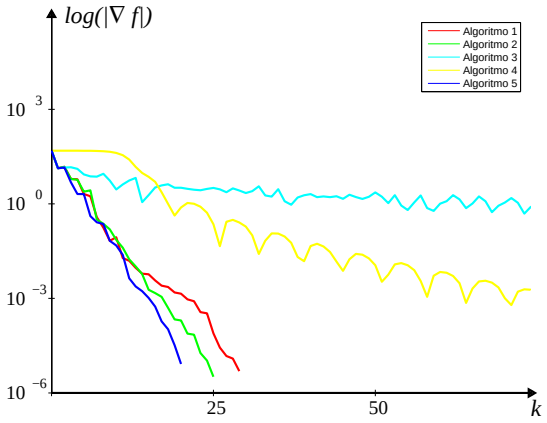

Figura 3.16: Caso em que $n = 5$

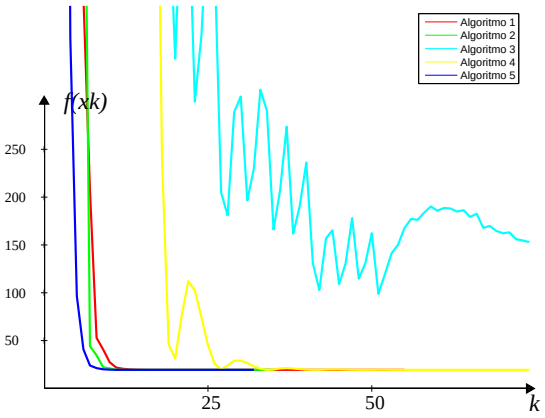

Figura 3.17: Caso em que $n = 1000$

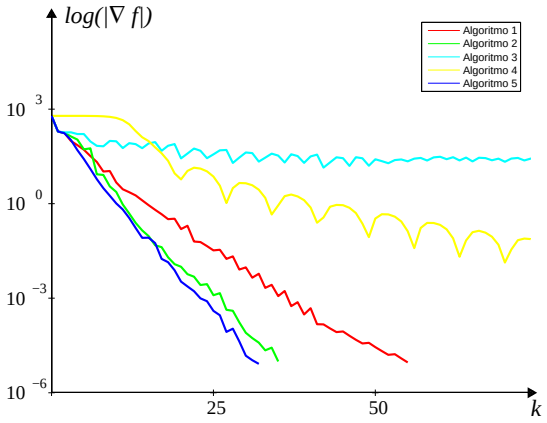

Figura 3.18: Caso em que $n = 1000$

caso de dimensão 5 com 55, 35, 179 e 32 iterações respectivamente. Mas o Algoritmo 3 gasta 75926 iterações para resolver o problema. Quanto mais se aumenta a dimensão do problema, pior o desempenho do Algoritmo 3 . As Figuras 3.17 e 3.18 mostram as 75 primeiras iterações.

Exemplo 4. Considere agora a função objetivo $f : \mathbb{R}^n \to \mathbb{R}$ dada por

$$f(x) = \sum_{i=1}^{n} e^{x_i}. \tag{3.4}$$

Para esta função, por não conhecer μ e L, fornecemos aos algoritmos os valores $\mu = 0$ e $L = 100000$. Simulamos um problema com dimensão 2 e posteriormente com dimensão 10000.

Todos os algoritmos resolvem o problema em menos de 100 iterações como mostram as Figuras 3.19 e 3.20. Neste caso, o Algoritmo 1 foi melhor que o Algoritmo 2, gastou 11 iterações, enquanto que o Algoritmo 2 gastou 13 iterações. O Algoritmo 3 gastou 73 iterações

