UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS FÍSICAS E MATEMÁTICAS
CURSO DE GRADUAÇÃO DE MATEMÁTICA

Luiz Augusto Turco Francisco

Métodos de Primeira Ordem para Otimização de Portfólio de Investimentos

Florianópolis
2021

Luiz Augusto Turco Francisco

Métodos de Primeira Ordem para Otimização de Portfólio de Investimentos

Trabalho de Conclusão de Curso submetido ao Curso de Graduação de Matemática da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Matemática.
Orientador: Prof. Douglas S. Gonçalves, Dr.

Florianópolis
2021

Dedicado a minha família

AGRADECIMENTOS

Primeiramente agradeço à toda minha família que me deram todo o apoio para concluir a graduação. Aos meus amigos Edson, Paulo e Danielli os quais sempre acreditaram em mim e, mesmo passando anos longe, lembraram de mim e passamos bons momentos nas minhas visitas à Joinville.

A todos os colegas do Curso de Matemática que conheci, em especial o Rafael, a Deborah e a Andresa que me acolheram e sempre me trataram bem.

Agradeço também o Tiago, colega da Matemática, mas que merece um parágrafo exclusivo, que me surpreendeu com sua amizade principalmente nos últimos meses. Graças a ajuda dele (varando a madrugada) este trabalho está completo, muito obrigado mesmo.

A Clara que fez parte de toda essa jornada acadêmica e é uma das pessoas mais especiais da minha vida. Agradeço principalmente pela companhia no início solitário do curso e pelo fato de ter me ajudado a crescer como pessoa.

Aos meus amigos Bruno, Eloy, Gabriela, Micca e Yan por terem me proporcionado momentos de alegria em momentos difíceis.

A Rhayssa por ter acreditado no meu potencial e acalmar meus ânimos nesta reta final estressante. Por todo amor e carinho que tem me dado e me ajudado a ver a vida de uma perspectiva mais leve.

Ao meu amigo Irandir por ser uma referência de pessoa resiliente, trabalhadora, focada e forte. Lembrar de sua linda história de superação fez eu enxergar meus problemas de maneira diferente.

Agradeço os professores Eliezer, Alda, Fernando, Bortolan, Gilles e Mariana. Ao professor Vinícius Albani por ter aceitado participar da banca de avaliação deste trabalho. Ao meu querido orientador, Douglas, por ter aceitado me orientar nesse último ano e por todo conhecimento passado. Por último, mas muito especial, a professora Silvia que foi como uma mãe no curso e sempre me ajudou nas dificuldades acadêmicas, e aceitou também fazer parte da banca.

RESUMO

O problema de otimização de portfólio de investimentos consiste em encontrar uma distribuição dos investimentos do portfólio a fim de minimizar o risco e maximizar o retorno esperado. Usando um modelo proposto por Markowitz, isto pode ser formulado como um problema de otimização quadrática e convexa. Para resolver este problema dois métodos foram estudados: o método de Frank-Wolfe e o método do Gradiente Projetado. Experimentos numéricos com problemas artificiais foram feitos para diferentes distribuições de autovalores para comparar ambos os métodos. O método que mais se destacou nos testes foi utilizado para a resolução de um problema de otimização de um portfólio de investimentos com dados reais. Com a devida tolerância, o método foi eficiente para encontrar o portfólio ótimo para diferentes níveis de aversão a risco.

Palavras-chave: Problema de otimização de portfólio. Frank-Wolfe. Gradiente Projetado. Otimização.

ABSTRACT

The portfolio optimization problem consists in finding a distribution of available funds over a finite number of investments in order to minimize risk and maximize expected return. Using the model proposed by Markowitz, this problem can be formulated as a convex quadratic optimization problem. Two methods were studied: the Frank-Wolfe method and the Projected Gradient method. Numerical experiments on artificial instances, with different eigenvalue distributions, were performed aiming to evaluate the behaviour of the methods under different scenarios. The best method in these tests was used to solve the portfolio optimization problem with real data. The method efficiently found the optimal portfolio for different risk aversion levels.

Keywords: Portfolio optimization problem. Frank-Wolfe. Projected Gradient. Optimization.

SUMÁRIO

1 INTRODUÇÃO 7
2 CONCEITOS PRELIMINARES 8
2.1 MINIMIZADORES 8
2.2 CONDIÇÕES DE OTIMALIDADE BASEADAS EM DIREÇÕES FACTÍVEIS 9
2.3 CONJUNTOS E FUNÇÕES CONVEXAS 12
3 ALGORITMOS DE DESCIDA 21
3.1 MÉTODOS DE DIREÇÕES DE DESCIDA 21
3.2 SELEÇÃO DO TAMANHO DE PASSO 22
3.2.1 Regra de minimização limitada (Busca linear exata) 22
3.2.2 Redução de tamanho de passo sucessivo e condição de Armijo (Busca linear inexata) 23
3.3 CONVERGÊNCIA GLOBAL 26
4 MÉTODOS DE FRANK-WOLFE E DO GRADIENTE PROJETADO . 28
4.1 MÉTODO DE FRANK-WOLFE 29
4.1.1 Subproblema de Frank-Wolfe no Simplex unitário 31
4.2 GRADIENTE PROJETADO 32
4.2.1 Subproblema do Gradiente Projetado no Simplex unitário 34
4.3 EXPERIMENTOS NUMÉRICOS EM PROBLEMAS ARTIFICIAIS . . 36
4.3.1 Testes utilizando busca linear inexata 37
4.3.2 Testes utilizando busca linear exata 39
4.3.3 Conclusão preliminar 41
5 OTIMIZAÇÃO DE PORTFÓLIO DE INVESTIMENTOS 42
5.1 ESTIMATIVAS PARA μ, σ E ρ 43
5.1.1 Taxa de retorno de um ativo 43
5.1.2 Risco de um ativo 43
5.1.3 Medindo a relação entre ativos 44
5.2 UM ESTUDO DE CASO 45
5.2.1 Fronteira Eficiente 47
6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS 49
REFERÊNCIAS 50
APÊNDICE A – OBTENÇÃO E PRÉ-PROCESSAMENTO DE DADOS REAIS 51

1 INTRODUÇÃO

O Problema de Otimização de Portfólio de Investimentos (POP) consiste em determinar a porcentagem de um capital a ser investida em cada um de n investimentos disponíveis de modo a maximizar o retorno esperado e minimizar o risco.

Segundo o modelo proposto em (MARKOWITZ, 1952), tal problema pode ser formulado como o problema de otimização

$$\begin{aligned} \text{minimizar} \quad & \frac{\kappa}{2} x^T G x - \mu^T x \\ \text{sujeito a} \quad & x \in \Delta_n, \end{aligned} \tag{1}$$

em que $\Delta_n = \{x \in \mathbb{R}^n \mid \sum_{i=1}^{n} x_i = 1, x \geq 0\}$ é conhecido como *Simplex Unitário*, G é uma matriz $n \times n$ e mu vetor coluna de dimensão n , o termo $x^T G x$ modela o risco do portfólio, $\mu^T x$ o retorno esperado, e $\kappa > 0$ controla o peso relativo entre estes objetivos.

Para a resolução de (1) utilizaremos métodos de primeira ordem para minimizar uma função convexa sobre um conjunto convexo e compacto, a saber, os métodos de Frank-Wolfe (FW) e Gradiente Projetado (GP) (BERTSEKAS, 1999).

Além da teoria matemática associada a otimização contínua, o interesse pelo tema do trabalho se explica pela importância dada pelo autor a investimentos. O POP é muito importante para qualquer investidor já que todos estes querem maximizar seus retornos admitindo o mínimo de risco possível. Um importantíssimo trabalho feito a respeito do POP é o do economista Harry Max Markowitz, (MARKOWITZ, 1952), o qual lhe proporcionou o prêmio Nobel de Economia em 1990.

O presente trabalho está organizado da seguinte maneira: no Capítulo 2 trata dos conceitos preliminares de otimização contínua e programação convexa. O Capítulo 3 discute os chamados Algoritmos de Descida (tanto FW quanto GP são métodos desta classe), e o importante teorema de convergência global. No Capítulo 4 apresentamos os métodos FW e GP, descrevemos como resolver seus subproblemas no simplex unitário e reportamos experimentos numéricos em problemas artificiais, inspirados em (1), a fim de analisar seu desempenho em diferentes cenários. O Capítulo 5 trata do POP e apresenta um estudo de caso com dados de investimentos reais. No Capítulo 6 são feitas as considerações finais e direções para trabalhos futuros. Por fim, no Apêndice A é feita a explicação detalhada de como foram obtidos e tratados os dados utilizados no Capítulo 5.

2 CONCEITOS PRELIMINARES

Neste capítulo iremos revisar alguns conceitos fundamentais em otimização contínua. O conteúdo apresentado é baseado na referência (MARTÍNEZ; SANTOS, 1995).

2.1 MINIMIZADORES

Em otimização contínua o objetivo é resolver o seguinte problema

$$\begin{aligned} \underset{x}{\text{minimizar}} \quad & f(x) \\ \text{sujeito a} \quad & x \in \Omega \subset \mathbb{R}^n, \end{aligned} \tag{2}$$

em que $f : \mathbb{R}^n \to \mathbb{R}$ é uma função contínua, chamada de *função objetivo* e Ω é um subconjunto de $\mathbb{R}^n$ chamado de *conjunto viável*. Dos elementos $x \in \Omega$, queremos encontrar $x^* \in \Omega$ tal que

$$f(x^*) \leq f(x), \quad \forall x \in \Omega. \tag{3}$$

Caso exista, x^* é chamado de *minimizador global* do problema (2) e $f(x^*)$ de *mínimo global* de f em Ω. Porém, muitas vezes pode ser muito difícil encontrarmos o minimizador global então nos contentamos em encontrar um *minimizador local* x^* de forma que valha (3) restrito a $\Omega \cap B(x^*, \epsilon)$, para algum $\epsilon > 0$.

É importante notar que nem sempre o problema (2) tem solução. Por exemplo, considere $f(x) = x$ e $\Omega = \mathbb{R}$. Perceba que, dado $x \in \mathbb{R}$, sempre existe y tal que $f(y) < f(x)$: definindo $y = x - 1$, por exemplo, obtemos $f(y) = f(x-1) = x - 1 < x = f(x)$, qualquer que seja x. Neste exemplo, a função objetivo não é limitada inferiormente.

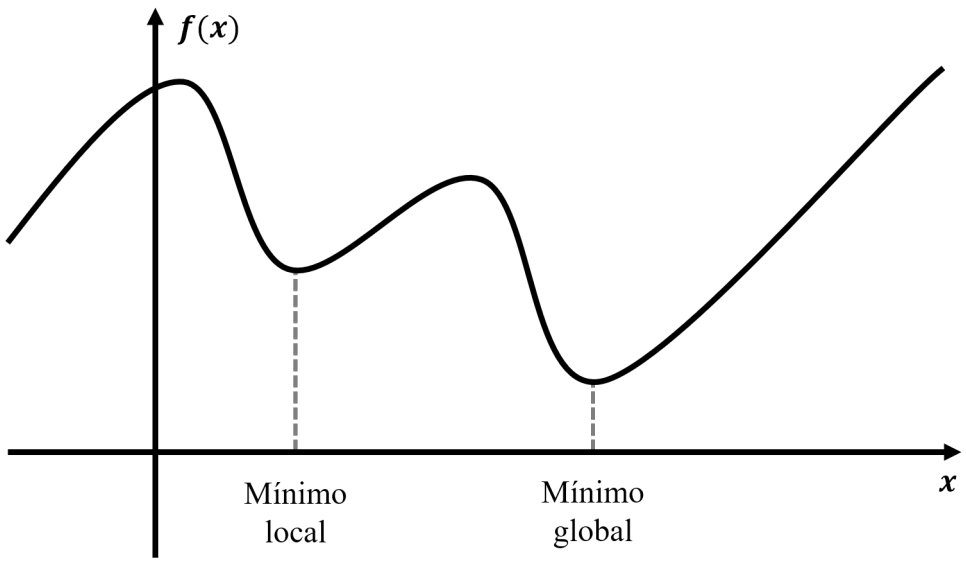

Figura 1 – Ilustração de mínimo local e global em uma dimensão.