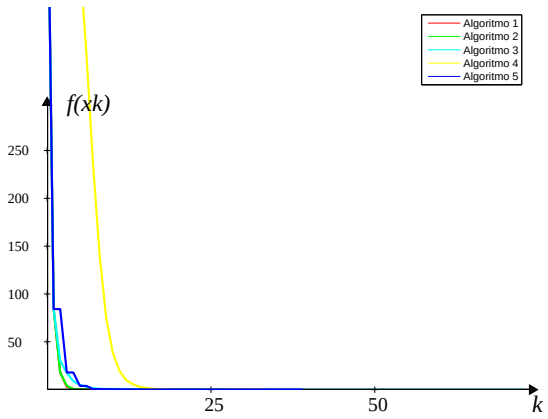

Figura 3.19: Caso em que $n = 2$

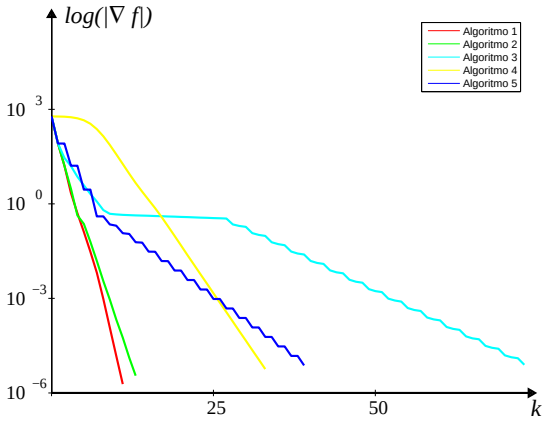

Figura 3.20: Caso em que $n = 2$

, o Algoritmo 4, 33 iterações e o Algoritmo 5 gastou 39 iterações.

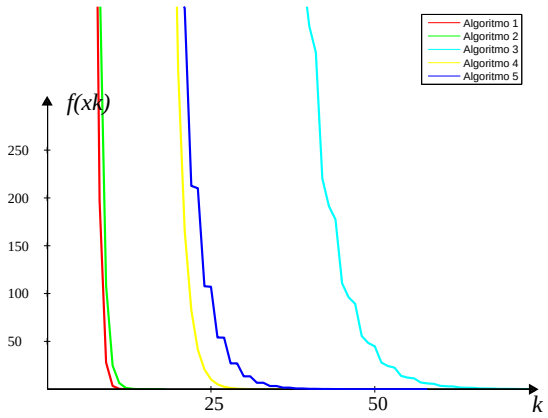

Figura 3.21: Caso em que $n = 10000$

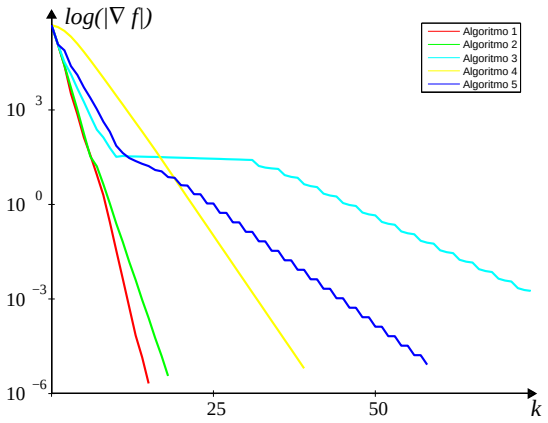

Figura 3.22: Caso em que $n = 10000$

Se aumentarmos a dimensão, os desempenhos não mudam muito. O Algoritmo 1 continua ganhando com 15 iterações, o Algoritmo 2 resolve em 18 iterações, o Algoritmo 3 em 96 iterações , o Algoritmo 4 em 39 iterações e o Algoritmo 5 em 58 iterações. Novamente o Algoritmo 3 tem um desempenho inferior por não conhecer a constante de Lipschitz L. As Figuras 3.21 e 3.22 mostram as 75 primeiras iterações para o caso de dimensão 10000. Note que nesses problemas, o Algoritmo 5 não foi o melhor.

Exemplo 5. Considere agora a função objetivo $f : \mathbb{R}^n \to \mathbb{R}$ dada por

$$f(x) = \frac{1}{2}\|x\|^2 + \sum_{i=1}^{n} e^{x_i} - 1. \tag{3.5}$$

Para essa função, não conhecemos o valor de μ e L. Nesse caso, os dados fornecidos aos algoritmos são $\mu = 0$ e $L = 100000$. Comparamos inicialmente o comportamento dos algoritmos para a dimensão 5 e posteriormente para a dimensão 500.