Mas mesmo f sendo limitada inferiormente não garante a existência de minimizador: considere por exemplo $f(x) = e^{-x}$ em $\Omega = \mathbb{R}_+$. Embora o ínfimo seja zero, este valor não é atingido para nenhum $x \in \Omega$.

Isto nos leva a pergunta: que condições garantem a existência de solução do problema (2)? Uma resposta é fornecida pelo seguinte teorema cuja a demonstração pode ser encontrada em (MARTÍNEZ; SANTOS, 1995, Teorema 1.3.1).

Teorema 2.1.1. ***Bolzano-Weierstrass***
Se Ω é compacto e $f : \Omega \to \mathbb{R}$ é contínua, então existe $x^* \in \Omega$ minimizador global do problema (2)*.*

Este teorema será muito útil para nós, pois no problema de otimização de portfólio o conjunto viável será compacto e a função objetivo contínua (e diferenciável). Assim, temos garantias teóricas que existe, de fato, um portfólio ótimo.

Porém, ainda parece muito difícil resolver o problema (2). A primeira tentativa intuitiva talvez fosse avaliar a função em todos os pontos de Ω, mas isso é inviável, na maioria das vezes impossível. Então, torna-se necessário um estudo sobre em que condições podemos afirmar que um ponto é minimizador, ou pelo menos ter um candidato a minimizador. Este estudo será feito na próxima seção que trata de *condições de otimalidade*. Estas servirão não apenas para determinar propriedades necessárias a um minimizador, mas também de inspiração para métodos iterativos que discutiremos nos próximos capítulos.

2.2 CONDIÇÕES DE OTIMALIDADE BASEADAS EM DIREÇÕES FACTÍVEIS

A seguir introduziremos novos conceitos a fim de estudar as propriedades que pontos minimizadores devem satisfazer necessariamente.

Definição 2.2.1. Dizemos que $d \in \mathbb{R}^n$ é uma direção factível a partir de $x \in \Omega$, se $\exists t^* > 0$ tal que

$$x + td \in \Omega, \forall t \in [0, t^*).$$

A seguir definimos o conceito de direções de descida: direções nas quais a função decresce (ao menos localmente).

Definição 2.2.2. Dizemos que $d \in \mathbb{R}^n$ é uma direção de descida para f a partir de x, se $\exists \epsilon > 0$ tal que

$$f(x + td) < f(x), \quad \forall t \in (0, \epsilon).$$

Para funções continuamente diferenciáveis (que denotaremos por $f \in C^1$) é possível determinar se uma direção é de descida analisando a derivada direcional.

Proposição 2.2.1. *Seja $f \in C^1$. Se $\nabla f(x)^T d < 0$, então d é direção de descida para f a partir de x.*

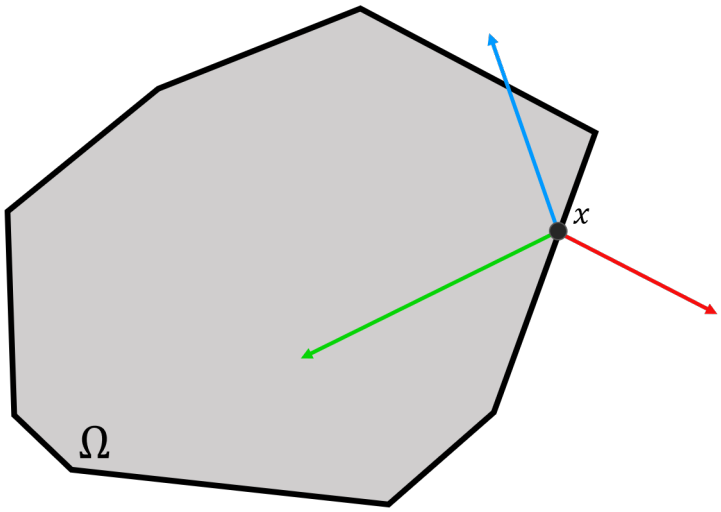

Figura 2 – A figura ilustra a região viável Ω. Veja que tanto o vetor azul quanto o verde são direções factíveis já que, para um tamanho de passo suficientemente pequeno, podemos "andar" nessas direções e continuaremos dentro do conjunto Ω. Já o vetor vermelho não é uma direção factível, já que não importa o quão pequeno seja o tamanho do passo, cairemos sempre fora do conjunto Ω.

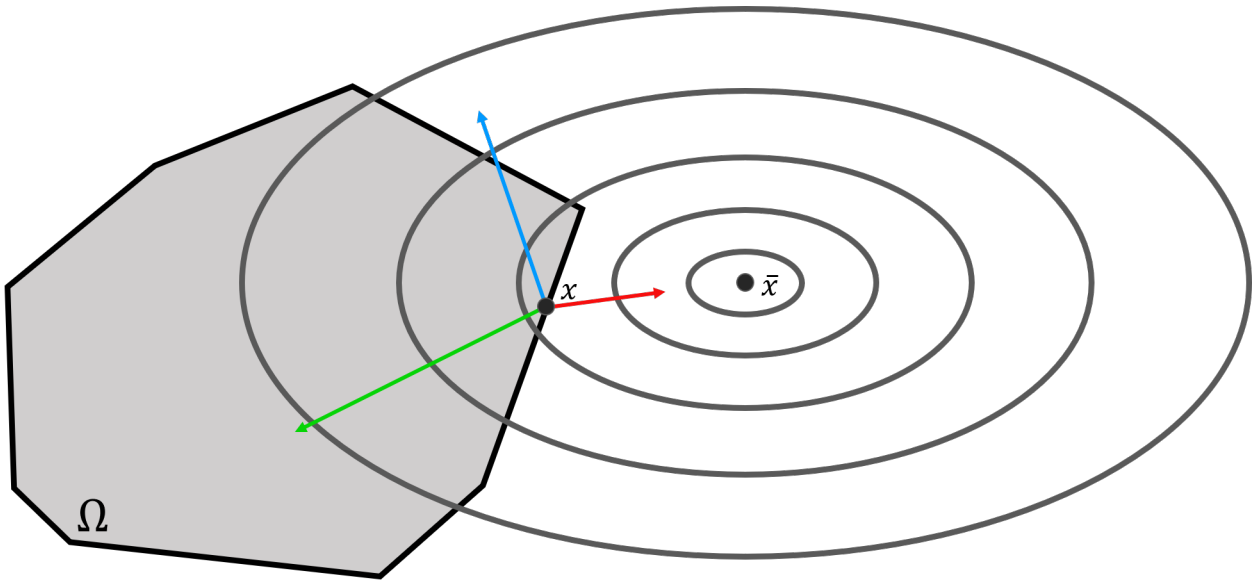

Figura 3 – Na figura são ilustradas as curvas de nível de uma função $f : \mathbb{R}^2 \to \mathbb{R}$ cujo o minimizador global em $\mathbb{R}^2$ é denotado por $\bar{x}$. Para as direções representadas pelos vetores verde e azul não importa o tamanho de passo $t > 0$ que "andemos", o valor funcional não diminui, muito pelo contrário. Já na direção do vetor vermelho conseguimos "andar" um tamanho de passo tal que o valor funcional diminui comparado ao valor em x. Portanto esta última é uma direção de descida para f a partir de x.

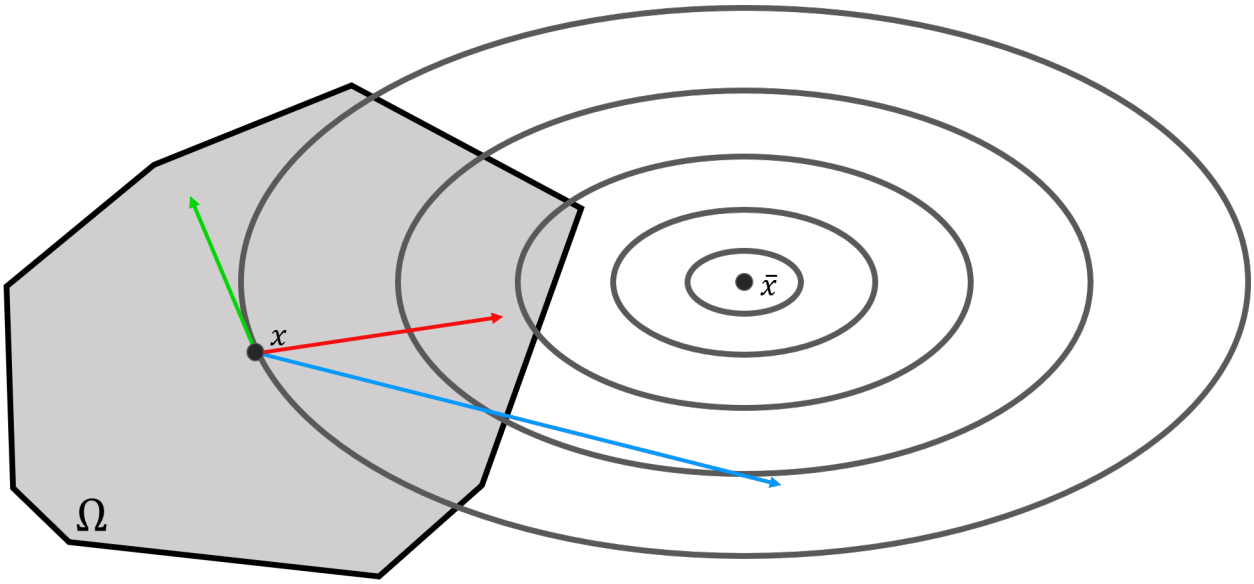

Figura 4 – Na figura são ilustradas as curvas de nível de uma função $f : \mathbb{R}^2 \to \mathbb{R}$ cujo o minimizador global em $\mathbb{R}^2$ é denotado por $\bar{x}$. Para a direção representada pelo vetor verde não importa o tamanho de passo $t > 0$ que "andemos", o valor funcional não diminui, pelo contrário, só aumenta. Já nos vetores vermelho e azul conseguimos "andar" um tamanho de passo tal que continuamos dentro do conjunto Ω e o valor funcional diminui comparado ao valor em x. Portanto estas últimas são direções factíveis e de descida para f a partir de x.

Demonstração. Sabemos que

$$\lim_{t\to 0} \frac{f(x+td)-f(x)}{t} = \nabla f(x)^T d < 0$$

Ou seja, existe $\bar{t} > 0$ tal que

$$\frac{f(x+td)-f(x)}{t} < 0, \quad \forall t \in (0, \bar{t})$$

Dessa forma, chegamos em $f(x+td) < f(x)\ \forall t \in (0, \bar{t})$ como queríamos demonstrar. □

Nos algoritmos que iremos estudar, buscaremos por direções que sejam tanto factíveis quanto de descida.

Então "bons" candidatos a minimizador do problema (2) são pontos do conjunto viável a partir dos quais não existe direção factível e de descida. Isto é formalizado no seguinte teorema no qual o conjunto de direções factíveis a partir de $x^* \in \Omega$ é denotado por $F(x^*)$.

Teorema 2.2.1. ***Condição necessária de primeira ordem.***
Se $x^* \in \Omega$ é um minimizador local de f em Ω, então não existe direção factível e de descida para f a partir de x^*. Além disso, se $f \in C^1$ em um aberto contendo Ω, então

$$\nabla f(x^*)^T d \geq 0, \forall d \in F(x^*).$$

Demonstração. Seja x^* um minimizador local de f em Ω ou seja, $f(x^*) \leq f(x) \quad \forall x \in B(x^*, \epsilon)$ para algum $\epsilon > 0$. Suponha que existe $d \in \mathbb{R}^n \backslash \{0\}$ direção factível e de descida. Então vale que

$$x^* + td \in \Omega, \quad \forall t \in [0, \widehat{t}) \tag{4}$$

e

$$f(x^* + td) < f(x), \quad \forall t \in (0, \widehat{\epsilon}) \tag{5}$$

Agora escolha $\bar{t} = min\{\widehat{t}, \widehat{\epsilon}, \frac{\epsilon}{2}\}$ então $\bar{x} = x^* + \bar{t}d \in \Omega$ além disso, $f(\bar{x}) = f(x^* + \bar{t}d) < f(x^*)$. Por outro lado, $\bar{x} \in B(x^*, \epsilon)$ e todos estes fatos contradizem a hipótese de que x^* é um minimizador local de f. Portanto, o resultado segue. □

Corolário 2.2.2. *Seja x^* minimizador local do problema* (2) *irrestrito. Então $\nabla f(x^*) = 0$.*

Demonstração. Como x^* é minimizador local do problema (2) irrestrito temos que

$$\mathbb{R}^n = F(x^*). \tag{6}$$

Dessa forma, para $d \in \mathbb{R}^n$ qualquer vale

$$\nabla f(x^*)^T d \geq 0, \tag{7}$$

mas ao mesmo tempo, pela equação (6), vale que

$$\nabla f(x^*)^T (-d) \geq 0,$$

ou seja,

$$\nabla f(x^*)^T d \leq 0. \tag{8}$$

Portanto, das desigualdades (7) e (8), $\nabla f(x^*) = 0$. □

2.3 CONJUNTOS E FUNÇÕES CONVEXAS

No problema de otimização de portfólio temos que para (2) a função f será convexa e Ω será conjunto convexo, ou seja, um problema de *otimização convexa* (ou um problema de programação convexa). Com estas propriedades adicionais tem-se novos resultados a cerca dos minimizadores do problema (2).

A priori estudemos a definição de conjunto convexo.

Definição 2.3.1. O conjunto $\Omega \subset \mathbb{R}^n$ é chamado um conjunto convexo se para quaisquer $x, y \in \Omega$ e para todo $\lambda \in [0, 1]$,

$$\lambda x + (1 - \lambda) y \in \Omega.$$

Com esta definição temos uma caracterização das direções factíveis a partir de um ponto de um conjunto convexo.

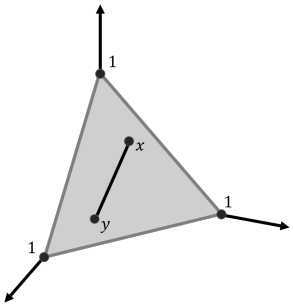

Figura 5 – (Simplex unitário de dimensão 3) Conjunto convexo

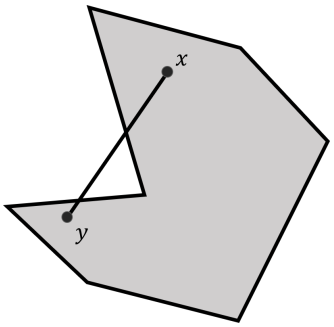

Figura 6 – Conjunto não convexo

Lema 2.3.1. *Seja Ω não vazio e convexo. Uma direção $d \in \mathbb{R}^n$ com $\|d\| = 1$ é direção factível a partir de $x^* \in \Omega$ se, e somente se,*

$$d = \frac{x - x^*}{\|x - x^*\|}$$

para algum $x \in \Omega \setminus \{x^*\}$.

Demonstração. ($\Rightarrow$) Seja d uma direção factível a partir de x^* com $\|d\| = 1$. Da Definição 2.2.1 temos que existe $t^* > 0$ tal que

$$x^* + td \in \Omega, \quad \forall t \in [0, t^*).$$

Defina $x = x^* + td$, e disso segue que $d = \frac{x-x^*}{t}$, para $t > 0$. Como $\|d\| = 1$, obtemos $\|d\| = \frac{\|x-x^*\|}{t} = 1$, ou seja, $t = \|x - x^*\|$. Portanto,

$$d = \frac{x - x^*}{\|x - x^*\|}.$$

($\Leftarrow$) Dado $x \in \Omega \setminus \{x^*\}$, tome $\epsilon > 0$ tal que $\epsilon < \|x - x^*\|$. Dessa forma, para todo $t \in (0, \epsilon)$ veja que

$$x^* + t\frac{(x - x^*)}{\|x - x^*\|} =$$
$$\frac{\|x - x^*\|x^* + t(x - x^*)}{\|x - x^*\|} =$$
$$\frac{(\|x - x^*\| - t)x^* + tx}{\|x - x^*\|} = x_t$$

Como,

$$\frac{\|x - x^*\| - t}{\|x - x^*\|} + \frac{t}{\|x - x^*\|} = 1.$$

e ambas as parcelas são não negativas, portanto, temos uma combinação convexa de x^* e x e, uma vez que Ω é convexo, temos que $x_t \in \Omega$ e $d = \frac{x-x^*}{\|x-x^*\|}$ é direção factível a partir de x^*. □

Com este lema pode-se formalizar, pelo teorema a seguir, uma condição necessária para que um ponto $x^* \in \Omega$ seja minimizador local de f em Ω convexo.

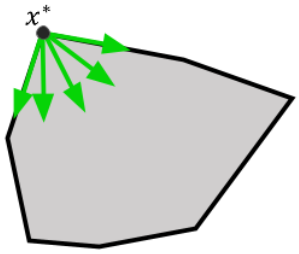

Figura 7 – Ilustração do Lema 2.3.1, todas as direções factíveis a partir de x^* podem ser escritas como $\frac{x-x^*}{\|x-x^*\|}$.

Teorema 2.3.1. *Seja $f \in C^1$ e Ω não vazio e convexo. Se $x^* \in \Omega$ é um minimizador local de f em Ω, então*

$$\nabla f(x^*)^T (x - x^*) \geq 0$$

para todo $x \in \Omega$.

Demonstração. Se $x = x^*$, o resultado é imediato. Suponha que $x \neq x^*$. Do Teorema 2.2.1 segue que

$$\nabla f(x^*)^T d \geq 0,$$

para qualquer direção viável d, a partir de x^*, e pelo Lema 2.3.1, obtemos

$$\nabla f(x^*)^T \frac{(x - x^*)}{\|x - x^*\|} \geq 0.$$

Portanto, segue que $\nabla f(x^*)^T (x - x^*) \geq 0, \quad \forall x \in \Omega$ □

De agora em diante será chamado de *ponto estacionário* o ponto que satisfizer a condição $\nabla f(x^*)^T (x - x^*) \geq 0, \quad \forall x \in \Omega$.

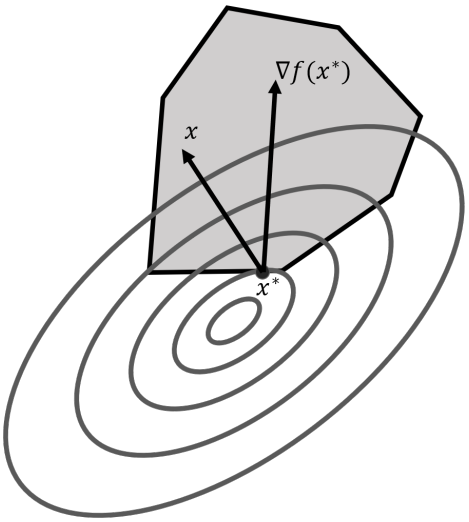

Figura 8 – Ilustração do Teorema 2.3.1, os vetores $x - x^*$ para todo $x \in \Omega$ faz um ângulo menor ou igual a 90° com o gradiente $\nabla f(x^*)$

Agora introduziremos o conceito de *função convexa*.

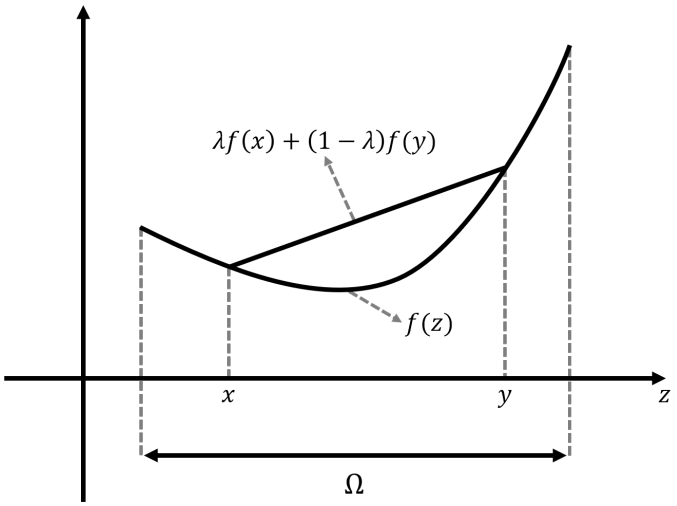

Figura 9 – Exemplo de função convexa. Note que o segmento de reta definido por $\lambda f(x) + (1-\lambda)f(y)$ está acima do gráfico da função, ou seja, admitem valores que superestima os valores funcionais de $f(\lambda x + (1-\lambda)y)$ para $\lambda \in (0,1)$.

Definição 2.3.2. Se Ω é um conjunto convexo, $f : \Omega \to \mathbb{R}$ é uma função convexa se, para todo $x, y \in \Omega, \lambda \in [0,1]$,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y).$$

Para funções continuamente diferenciáveis, a convexidade pode ser caracterizada pelo seguinte lema.