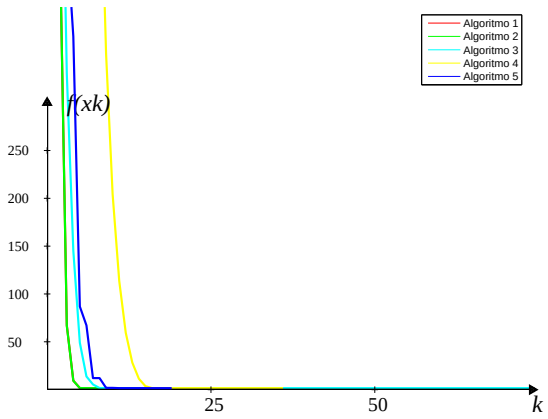

Figura 3.23: Caso em que $n = 5$

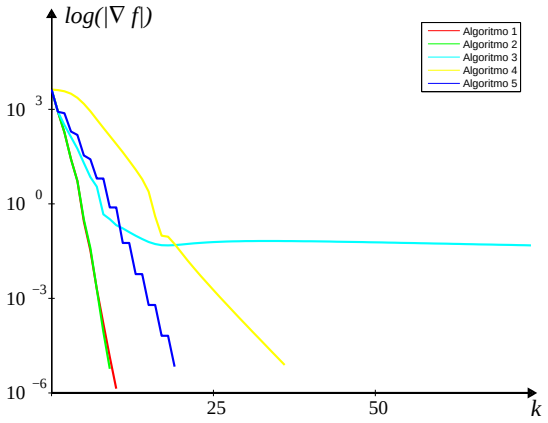

Figura 3.24: Caso em que $n = 5$

As Figuras 3.23 e 3.24 mostram as 75 primeiras iterações. Para esse problema, os Algoritmos 1 e 2 tem praticamente o mesmo desempenho. O Algoritmo 1 gasta 10 iterações e o Algoritmo 2, 9 iterações para minimizar (3.5). O Algoritmo 4 gasta um pouco mais, 36 iterações e o Algoritmo 5 gasta 19 iterações, o que não é muito. Mas o Algoritmo 1 gasta 8394 iterações para resolver o problema, ou seja, 932 vezes mais iterações que o melhor algoritmo.

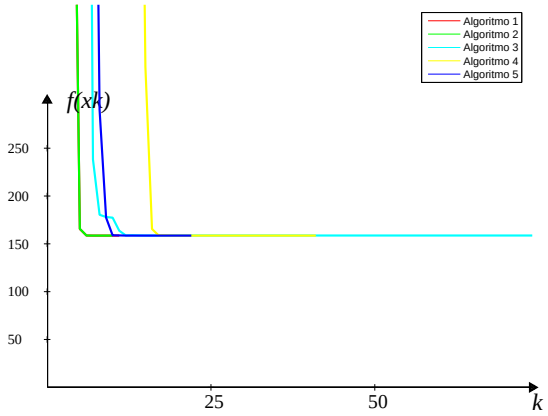

Figura 3.25: Caso em que $n = 500$

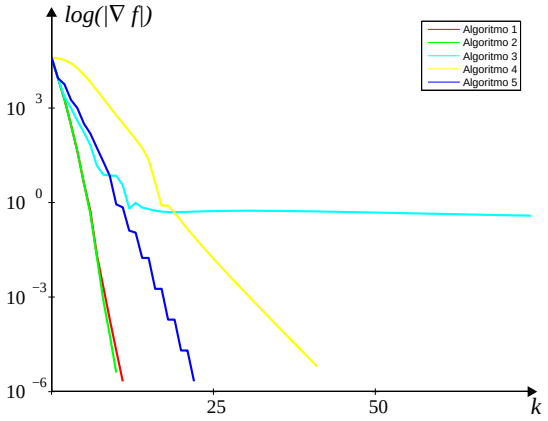

Figura 3.26: Caso em que $n = 500$

Se aumentarmos a dimensão para 500, o desempenho dos Algoritmos 1, 2, 4 e 5 não mudam muito em relação à dimensão 5, com 11, 10, 41 e 22 iterações respectivamente. O Algoritmo 3 gasta 26853 iterações para minimizar (3.5). As Figuras 3.25 e 3.26 mostram as 75 primeiras iterações.