Lema 2.3.2. *Sejam $\Omega \subset \mathbb{R}^n$, Ω convexo, $f : D \to \mathbb{R}$ e $f \in C^1$ em um aberto D contendo Ω. Então a função f é convexa em Ω se e somente se para todo $x, y \in \Omega$*

$$f(y) \geq f(x) + \nabla f(x)^T (y-x).$$

Demonstração. ($\Rightarrow$) Seja f convexa, então para $x, y \in \Omega$ e $\lambda \in [0,1]$ temos $f(\lambda y + (1-\lambda)x) \leq \lambda f(y) + (1-\lambda)f(x)$. Logo,

$$f(\lambda y + (1-\lambda)x) - f(x) \leq \lambda(f(y) - f(x)).$$

Dessa forma,

$$\lim_{\lambda \to 0} \frac{f(\lambda y + (1-\lambda)x) - f(x)}{\lambda} \leq f(y) - f(x).$$

Então,

$$\nabla f(x)^T (y-x) \leq f(y) - f(x).$$

Assim, provamos que

$$f(y) \geq f(x) + \nabla f(x)^T (y-x), \quad \forall \quad x, y \in \Omega.$$

($\Leftarrow$) Se $f(y) \geq f(x) + \nabla f(x)^T (y-x), \quad \forall \quad x, y \in \Omega$, definindo $z_\lambda = \lambda y + (1-\lambda)x$, temos

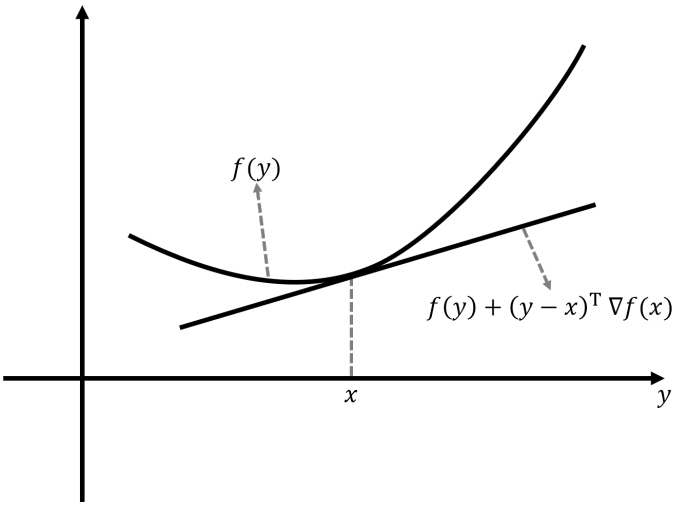

Figura 10 – Ilustração do Lema 2.3.2. Na figura vemos que a condição $f(z) \geq f(x) + (z-x)^T \nabla f(x)$ afirma que a aproximação linear, baseada na expansão de primeira ordem da série de Taylor, subestima a função convexa.

$$f(x) \geq f(z_\lambda) + \nabla f(z_\lambda)^T (x - z_\lambda)$$
$$f(y) \geq f(z_\lambda) + \nabla f(z_\lambda)^T (y - z_\lambda)$$

Desta maneira,

$$\begin{aligned}(1-\lambda)f(x) + \lambda f(y) &\geq (1-\lambda)(f(z_\lambda) + \nabla f(z_\lambda)^T (x - z_\lambda)) + \lambda(f(z_\lambda) + \nabla f(z_\lambda)^T (y - z_\lambda)) \\ &= f(z_\lambda) + \nabla f(z_\lambda)^T (x - z_\lambda - \lambda x + \lambda z_\lambda + \lambda y - \lambda z_\lambda) \\ &= f(z_\lambda) + \nabla f(z_\lambda)^T (\lambda y + (1-\lambda)x - z_\lambda) = f((1-\lambda)x + \lambda y).\end{aligned}$$

□

Com esta caracterização de função convexa, no teorema a seguir veremos que a condição do Teorema 2.3.1 é uma condição suficiente para problemas de otimização convexa.

Teorema 2.3.2. *Em um problema de programação convexa, temos que*

$$\nabla f(x^*)^T (x - x^*) \geq 0, \quad \forall x \in \Omega$$

é condição suficiente para que x^* seja minimizador global do problema (2)*.*

Demonstração. Do Lema 2.3.2 temos que

$$f(x) \geq f(x^*) + \nabla f(x^*)^T (x - x^*), \quad \forall x \in \Omega.$$

Se $\nabla f(x^*)^T (x - x^*) \geq 0$, obtemos $f(x) \geq f(x^*)$ para todo $x \in \Omega$, portanto x^* é o minimizador global do problema. □

Veremos no Capítulo 5 que o problema de otimização de portfólio de investimentos pode ser formulado como um problema de otimização convexa. Assim, o Teorema 2.3.2 nos dá que no problema de otimização de portfólio todos os minimizadores que encontrarmos corresponderão a portfólios ótimos (minimizadores globais).

Agora vejamos como a convexidade pode ser caracterizada para funções duas vezes continuamente diferenciáveis.

Teorema 2.3.3. *Seja $\Omega \subset \mathbb{R}^n$ aberto e convexo, $f : \Omega \to \mathbb{R}$ e $f \in C^2(\Omega)$. Então f é convexa se e somente se a matriz Hessiana $\nabla^2 f(x)$ é positiva semidefinida para todo $x \in \Omega$.*

Demonstração. ($\Rightarrow$)
Para todos $x, y \in \Omega$, pela expansão de Taylor com resto de Lagrange, temos que

$$f(y) = f(x) + (y-x)^T \nabla f(x) + \frac{1}{2}(y-x)^T \nabla^2 f(x + \lambda(y-x))(y-x)$$

para algum $\lambda \in [0,1]$. Dessa forma, como $\nabla^2 f$ é positiva semidefinida para todo ponto de Ω e $x + \lambda(y-x) \in \Omega$, já que Ω é convexo, obtemos

$$f(y) \geq f(x) + (y-x)^T \nabla f(x), \quad \forall x, y \in \Omega$$

que, pelo Lema 2.3.2 implica que f é convexa.
($\Leftarrow$)
Suponha que existe $x \in \Omega$ tal que $\nabla^2 f(x)$ não é positiva semidefinida, isto é, existe $d \in \mathbb{R}^n \setminus \{0\}$ tal que $d^T \nabla^2 f(x) d < 0$. Novamente por Taylor

$$f(x+d) = f(x) + d^T \nabla f(x) + \frac{1}{2} d^T \nabla^2 f(x + \lambda d) d,$$

para algum $\lambda \in [0,1]$. Usando a continuidade de $\nabla^2 f$, podemos escolher $\|d\|$ tão pequena quanto se queira de modo que $d^T \nabla^2 f(x + \lambda d) d < 0$, que implica em

$$f(x+d) < f(x) + d^T \nabla f(x),$$

o que contradiz, pelo Lema 2.3.2, a convexidade da função f. □

Definição 2.3.3. Se Ω é um conjunto convexo, $f : \Omega \to \mathbb{R}$ é uma função *estritamente* convexa se, para todos $x, y \in \Omega$, $\lambda \in (0,1)$,

$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda) f(y).$$

Com as devidas modificações[1], o Lema 2.3.2 e o Teorema 2.3.3 também valem para funções estritamente convexas. Além disso, com este novo conceito obtemos o seguinte teorema:

[1] Trocar as desigualdades dos enunciados por desigualdades estritas.

Teorema 2.3.4. *Em um problema de programação convexa o conjunto dos minimizadores de* (2) *é convexo. Se f é estritamente convexa, não pode haver mais de um minimizador.*

Demonstração. Chamemos de S o conjunto dos minimizadores globais do problema (2). Sejam $x, y \in S$. Então $f(x) = f(y) \leq f(\lambda x + (1-\lambda)y), \quad \lambda \in [0,1]$. Pela convexidade de f,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) = f(y) + \lambda(f(x) - f(y) = f(y).$$

Logo, $\lambda x + (1-\lambda)y \in S$ e portanto S é convexo. Suponha agora que existam $x, y \in S$, $x \neq y$ e f seja estritamente convexa. Para $\lambda \in [0,1]$, $f(\lambda x + (1-\lambda)y) \geq f(x) = f(y)$ pois x, y são minimizadores globais, mas $f(\lambda x + (1-\lambda)y) < f(x) = f(y)$ pelo fato de f ser estritamente convexa. Portanto, segue a contradição e o resultado segue. □

Um último resultado que veremos neste capítulo diz respeito a projeção sobre um conjunto convexo e fechado. Este resultado será bem importante ao discutirmos o método de Gradiente Projetado (Capítulo 4).

Teorema 2.3.5. ***Teorema da Projeção***
Sejam $\Omega \subset \mathbb{R}^n$, e considere $\|\cdot\|$ uma norma arbitrária então considere o seguinte problema

$$\begin{aligned} \underset{y}{\text{minimizar}} \quad & \|y - x\| \\ \text{sujeito a} \quad & y \in \Omega. \end{aligned} \tag{9}$$

a) Se Ω é fechado e $x \in \mathbb{R}^n$, o problema (9) *tem solução.*

b) Se Ω é convexo e $\|\cdot\|$ é a norma euclidiana, a solução do problema é única. Nesse caso, chamamos $P_\Omega(x)$ de projeção de x em Ω.

c) De agora em diante, Ω é convexo e fechado e a norma é a euclidiana. Para todo $y \in \Omega$,

$$(y - P_\Omega(x))^T (x - P_\Omega(x)) \leq 0. \tag{10}$$

d) Para todo $y \in \Omega$,

$$\|y - P_\Omega(x)\| \leq \|y - x\|. \tag{11}$$

e) Para quaisquer $x, z \in \mathbb{R}^n$,

$$\|P_\Omega(x) - P_\Omega(z)\| \leq \|x - z\|. \tag{12}$$

Demonstração. a) Seja $z \in \Omega$ qualquer. Defina a bola fechada $B[x, \|z-x\|]$. Veja que $\|y-x\| \leq \|z-x\|$ para algum $y \in \Omega$. Note que queremos resolver o seguinte problema

$$\begin{aligned} \underset{y}{\text{minimizar}} \quad & f(y) = \frac{1}{2}\|y-x\|^2 \\ \text{sujeito a} \quad & y \in \Omega \cap B[x, \|z-x\|] = A. \end{aligned} \tag{13}$$

Como A é compacto, e f é contínua, segue do Teorema 2.1.1 que o problema acima tem solução. Como este é equivalente ao problema (9), então o resultado segue.
b) Basta ver que

$$\nabla^2 f(y) = I, \quad \forall y \in \Omega.$$

em que I denota a matriz identidade $n \times n$. Tal matriz obviamente é positiva definida então pelo Teorema 2.3.3 segue que f é estritamente convexa. Portanto, o problema (13) tem solução única pelo Teorema 2.3.4.

c) Primeiro perceba que o gradiente da função f é dado por

$$\nabla f(y) = y - x.$$

Então $\nabla f(P_\Omega(x)) = P_\Omega(x) - x$. Como $P_\Omega(x)$ é solução do problema (13) do Teorema 2.3.1 segue que

$$(P_\Omega(x) - x)^T (y - P_\Omega(x)) \geq 0,$$

ou seja,

$$(y - P_\Omega(x))^T (x - P_\Omega(x)) \leq 0.$$

como desejado.
d) Primeiramente considere o seguinte produto interno

$$(y-x)^T(y-x) = (y - x + P_\Omega(x) - P_\Omega(x))^T (y - x + P_\Omega(x) - P_\Omega(x)),$$

e então obtemos

$$(y - P_\Omega(x))^T (y - P_\Omega(x)) + 2(y - P_\Omega(x))^T (P_\Omega(x) - x) + (P_\Omega(x) - x)^T (P_\Omega(x) - x).$$

Do item (c) temos que $(y - P_\Omega(x))^T (P_\Omega(x) - x) \geq 0$ então

$$(y-x)^T(y-x) \geq (y - P_\Omega(x))^T (y - P_\Omega(x))$$

Portanto, $\|y - P_\Omega(x)\| \leq \|y - x\|$.
e) Como o caso em que $P_\Omega(x) = P_\Omega(z)$ é imediato, suponha que $P_\Omega(x) \neq P_\Omega(z)$. Agora veja que do item a) vale

$$(P_\Omega(z) - P_\Omega(x))^T (x - P_\Omega(x)) \leq 0,$$

e

$$(z - P_\Omega(z))^T (P_\Omega(x) - P_\Omega(z)) \leq 0.$$

Somando estas desigualdades, obtemos

$$\begin{aligned}
&(P_\Omega(z) - P_\Omega(x))^T (x - P_\Omega(x)) + (z - P_\Omega(z))^T (P_\Omega(x) - P_\Omega(z)) \\
&= (P_\Omega(x) - P_\Omega(z))^T (P_\Omega(x) - x) + (z - P_\Omega(z))^T P_\Omega(x) - P_\Omega(z)) \\
&= (P_\Omega(x) - x + z - P_\Omega(z))^T (P_\Omega(x) - P_\Omega(z)) \\
&= \|P_\Omega(x) - P_\Omega(z)\|^2 + \langle z - x, P_\Omega(x) - P_\Omega(z) \\
&= \|P_\Omega(x) - P_\Omega(z)\|^2 - (x - z)^T (P_\Omega(x) - P_\Omega(z)) \leq 0
\end{aligned}$$

Logo,

$$\begin{aligned}
\|P_\Omega(x) - P_\Omega(z)\|^2 &\leq (x - z)^T P_\Omega(x) - P_\Omega(z)) \\
\|P_\Omega(x) - P_\Omega(z)\|^2 &\leq \|P_\Omega(x) - P_\Omega(z)\| \|x - z\|
\end{aligned}$$

Portanto, $\|P_\Omega(x) - P_\Omega(z)\| \leq \|x - z\|$ e disso segue a continuidade da função P_Ω.