Exemplo 6. Dado $b \in \mathbb{R}^n$, considere a função objetivo $f : \mathbb{R}^n \to \mathbb{R}$ sugerida em [Nes04, pág 56] e dada por:

$$f(x) = \sum_{i=1}^{n} e^{x_i + \langle b, x \rangle}. \tag{3.6}$$

Em nossos testes numéricos, consideramos $n = 10$ e $n = 60$ e o vetor b foi gerado aleatoriamente com componentes entre 0 e 1. Pelo fato de não conhecermos μ e L, fornecemos aos algoritmos, $\mu = 0$ e $L = 100000$.

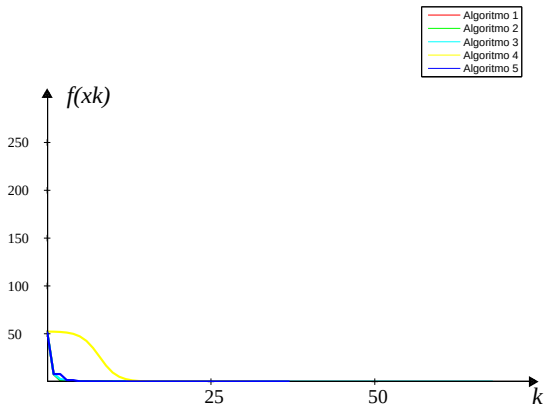

Figura 3.27: Caso em que $n = 10$

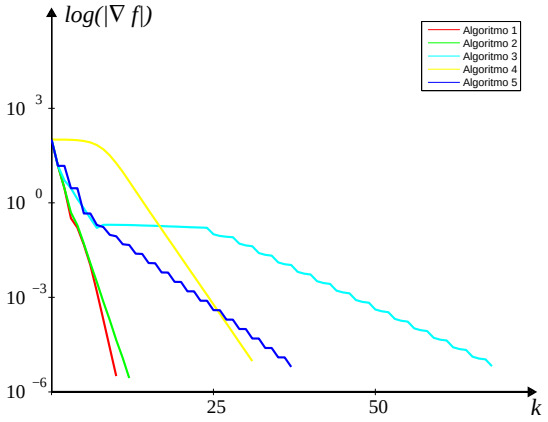

Figura 3.28: Caso em que $n = 10$

As Figuras 3.27 e 3.28 mostram o desempenho dos algorimos para a dimensão 10. Nesse caso há um empate técnico entre o Algoritmo 1 e o Algoritmo 2. O primeiro resolve em 10 iterações e o segundo em 12. O Algoritmo 4 fica em terceiro com 31 iterações, o Algoritmo 3 gasta 68 iterações para resolver o problema, enquanto que o Algoritmo 5 resolve em 37 iterações.