□

3 ALGORITMOS DE DESCIDA

Neste capítulo, baseado na referência (BERTSEKAS, 1999), vamos considerar o problema (2) irrestrito, isto é $\Omega = \mathbb{R}^n$, com f contínua e diferenciável. Os métodos que iremos estudar fazem parte de uma classe de algoritmos ditos de "descida iterativa".

A ideia principal desses algoritmos é a seguinte: a partir de um ponto $x_0 \in \mathbb{R}^n$ (um "chute" inicial), sucessivamente geramos pontos $x_1, x_2, \ldots$ tais que o valor funcional diminua a cada iteração, ou seja, $f(x_{k+1}) < f(x_k)$, para $k = 0, 1, \ldots$. Desta maneira, sucessivamente melhoramos a estimativa de solução atual e esperamos que f decresça de forma a atingir o mínimo.

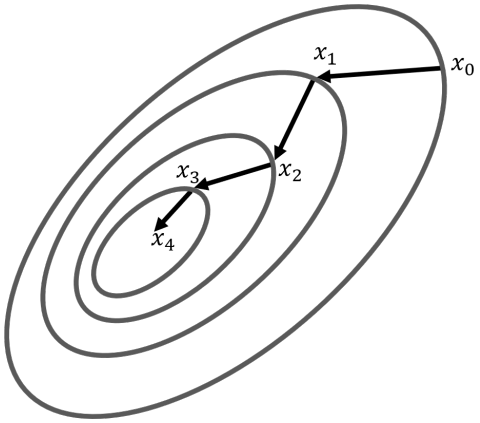

Figura 11 – Ilustração da ideia de um algoritmo de descida.

3.1 MÉTODOS DE DIREÇÕES DE DESCIDA

Seja $x \in \mathbb{R}^n$ tal que $\nabla f(x) \neq 0$. Da condição necessária de primeira ordem (Teorema 2.2.1) segue que existe $d \in \mathbb{R}^n$ tal que $\nabla f(x)^T d < 0$, ou seja, pela Proposição 2.2.1, d é uma *direção de descida*.

Considere o seguinte vetor

$$x_t = x + td, \quad t \geq 0.$$

Da expansão de Taylor de primeira ordem com resto infinitesimal, temos que

$$f(x_t) = f(x) + t\nabla f(x)^T d + o(t).$$

Para t próximo de zero, o termo $t\nabla f(x)^T d$ domina $o(t)$ e dessa forma, para t positivo e suficientemente pequeno, $f(x + td)$ é menor que $f(x)$.

Esta é a base dos algoritmos que iremos estudar. Se x_k é um iterado tal que $\nabla f(x_k) \neq 0$, calculamos uma direção de descida d_k, um tamanho de passo t_k que garanta que $f(x_k + t_k d_k) < f(x_k)$ e atualizamos

$$x_{k+1} = x_k + t_k d_k. \tag{14}$$

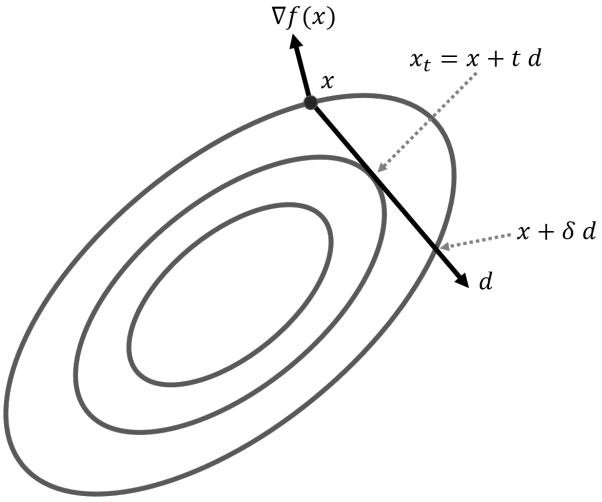

Figura 12 – Caso a direção d faça um ângulo maior que 90° com $\nabla f(x)$, ou seja,$\nabla f(x)^T d < 0$, existe um intervalo $(0, \delta)$ de tamanhos de passo de forma que $f(x + td) < f(x)$ para todo $t \in (0, \delta)$.

Se $\nabla f(x_k) = 0$, o método para, e retorna x_k como um *ponto estacionário*.

Como exemplo de direções de descida em otimização irrestrita temos a direção do *Método do Gradiente*: $d_k = -\nabla f(x_k)$, e a direção de *Métodos tipo Newton*: $d_k = -B_k^{-1}\nabla f(x_k)$, em que, para todo k, B_k é simétrica positiva definida e existem e $0 < c_1 < c_2$ tais que

$$c_1\|z\|^2 \leq z^T B_k z \leq c_2\|z\|^2, \quad \forall z \in \mathbb{R}^n.$$

3.2 SELEÇÃO DO TAMANHO DE PASSO

Escolhida uma direção para "andarmos" nos deparamos com outro problema: qual o tamanho do passo que devemos dar na direção escolhida?

Abaixo discutiremos duas regras para a escolha do tamanho de passo.

3.2.1 Regra de minimização limitada (Busca linear exata)

Dado $s > 0$, t_k é escolhido de forma a produzir a maior redução da função objetivo entre todos os tamanhos de passo no intervalo $[0, s]$, ou seja,

$$f(x_k + t_k d_k) = \min_{t \in [0,s]} f(x_k + td_k). \tag{15}$$

Nem sempre é fácil resolver o problema (15). Por exemplo, quando a função f não é convexa tal problema pode ser muito difícil. Existem algoritmos como interpolação quadrática e cúbica, e método da seção áurea (veja o apêndice C de (BERTSEKAS, 1999)) que permitem resolver (15) de forma aproximada. Felizmente, para o POP, o problema (15) é simples pois nossa função objetivo será quadrática e convexa.

3.2.2 Redução de tamanho de passo sucessivo e condição de Armijo (Busca linear inexata)

Uma regra mais simples que consiste em dado um tamanho de passo t escolhido previamente, se o vetor correspondente $x_k + td_k$ não produz uma redução no valor funcional, ou seja, $f(x_k + td_k) \geq f(x_k)$, o tamanho de passo é reduzido, às vezes repetidamente, por um certo fator pré-estabelecido, até que haja boa redução do valor funcional: $f(x_k + t_k d_k) < f(x_k)$. Esta redução é chamada *decréscimo simples*.

Apesar desta regra ser bastante prática, apenas o decréscimo simples e o fato de d_k ser de descida não garantem que a sequência $\{x_k\}$ gerada por um algoritmo de descida convirja a algum mínimo local ou global. Um exemplo é apresentado na figura a seguir.

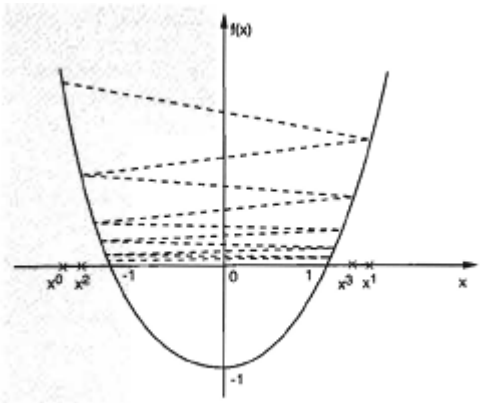

Figura 13 – (BERTSEKAS, 1999)

Exemplo 3.2.1. Na Figura 13 a função apresentada é

$$f(x) = \begin{cases} \frac{3(1-x)^2}{4} - 2(1-x) & \text{se} x > 1, \\ \frac{3(1+x)^2}{4} - 2(1+x) & \text{se} x > -1, \\ x^2 - 1, & \text{se} -1 \leq x \leq 1. \end{cases}$$

e o gradiente é dado por

$$\nabla f(x) = \begin{cases} \frac{3x}{2} + \frac{1}{2} & \text{se} x > 1, \\ \frac{3x}{2} - \frac{1}{2} & \text{se} x < -1, \\ 2x, & \text{se} -1 \leq x \leq 1. \end{cases}$$

Temos que a função f é estritamente convexa, continuamente diferenciável, e minimizada em $x^* = 0$. Além disso, para quaisquer dois escalares x e $\bar{x}$, nós temos

$$f(x) < f(\bar{x}) \quad \text{se e somente se} \quad |x| < |\bar{x}|.$$

Para $x > 1$, temos

$$x - \nabla f(x) = x - \frac{3x}{2} - \frac{1}{2} = -(1 + \frac{x-1}{2}),$$

de onde pode ser verificado que $|x-\nabla f(x)| < |x|$, então $f(x-\nabla f(x)) < f(x)$ e $x-\nabla f(x) < -1$. De maneira análoga, para $x < 1$ nós obtemos $f(x-\nabla f(x)) < f(x)$ e $x-\nabla f(x) < -1$. Considerando agora a iteração com a direção $-\nabla f(x_k)$ onde o tamanho de passo é sucessivamente reduzido a partir do tamanho de passo $t = 1$ até que a redução seja obtida. Comecemos de um ponto que satisfaça $|x_0| > 1$. Das equações anteriores, segue que $f(x_0 - \nabla f(x_0)) < f(x_0)$ e o tamanho de passo $t = 1$ é aceito pelo método. Então, o próximo ponto $x_1 = x_0 - \nabla f(x_0)$ satisfaz $|x_1| > 1$. Repetindo o argumento prévio, nós obtemos uma sequência $\{x_k\}$ que satisfaz $|x_k| > 1$ para todo k, e não converge para o único ponto estacionário $x^* = 0$. De fato, é possível mostrar que a sequência $\{x_k\}$ terá dois pontos limites, $\tilde{x}_1 = 1$ e $\tilde{x}_2 = -1$, para todo x_0 tal que $|x_0| > 1$.

Para evitar este tipo de contraexemplo podemos exigir mais na escolha do tamanho de passo de modo a garantir um *decréscimo suficiente*. Dados escalares t, β, e σ, tais que $0 < \beta < 1$, $0 < \sigma < 1$, definimos $t_k = \beta^{m_k} t$, em que m_k é o primeiro inteiro m não negativo tal que a *condição de Armijo*

$$f(x_k) \geq f(x_k + \beta^m t d_k) - \sigma\beta^m t \nabla f(x_k)^T d_k \tag{16}$$

é satisfeita.

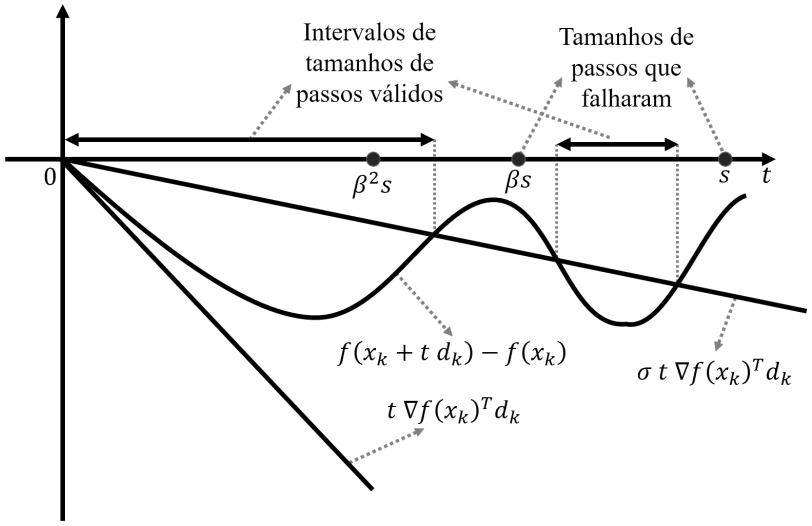

Figura 14 – Condição de Armijo

Usualmente escolhe-se σ próximo de zero, mais especificamente $\sigma \in [10^{-5}, 10^{-1}]$, e o fator de redução β geralmente entre $\frac{1}{2}$ e $\frac{1}{10}$.

Embora a condição de Armijo seja mais exigente que o decréscimo simples, ainda é possível construir contraexemplos nos quais os pontos limites da sequência $\{x_k\}$ gerada por algoritmos de descida não são estacionários. O problema é que as direções de descida d_k podem se tornar assintoticamente ortogonais ao gradiente.

Para evitar este outro problema precisaremos impor condições adicionais sobre a sequência de direções de busca $\{d_k\}$ gerada por um algoritmo de descida.

Definição 3.2.1. Sejam $\{x_k\}$, $\{d_k\}$ sequências geradas por um algoritmo de descida. A sequência de direções $\{d_k\}$ é dita *gradient related* se para qualquer subsequência $\{x_k\}_{k\in K}$ que converge a um ponto não-estacionário, a subsequência correspondente de direções $\{d_k\}_{k\in K}$ é limitada e satisfaz

$$\lim_{k\to\infty} \sup_{k\in K} \nabla f(x_k)^T d_k < 0.$$

Como exemplo de direções de busca para otimização irrestrita que são *gradient related* temos a direção do *Método do Gradiente* e a direção de *Métodos tipo Newton* como foram mencionadas na Seção 3.1.

Proposição 3.2.1. *As direções do Método do Gradiente e dos Métodos tipo Newton são gradient related.*

Demonstração. Seja $\{x_k\}_{k\in \mathrm{K}}$ convergindo a $\bar{x}$ não estacionário.

a) Direção do *Método do Gradiente*.
Esta direção é limitada, pois f é diferenciável e veja que

$$\lim_{k\to\infty} sup_{k\in \mathrm{K}} -\nabla f(x_k)^T \nabla f(x_k) = \lim_{k\to\infty} sup_{k\in \mathrm{K}} -\|\nabla f(x_k)\|^2 \leq -\|\nabla f(\bar{x})\|^2 < 0.$$

Portanto, a direção do Método do Gradiente é *gradient related*.

b) Direção do *Método tipo Newton*.
Esta direção é limitada pela sua própria definição e note que

$$\lim_{k\to\infty} sup_{k\in \mathrm{K}} -\nabla f(x_k)^T B_k^{-1} \nabla f(x_k) \leq -\frac{1}{c_2}\|\nabla f(\bar{x})\|^2 < 0.$$

Portanto, a direção do Método tipo Newton é *gradient related*. □

Com a condição de gradient related para as direções de descida e escolhendo o tamanho de passo de modo a cumprir a condição de Armijo, finalmente chegamos a um protótipo de algoritmo de descida *globalmente convergente*: independente do ponto inicial x_0, todo ponto limite da sequência $\{x_k\}$ gerada pelo algoritmo será um ponto estacionário (cumprirá a condição necessária de primeira ordem).

Algoritmo 1: Busca linear com direções gradient related.

Entrada: Dados $x_0 \in \Omega$, $\varepsilon > 0$, $\beta, \sigma \in (0,1)$, $M > 0$, faça $k = 0$.

Passo 1. Se $\|\nabla f(x_k)\| < \varepsilon$, pare.

Passo 2. Encontre uma direção d_k tal que $\|d_k\| \leq M$ e

$$\lim_{k\to\infty} \sup_{k\in K} \nabla f(x_k)^T d_k < 0$$

Passo 3. Faça t = 1. Enquanto

$$f(x_k + td_k) > f(x_k) + \sigma t \nabla f(x_k)^T d_k$$

faça $t \leftarrow \beta t$.
Passo 4. Faça $t_k = t$, $x_{k+1} = x_k + t_k d_k$, $k \leftarrow k + 1$ e volte ao Passo 1.

Como estamos considerando o problema (2) irrestrito, a condição necessária de primeira ordem se reduz ao Corolário 2.2.2: $\nabla f(x^*) = 0$. No Passo 1 verificamos se a aproximação x_k satisfaz essa condição de forma aproximada. No Passo 2 escolhemos uma direção que satisfaça a condição de *gradient related* (como a direção do *Método do gradiente* ou as direções dos *Métodos tipo Newton*) e no Passo 3 buscamos um tamanho de passo que satisfaça a condição de Armijo. Com essas condições satisfeitas temos garantias teóricas que todo ponto limite da sequência $\{x_k\}$ gerada pelo algoritmo será estacionário. Isto é tratado na próxima seção.

3.3 CONVERGÊNCIA GLOBAL

O teorema a seguir estabelece a convergência global do Algoritmo 1.

Teorema 3.3.1. *Seja $\{x_k\}$ uma sequência gerada pelo Algoritmo 1. Então todo ponto limite de $\{x_k\}$ é estacionário.*

Demonstração. Suponha por absurdo que uma subsequência $\{x_k\}_{k\in K}$convirja à $\bar{x}$ não estacionário. Da condição de Armijo segue que:

$$\frac{f(x_k) - f(x_k + t_k d_k)}{\sigma \nabla f(x_k)^T d_k} \geq t_k \geq 0 \tag{17}$$

Aplicando o limite em (17), do fato que d_k é *gradient related* temos que

$$0 \geq \lim_{k\in K} \frac{f(x_k) - f(x_k + t_k d_k)}{\sigma \nabla f(x_k)^T d_k} \geq 0,$$

ou seja,

$$\lim_{k\in K} \frac{f(x_k) - f(x_k + t_k d_k)}{\sigma \nabla f(x_k)^T d_k} = 0 \geq \lim_{k\in K} t_k \geq 0,$$

portanto,

$$\lim_{k\in K} t_k = 0.$$

Como $\{d_k\}$ é limitada, existe uma subsequência $\{d_k\}_{k\in\bar{K}}$, com $\bar{K} \subset K$, convergindo para $\bar{d}$. Além disso, do fato de $\{t_k\}_{k\in K} \to 0$ segue que para algum $\bar{t}_k$ a condição de Armijo falha , ou seja

$$f(x_k + \bar{t}_k d_k) > f(x_k) + \sigma \nabla f(x_k)^T \bar{t}_k d_k.$$

Usando Taylor de primeira ordem do lado esquerdo

$$f(x_k) + \bar{t}_k \nabla f(x_k)^T d_k + o(\bar{t}_k) > f(x_k) + \sigma \nabla f(x_k)^T \bar{t}_k d_k.$$

Logo

$$\nabla f(x_k)^T d_k + \frac{o(\bar{t}_k)}{\bar{t}_k} > \sigma \nabla f(x_k)^T d_k$$

$$\frac{o(\bar{t}_k)}{\bar{t}_k} > (\sigma - 1) \nabla f(x_k)^T d_k$$

e tomando limite na subsequência $\bar{K}$, temos

$$0 \geq (\sigma - 1) \nabla f(\bar{x})^T \bar{d}$$

implicando em

$$\nabla f(\bar{x})^T \bar{d} \geq 0.$$

já que $\sigma \in (0, 1)$. Isto contradiz o fato de $\{d_k\}$ ser *gradient related*. □

Mesmo o Teorema 3.3.1 tendo considerado um algoritmo de descida para o problema de otimização *irrestrito*, é possível adaptá-lo e aplicá-lo no contexto de otimização com restrições se considerarmos métodos que usem direções *gradient related e factíveis*. Este é o caso dos métodos de Frank-Wolfe e Gradiente Projetado que apresentaremos no próximo capítulo.

4 MÉTODOS DE FRANK-WOLFE E DO GRADIENTE PROJETADO

Estudamos no capítulo anterior métodos de otimização para problema (2) irrestrito ($\Omega = \mathbb{R}^n$). Agora abordaremos o mesmo problema com restrições. Como nosso problema de interesse, o POP, está restrito a um conjunto convexo e compacto, estudaremos o caso em que Ω tem tal propriedade. Nesse caso, além de buscarmos uma direção de descida, também gostaríamos que tal direção fosse factível, como descrito na Definição 2.2.1.

Abordaremos métodos de direção factível que começam em um ponto x_0 viável ($x_0 \in \Omega$) e geram uma sequência de pontos viáveis $\{x_k\}$ pela seguinte iteração: se x_k não for estacionário (segundo Teorema 2.3.1), então

$$x_{k+1} = x_k + t_k d_k,$$

em que d_k é direção factível e de descida, ou seja, $\nabla f(x_k)^T d_k < 0$, e o tamanho de passo $t_k > 0$ é escolhido tal que

$$x_k + t_k d_k \in \Omega.$$

Como vimos no Lema 2.3.1, direções factíveis a partir de x_k podem ser escritas da forma

$$d_k = \bar{x}_k - x_k,$$

em que $\bar{x}_k$ é algum ponto viável diferente de x_k. Portanto, com o fato de que $x_k + t_k d_k \in \Omega$, os métodos de direção factíveis podem ser escritos da seguinte maneira

$$x_{k+1} = t_k \bar{x}_k + (1 - t_k) x_k, \tag{18}$$

em que $t_k \in (0, 1]$, e se x_k não é estacionário,

$$\bar{x}_k \in \Omega, \qquad \nabla f(x_k)^T (\bar{x}_k - x_k) < 0. \tag{19}$$

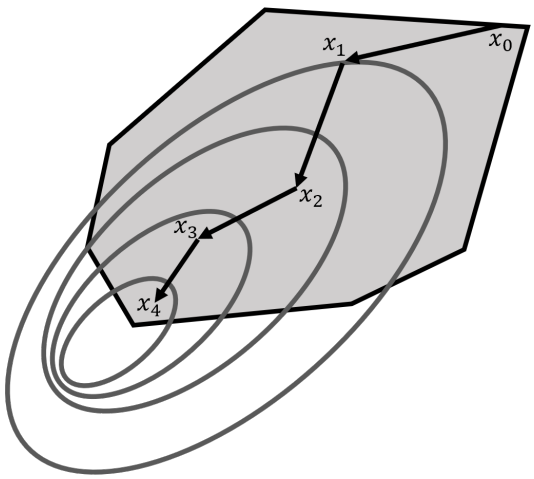

Figura 15 – Ilustração do método de direção factível e de descida

Perceba que como Ω é convexo, temos que $t_k\bar{x}_k + (1-t_k)x_k \in \Omega$ para todo $t_k \in (0,1]$ quando $x_k \in \Omega$. Logo, a sequência de iterados $\{x_k\}$ gerada é viável. Além disso, se x_k não é estacionário, sempre existirá uma direção factível $\bar{x}_k - x_k$ com a propriedade da desigualdade (19), já que, caso contrário, teríamos $\nabla f(x_k)^T(x-x_k) \geq 0$ para todo $x \in \Omega$ e aí pelo Teorema 2.3.1 x_k seria estacionário.