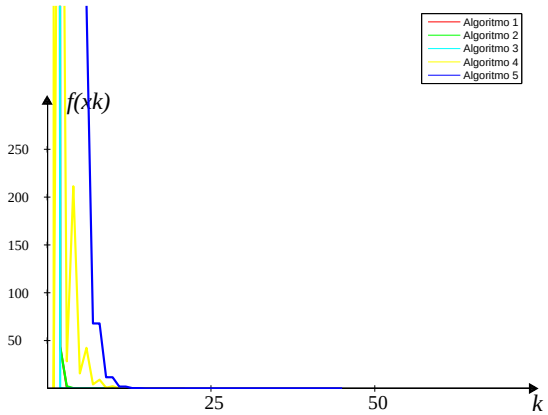

Figura 3.29: Caso em que $n = 60$

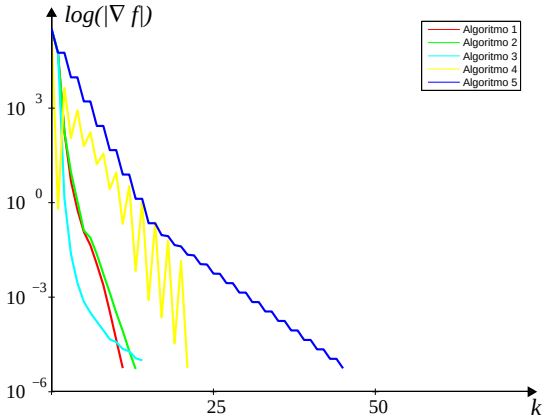

Figura 3.30: Caso em que $n = 60$

Já para a dimensão 60, o desempenho dos Algoritmos 1 e 2 foi semelhante que no caso de dimensão 10, gastando 11 e 13 iterações respectivamente. Mas os Algoritmos 3 e 4 apresentam um desempenho melhor que no caso de dimensão 10. O Algoritmo 3 gasta 14 iterações, enquanto que o Algoritmo 4 gasta 21 iterações. O Algoritmo 5 gasta 45 iterações para resolver o problema.

Aqui ocorreu um detalhe interessante, o computador criou um problema na dimensão 60 em que o Algoritmo 3 ganhou dos demais. Ele resolveu em 10 iterações, o Algoritmo 1 gastou 11, o Algoritmo 2 gastou 15 e o Algoritmo 4 gastou 27 iterações para resolver o problema. O Algoritmo 5 teve o pior desempenho com 45 iterações. Isso é mostrado nas Figuras 3.31 e 3.32.

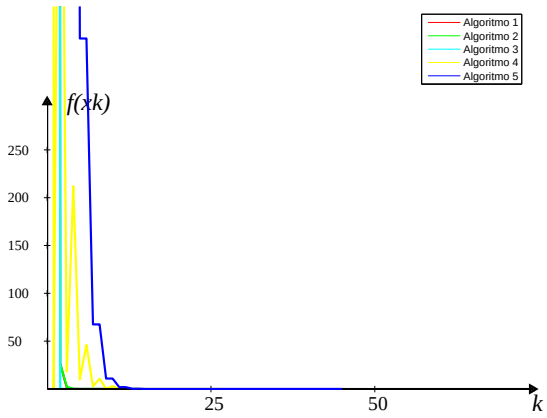

Figura 3.31: Caso em que $n = 60$

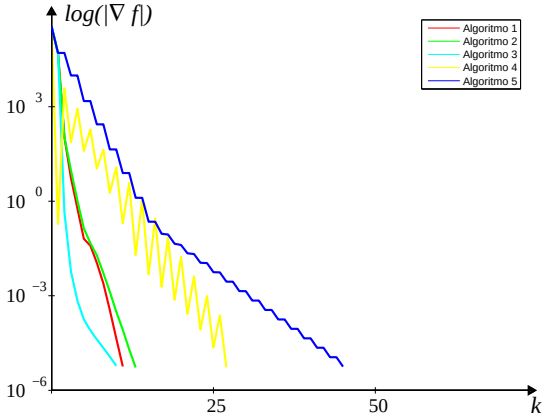

Figura 3.32: Caso em que $n = 60$

Para dimensões maiores que 60, já não é possível simular problemas com a função objetivo do tipo (3.6), pois os valores da função cresce muito, o que torna a sua solução muito difícil do ponto de vista computacional.

Exemplo 7. Considere agora a função objetivo $f : \mathbb{R}^n \to \mathbb{R}$ dada por

$$f(x) = \frac{1}{2}\langle Qx, x\rangle + \sum_{i=1}^{n} e^{x_i} - 1, \tag{3.7}$$

onde Q é definida positiva gerada aleatoriamente, a partir da dimensão dada.

Como não conhecemos o valor de μ e L, fornecemos aos algoritmos $\mu = 0$ e $L = 100000$. Comparamos os algoritmos para um problema de dimensão 10 e posteriormente um de dimensão 100.