Daqui em diante assumiremos que a direção factível d_k em qualquer método de direção factível é da forma $d_k = \bar{x}_k - x_k$ tal que $\bar{x}_k$ satisfaz a desigualdade (19). E para a escolha do tamanho de passo utilizaremos as mesmas regras abordadas no Capítulo 3.

No restante do capítulo abordaremos os dois métodos que utilizaremos para resolução do POP. Primeiramente estudaremos a ideia geral do método de *Frank-Wolfe* e sua convergência global e em seguida o mesmo para o método de *Gradiente Projetado*. Feito o estudo dos métodos, serão apresentados experimentos numéricos com problemas artificiais, conduzidos a fim de avaliar a performance dos dois métodos em diferentes cenários.

4.1 MÉTODO DE FRANK-WOLFE

A maneira mais natural de gerar direções factíveis $d_k = \bar{x}_k - x_k$ de modo a satisfazer a condição (19) é resolver o seguinte subproblema

$$\begin{aligned} \underset{x}{\text{minimizar}} \quad & \nabla f(x_k)^T(x-x_k) \\ \text{sujeito a} \quad & x \in \Omega. \end{aligned} \tag{20}$$

A função objetivo deste subproblema é inspirada em uma aproximação linear para f a partir de x_k: $f(x) \approx f(x_k) + \nabla f(x_k)^T(x-x_k)$. Para garantir que (20) tenha solução, vamos assumir que o conjunto Ω é compacto (assim a existência de minimizador é assegurada pelo Teorema 2.1.1). Denotaremos uma solução de (20) por $\bar{x}_k$.

Esta é a base do método de *Frank-Wolfe*, que se enquadra na classe de algoritmos que usam direções factíveis e de descida. O Algoritmo 2 apresenta um protótipo de algoritmo de Frank-Wolfe com busca linear.

Algoritmo 2: Frank-Wolfe.

Entrada: Dados $x_0 \in \Omega$ e $\sigma > 0$ faça $k = 0$.
Passo 1. Calcule $\nabla f(x_k)$ e

$$\bar{x}_k \in \underset{x\in\Omega}{\text{argmin}} \nabla f(x_k)^T(x-x_k)$$

Passo 2. Se $\nabla f(x_k)^T(\bar{x}_k - x_k) \geq 0$, pare. Caso contrário, defina $d_k = \bar{x}_k - x_k$ e faça $t = 1$:

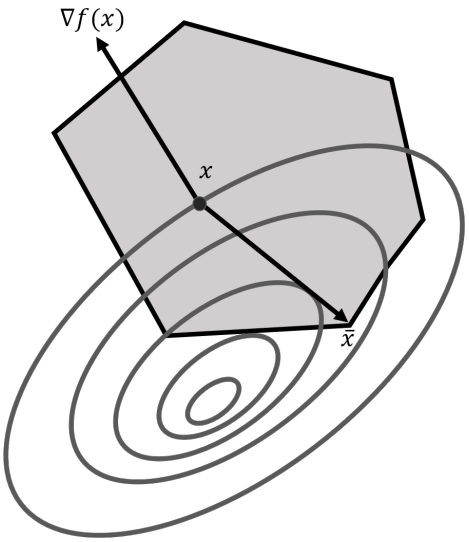

Figura 16 – Direção factível $\bar{x} - x$ a partir do ponto x. $\bar{x}$ é o ponto de Ω que fica mais distante ao longo da direção do $-\nabla f(x)$.

Enquanto

$$f(x_k + td_k) > f(x_k) + \sigma t \nabla f(x_k)^T d_k$$

faça $t \leftarrow \frac{t}{2}$

Passo 3. Faça $t_k = t$, $x_{k+1} = t_k \bar{x}_k + x_k(1 - t_k)$ e $k \leftarrow k + 1$ e volte ao Passo 1.

Claramente, para que faça sentido que este método seja utilizado, o subproblema (20), que precisa ser resolvido no Passo 1, deve ser mais simples que o problema original. E este é o caso quando Ω é o simplex unitário (estudaremos este fato com mais detalhes na Seção 4.1.1).

O critério de parada no Passo 2 segue do Teorema 2.3.1 e, como discutimos anteriormente, a direção d_k escolhida da forma apresentada é factível e de descida. Utilizamos a busca linear inexata no algoritmo mas esta poderia ser substituída por uma busca linear exata, desde que encontrar o minimizador de f ao longo da direção d_k não seja muito difícil. Como Ω é convexo, o iterado seguinte definido no Passo 3 é ponto viável.

Por argumentos similares aos da Seção 3.3 é possível mostrar que todo ponto limite da sequência $\{x_k\}$ gerada pelo Algoritmo 2 é um ponto estacionário para o problema (2). Para tanto, precisamos mostrar que a sequência de direções associada $\{d_k\}$ é *gradient related*. Isto é feito no seguinte lema:

Lema 4.1.1. *A sequência de direções $\{d_k\}$ gerada pelo Algoritmo Frank-Wolfe é gradient related.*

Demonstração. Sejam $\{x_k\}_{k \in K}$ subsequência que converge a um ponto não-estacionário $\hat{x}$ e a subsequência de direções correspondentes $\{d_k\}_{k \in K}$. Suponha por absurdo que

$\nabla f(x_k)^T(\bar{x}_k - x_k) \geq 0$. Mas, pela definição de $\bar{x}_k$

$$\forall x \in \Omega \ : \ \nabla f(x_k)^T(x - x_k) \geq \nabla f(x_k)^T(\bar{x}_k - x_k) \geq 0,$$

logo, do Teorema 2.3.1 temos que x_k é ponto estacionário, o que contradiz a hipótese. Portanto, temos

$$\nabla f(x_k)^T d_k = \nabla f(x_k)^T(\bar{x}_k - x_k) < 0, \forall k,$$

o que implica que

$$\lim_{k\to\infty} \sup_{k\in K} \nabla f(x_k)^T d_k \leq \nabla f(\hat{x})^T d_k < 0.$$

Como Ω é compacto, segue que a subsequência das direções d_k é limitada. Portanto, temos que $\{d_k\}$ é *gradient related*. □

4.1.1 Subproblema de Frank-Wolfe no Simplex unitário

Lembramos que no POP o conjunto viável Ω correspondente ao simplex unitário $\Delta_n = \{x \in \mathbb{R}^n \ : \ \sum_{i=1}^n x_i = 1, x \geq 0\}$ que é conjunto convexo e compacto. Portanto, os subproblemas de Frank-Wolfe estão bem definidos. Mais ainda, como veremos a seguir, estes subproblemas admitem solução analítica.

Perceba que o subproblema (20) com $\Omega = \Delta_n$ pode ser escrito como:

$$\begin{aligned} \underset{x}{\text{minimizar}} \quad & c^T x =: g(x) \\ \text{sujeito a} \quad & \sum_{i=1}^n x_i = 1, \quad x \geq 0, \end{aligned} \tag{21}$$

em que $c = \nabla f(x_k)$. Note que $\nabla g(x) = c, \ \forall x \in \Omega$ e agora defina $\bar{c} \in \mathbb{R}^n$ tal que todas as entradas são iguais a $c_{min} = min\{c_j : j = 1, \ldots, n\}$. Como $x \geq 0$, temos que

$$c^T x \geq \bar{c}^T x.$$

Além disso, como $\sum_{i=1}^n x_i = 1$, veja que

$$\bar{c}^T x = c_{min}.$$

Agora, observe que para $x^* = e_i$, o i-ésimo vetor canônico de $\mathbb{R}^n$, em que i é um dos índices tais que $c_i = c_{min}$, obtemos que $c^T x^* = c_{min}$. Ou seja,

$$c^T(x - x^*) = c^T x - c^T x^* = c^T x - c_{min} \geq 0,$$

para todo $x \in \Omega$. Portanto, temos que x^* é um minimizador global do subproblema (21).

O custo de resolver (21) é $O(n)$, que é custo de encontrar a menor componente do vetor c.

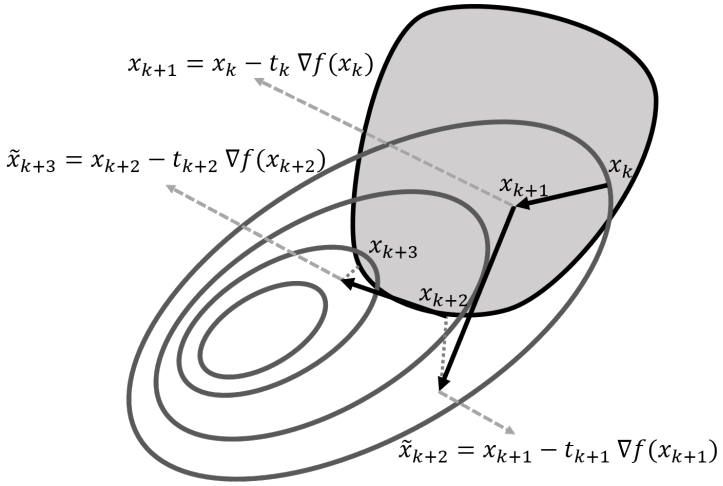

Figura 17 – Ilustração de algumas iterações do método GP

4.2 GRADIENTE PROJETADO

O método do Gradiente Projetado é um método de direção factível no qual a direção de busca $d_k = \bar{x}_k - x_k$ é obtida de

$$\bar{x}_k = P_\Omega(x_k - \nabla f(x_k)), \tag{22}$$

em que $P_\Omega(x)$ denota a projeção (com respeito à norma euclidiana) de x sobre o conjunto convexo e fechado Ω. Para obtermos o vetor $\bar{x_k}$ damos um passo na direção $-\nabla f(x_k)$ e em seguida projetamos $x_k - \nabla f(x_k)$ no conjunto Ω. Logo, $\bar{x_k}$ é viável. Por fim, damos um passo $t_k > 0$ na direção do vetor $d_k = \bar{x_k} - x_k$ para obter o novo iterado $x_{k+1} = x_k + t_k d_k$.

Esta é a base do método de *Gradiente Projetado*, com esquema de busca linear, que é descrito no Algoritmo 3.

Algoritmo 3: Gradiente Projetado.

Entrada: Dados $x_0 \in \Omega$ e $\sigma > 0$ faça $k = 0$.
Passo 1. Calcule $\nabla f(x_k)$ e

$$\bar{x}_k = P_\Omega(x_k - \nabla f(x_k))$$

Passo 2. Se $\bar{x}_k = x_k$, pare. Caso contrário, defina $d_k = \bar{x}_k - x_k$ e faça $t = 1$: Enquanto

$$f(x_k + td_k) > f(x_k) + \sigma t \nabla f(x_k)^T d_k$$

faça $t \leftarrow \frac{t}{2}$
Passo 3. Faça $t_k = t$, $x_{k+1} = t_k \bar{x}_k + x_k(1 - t_k)$, $k \leftarrow k + 1$ e volte ao Passo 1.