As Figuras 3.33 e 3.34 mostram o comportamento dos algoritmos para o caso de dimensão 10. Os Algoritmos 5, 1 e 2 tem um comportamento similar com 49, 62 e 50 iterações respectivamente. O Algoritmo 3 resolve o problema em 380 iterações enquanto que o Algoritmo 4 resolve em 508 iterações. Apesar de μ e L ser desconhecido, o Algoritmo 3 teve um desempenho melhor que o Algoritmo 4.

Para o problema de dimensão 100, o Algoritmo 5 continua sendo o melhor, resolve em 81 iterações. O Algoritmo 2 gasta 129 iterações e o Algoritmo 1 gasta 200 iterações para resolver o problema. O desempenho dos Algoritmos 3 e 4 são inferiores com 572 e 802 iterações respectivamente. As Figuras 3.35 e 3.36 mostram as 75 primeiras iterações.

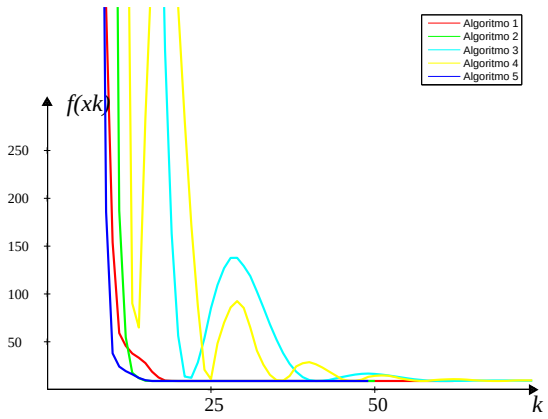

Figura 3.33: Caso em que $n = 10$

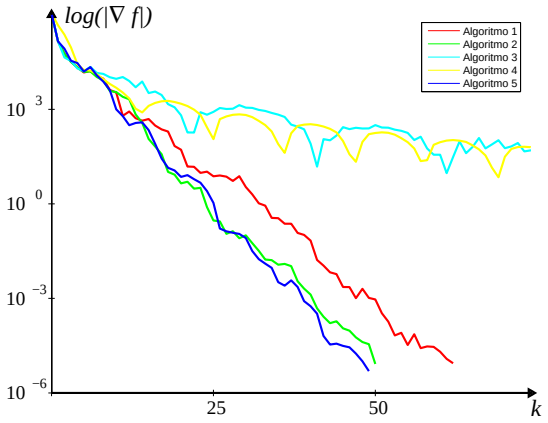

Figura 3.34: Caso em que $n = 10$

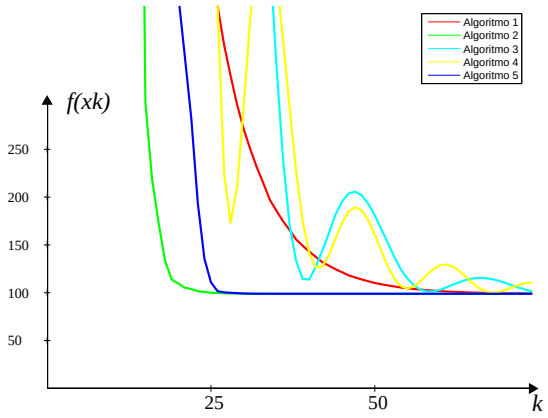

Figura 3.35: Caso em que $n = 100$

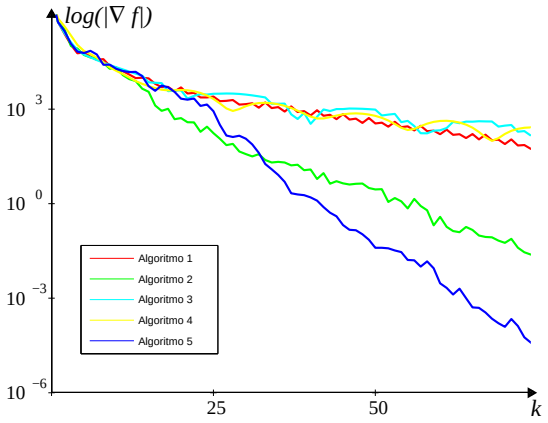

Figura 3.36: Caso em que $n = 100$

Perfil de desempenho

Comparamos diversos problemas de diversas dimensões arbitrárias. Rodamos os algoritmos para o caso de μ e L conhecido e depois desconhecidos. Rodamos 2000 problemas para as funções convexas com a dimensão aleatória variando entre 1 e 96619 para o caso de μ e L conhecidos, isso é mostrado na Figura 3.37. Para o caso de μ e L desconhecidos, rodamos 4000 problemas com dimensão aleatória variando entre 1 e 95654. A Figura 3.38 mostra o resultado.

Quando μ e L são conhecidos, o Algoritmo 1 e 2 são os mesmos e portanto é exibido somente um deles na Figura 3.37. Olhando a Figura 3.37, notamos que o Algoritmo 2 perde em poucos problemas para o Algoritmo 5. O Algoritmo 3 resolve 100% dos problemas gastando não mais que 2 vezes o número de iterações do melhor algoritmo. Note que no Algoritmo 1 e 2, há duas buscas lineares por iteração enquanto que o Algoritmo 3 tem somente uma. Isso significa que o Algoritmo 3 tem um desempenho muito bom na prática se μ e L são conhecidos. Isso varia muito de problema para problema. Para resolver 10% dos problemas, o Algoritmo 4 pode gastar até 5,5 vezes o número de iterações do melhor algoritmo.

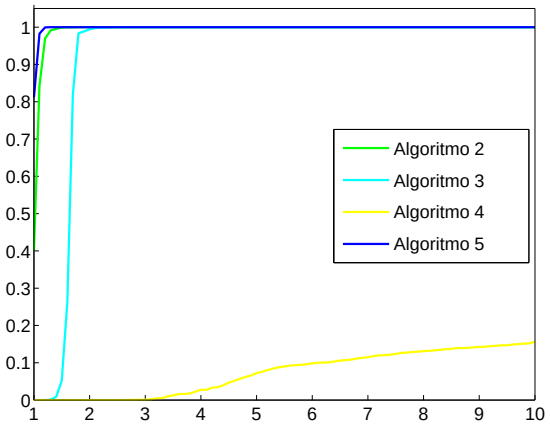

Figura 3.37: μ e L conhecidos

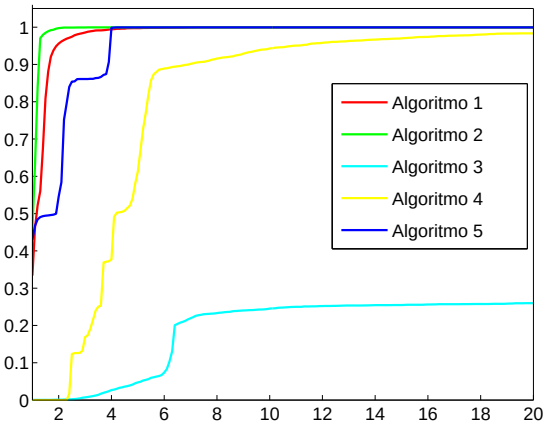

Figura 3.38: μ e L desconhecidos

Já para o caso de μ e L desconhecido, o Algoritmo 2 ganha em 100% dos problemas como mostra a Figura 3.38. O Algoritmo 1 perde em poucos problemas para o Algoritmo 2. Para resolver 100% dos problemas o Algoritmo 5 pode gastar não mais que 4 vezes o número de iterações do melhor algoritmo. O Algoritmo 4 que teve um desempenho bastante fraco caso μ e L eram conhecidos, passou a resolver 90% dos problemas usando não mais que 7 vezes o número de iterações gasto pelo melhor algoritmo. Por outro lado, o Algoritmo 3 pode gastar até 7 vezes o número de iterações do melhor algoritmo para resolver apenas 22% dos problemas. Para resolver 100% dos problemas, ele pode gastar até 700 vezes o número de iterações do melhor algoritmo. Isso ocorre pelo fato do Algoritmo 3 ser dependente das constante μ e L.

Conclusão

Essa dissertação foi dedicada ao estudo de complexidade de métodos de primeira ordem para minimização de funções convexas com constante de Lipschitz L para o gradiente e parâmetro de convexidade $\mu \geq 0$.

O objetivo principal foi apresentar um algoritmo (Algoritmo 1) e provar sua complexidade ótima. Como as constantes μ e L nem sempre são disponíveis, apresentamos o Algoritmo 2 que é uma modificação do anterior para não depender do conhecimento dessas constantes. Os algoritmos apresentados são variantes dos algoritmos discutidos em [GK08] que por sua vez fazem um ajuste fino no método ótimo proposto por Nesterov [Nes04] (Algoritmo 3). Como esse algoritmo depende do conhecimento da constante L, Nesterov propôs em [Nes07] um algoritmo (Algoritmo 4) que faz uma estimativa dessa constante.

Comparamos o desempenho desses métodos em termos de número de iterações para minimização de algumas funções convexas.

No caso em que as constantes μ e L são conhecidas, os Algoritmos 1 e 2 perdem em poucos problemas para o Algoritmo 5. O Algoritmo 4 teve o pior desempenho, pois ele não faz uso do conhecimento do parâmetro de convexidade μ.

Testamos também funções em que de fato não conhecemos os parâmetros μ e L, tais como, funções exponenciais. Nesses casos o Algoritmo 2 teve o melhor desempenho, seguido de perto pelo Algoritmo 1. O Algoritmo 5, que embora não faça uso dessas constantes, ficou em terceiro lugar. No entanto, é o Algoritmo 3 que tem o pior desempenho pois ele é altamente dependente dessas constantes.

Nos testes de minimização de funções quadráticas, o Algoritmo 5 de gradiente conjugados ganha em 100% dos problemas. Como provado em [Ber99, Pol87] esse algoritmo também é ótimo para minimização de funções quadráticas e seu desempenho é excelente como vemos nos testes númericos.

Finalizamos dizendo que o conceito de método ótimo está relacionado ao estudo de pior caso. Um algoritmo ser ótimo não significa que na prática ele tenha sempre um desempenho melhor que algoritmos que não sejam ótimos. Significa que a sua complexidade é proporcional à complexidade da classe de problemas que estamos resolvendo.

Referências Bibliográficas

[Ber99] D. P. Bertsekas, *Nonlinear programming*, Athena Scientific, Belmont, USA, 1999.

[DM02] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming **91** (2002), 201–213.

[GK08] C. C. Gonzaga and E. W. Karas, *Optimal steepest descent algorithms for unconstrained convex problems: Fine tuning Nesterov's method*, Tech. report, Dep. Matematica, Universidade Federal do Paraná, Brasil, 2008.

[HUL93a] J-B. Hiriart-Urruty and C. Lemaréchal, *Convex analysis and minimization algorithms i*, Springer-Verlag, New York, 1993.

[HUL93b] _____, *Convex analysis and minimization algorithms ii*, Springer-Verlag, New York, 1993.

[IS07] A. Izmailov and M. Solodov, *Otimização - métodos computacionais*, vol. 2, IMPA, Rio de Janeiro, 2007.

[Nes04] Y. Nesterov, *Introductory lectures on convex optimization. a basic course*, Kluwer Academic Publishers, Boston, 2004.

[Nes07] _____, *Gradient methods for minimizing composite objective function*, Discussion paper 76, CORE, UCL, Belgium, 2007.

[NW99] J. Nocedal and S. J. Wright, *Numerical optimization*, Springer Series in Operations Research, Springer-Verlag, 1999.

[Pol87] B. T. Poljak, *Introduction optimization*, Optimization Software Inc., New York, 1987.

[PR69] E. Polak and G. Ribière, *Note sur la convergence de méthodes de directions conjuguées*, Revue Française d'Informatique et de Recherche Opérationnelle **16** (1969), 35 – 43.

[Roc70] R. T. Rockafellar, *Convex analysis*, Princeton University Press, New Jersey, 1970.

[Weg05] I. Wegener, *Complexity theory*, Springer-Verlag, Berlin, 2005.