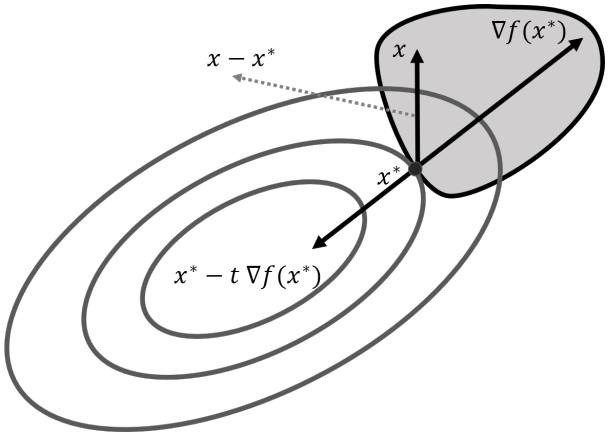

Figura 18 – Ilustração da Proposição 4.2.1, tem-se $x^* = P_\Omega(x^* - \nabla f(x^*)$ e o vetor $x - x^*$ para todo $x \in \Omega$ faz um ângulo menor ou igual a 90° com o gradiente $\nabla f(x^*)$.

Proposição 4.2.1. *$x^* = P_\Omega(x^* - \nabla f(x^*))$ se, e somente se, x^* é estacionário.*

Demonstração. ($\Rightarrow$) Do item (c) do Teorema 2.3.5 temos que

$$(x - P_\Omega(x^* - \nabla f(x^*)))^T(x^* - \nabla f(x^*) - P_\Omega(x^* - \nabla f(x^*))) \leq 0, \qquad \forall x \in \Omega.$$

Substituindo $P_\Omega(x^* - \nabla f(x^*)) = x^*$, obtemos o que desejávamos:

$$(x - x^*)^T(x^* - \nabla f(x^*) - x^*) \leq 0$$

$$\nabla f(x^*)^T(x - x^*) \geq 0, \qquad \forall x \in \Omega.$$

($\Leftarrow$) Por outro lado, seja x^* ponto estacionário de f em Ω. Então

$$\|x^* - P_\Omega(x^* - \nabla f(x^*))\|^2 = (x^* - P_\Omega(x^* - \nabla f(x^*)))^T(x^* - P_\Omega(x^* - \nabla f(x^*)))$$

$$= (x^* - P_\Omega(x^* - \nabla f(x^*)))^T(x^* - \nabla f(x^*) - P_\Omega(x^* - \nabla f(x^*)) + \nabla f(x^*))$$

$$= (x^* - P_\Omega(x^* - \nabla f(x^*)))^T(x^* - \nabla f(x^*) - P_\Omega(x^* - \nabla f(x^*))) + (x^* - P_\Omega(x^* - \nabla f(x^*)))^T \nabla f(x^*).$$

Do item (c) do Teorema 2.3.5 e do fato que x^* é ponto estacionário, temos que ambas as parcelas da soma acima são ≤ 0. Logo,

$$\|x^* - P_\Omega(x^* - \nabla f(x^*))\|^2 = 0.$$

Portanto, $x^* = P_\Omega(x^* - \nabla f(x^*))$. □

A Proposição 4.2.1 justifica o critério de parada no Passo 2 do Algoritmo 3.

Vale destacar que o principal custo computacional do Algoritmo 3 está no cálculo da projeção apresentada no Passo 1. Logo, ele passa a ser interessante quando tal projeção pode ser calculada de maneira eficiente. Isso normalmente ocorre quando Ω tem uma estrutura relativamente simples, por exemplo, uma "caixa" em $\mathbb{R}^n$, $\Omega = \{x \in \mathbb{R}^n \mid \ell \leq x \leq u\}$.

No caso que nos interessa, quando Ω é o simplex unitário, ou seja, $\Delta_n = \{x \in \mathbb{R}^n \ : \ e^T x = 1, x \geq 0\}$, a projeção é relativamente barata do ponto de vista computacional. Isto será discutido em detalhes Subseção 4.2.1.

Por fim, no Passo 3 como estamos considerando Ω convexo, temos que todos os iterados são pontos viáveis.

Em relação a convergência global do Algoritmo 3, por argumentos similares aos da Seção 3.3 é possível mostrar que todo ponto limite da sequência $\{x_k\}$ gerada pelo algoritmo é um ponto estacionário para o problema (2), desde que sejamos capazes de mostrar que a sequência de direções associada $\{d_k\}$ é *gradient related*. Isto é feito no seguinte lema.

Lema 4.2.1. *A sequência de direções $\{d_k\}$ gerada pelo Algoritmo de Gradiente Projetado é gradient related.*

Demonstração. Sejam $\{x_k\}_{k\in K}$ subsequência que converge a um ponto não-estacionário $\hat{x}$ e a subsequência de direções correspondentes $\{d_k\}_{k\in K}$. Do Teorema 2.3.5(c), temos que

$$\begin{aligned} 0 &\geq (x_k - P_\Omega(x_k - \nabla f(x_k)))^T (x_k - \nabla f(x_k) - P_\Omega(x_k - \nabla f(x_k))) \\ &= \|x_k - P_\Omega(x_k - \nabla f(x_k))\|^2 + (P_\Omega(x_k - \nabla f(x_k)) - x_k)^T \nabla f(x_k) \\ &= \|x_k - P_\Omega(x_k - \nabla f(x_k))\|^2 + \nabla f(x_k)^T d_k. \end{aligned}$$

Já que x_k é não estacionário (ie, $x_k \neq P_\Omega(x_k - \nabla f(x_k))$, segue que

$$\nabla f(x_k)^T d_k \leq -\|x_k - P_\Omega(x_k - \nabla f(x_k))\|^2 < 0,$$

para todo k, e em particular para $k \in K$. Dessa forma, tomando limite na subsequência convergente

$$\lim_{k\to\infty} \sup_{k\in K} \nabla f(x_k)^T d_k \leq -\|\hat{x} - P_\Omega(x_k - \nabla f(\hat{x}))\|^2 < 0.$$

Como Ω é compacto, segue que a subsequência das direções $\{d_k\}_{k\in K}$ é limitada. Portanto, temos que a sequência $\{d_k\}$ direção *gradient related*. □

4.2.1 Subproblema do Gradiente Projetado no Simplex unitário

Nesta subseção discutiremos o método para resolver o seguinte problema

$$\min_{x\in\Omega} \|x - c\|^2, \tag{23}$$

em que $\Omega = \Delta_n = \{x \in \mathbb{R}^n \ : \ e^T x = 1, x \geq 0\}$. Denota-se,

$$I_n = \{1, 2, \ldots, n\},$$

$$V = \left\{ x \in \mathbb{R}^n | \sum_{i=1}^{n} x_i = 1 \right\}.$$

Para um subconjunto $I \subset I_n$,

$$X_I = \{x \in \mathbb{R}^n | x_i = 0, \quad \forall i \in I\},$$

$$V_I = X_I \cap V,$$

$$\Omega_I = X_I \cap \Omega,$$

$$n_I = dim(X_I),$$

tal que X_I é um subespaço linear de $\mathbb{R}^n$. De resultados discutidos no artigo (MICHELOT, 1986) obtêm-se um método simples para resolver (23). Primeiro, achamos a projeção no subespaço afim V_I. Segundo, encontra-se $P_{\Omega_J}(c)$ a projeção de c em um conjunto fechado e convexo Ω_J com $I \subset J$. Terceiro, calculamos a projeção em um subespaço linear X_J. Substitua I por J e repita o processo. Obtemos o seguinte algoritmo.

Algoritmo 4: Projeção no Simplex unitário.

Entrada: Dados $c \in \mathbb{R}^n$, faça $x = c$ e $I = \varnothing$.

Passo 1. Calcule $\tilde{x} = P_{V_I}(x)$.

Passo 2. Se $\tilde{x} \geq 0$, pare. Caso contrário, atualize I por $I \cup \{i | \tilde{x} < 0\}$, e x por $P_{X_I}(\tilde{x})$ e volte ao Passo 1.

No Passo 1 é necessário o cálculo de $\tilde{x} = P_{V_I}(x)$, que é dado por

$$\tilde{x}_i = x_i - \frac{\sum_i x_i - 1}{n_I}, \qquad \text{se} \quad i \notin I,$$

$$\tilde{x}_i = 0, \qquad \text{se} \quad i \in I.$$

E no Passo 2 é necessário o cálculo de $x = P_{X_I}(\tilde{x})$, que consiste em

$$x_i = \tilde{x}_i, \quad \text{se} \quad \tilde{x}_i \geq 0,$$

$$x_i = 0, \quad \text{caso contrário.}$$

O algoritmo converge em no máximo n iterações, já que a dimensão n_I do subespaço X_I decresce ao menos uma unidade a cada iteração.

O custo total do algoritmo, no pior caso, é $O(n \log n)$ (MICHELOT, 1986) (GONÇALVES, 2016).

4.3 EXPERIMENTOS NUMÉRICOS EM PROBLEMAS ARTIFICIAIS

Nesta seção, vamos considerar problemas artificiais a fim de avaliar o desempenho dos métodos de Frank-Wolfe e Gradiente Projetado que acabamos de discutir.

Para isso, motivados pelo POP, vamos considerar o problema de minimizar uma função quadrática, com Hessiana positiva semidefinida, sobre o simplex unitário, isto é

$$\begin{aligned} \underset{x}{\text{minimizar}} \quad & \frac{1}{2}x^T Gx - x^T \mu \\ \text{sujeito a} \quad & \sum_{i=1}^{n} x_i = 1, \\ & x \geq 0. \end{aligned} \tag{24}$$

Em (24), a matriz $G \in \mathbb{R}^{n\times n}$ e o vetor $\mu \in \mathbb{R}^n$ foram gerados de forma aleatória. Para gerar a matriz G, utilizamos um vetor (aleatório) auxiliar v com $\|v\| = 1$ e então definimos a matriz ortogonal

$$Q = I_n - 2vv^T,$$

em que I_n denota a matriz identidade de ordem n. Vamos considerar as colunas de Q como autovetores de G. A seguir, definimos um vetor $\lambda \in \mathbb{R}^n$, cujos os elementos são autovalores (escolhidos previamente) da matriz G e definimos

$$G = Q\text{diag}(\lambda)Q^T,$$

em que diag(λ) é uma matriz diagonal, cujos elementos da diagonal são as entradas do vetor λ. Com isso temos uma matriz G simétrica e, escolhendo $\lambda_i > 0$, $i = 1, \ldots, n$, definida positiva.

Para o vetor $\mu \in \mathbb{R}^n$, as componentes μ_i foram geradas de uma distribuição uniforme no intervalo $[0, 1]$.

Utilizamos os dois métodos estudados, Frank-Wolfe (FW) e Gradiente Projetado (GP), para a resolução do problema (24). Consideramos duas variações dos métodos: com busca linear exata e busca linear inexata (descritas no Capítulo 3) utilizando como parâmetros: $\sigma = 10^{-4}$, $\beta = \frac{1}{2}$. Para a tolerância nos critérios de parada utilizamos $\varepsilon = 10^{-4}$. Também estabelecemos um número máximo de 10000 iterações e como ponto inicial o vetor canônico $e_1 \in \mathbb{R}^n$.

Nas tabelas a seguir são apresentados: a dimensão n, *it.* que é o número de iterações, *av.* que é o número de avaliações de função, o tempo em segundos e *fmin* que é o valor funcional no ponto retornado por cada algoritmo.

Realizamos testes com diferentes distribuições de autovalores as quais estão descritas nas tabelas. Isto foi feito para estudar o impacto da distribuição de autovalores de G e, consequentemente, seu número de condicionamente no desempenho dos métodos de primeira ordem considerados.

4.3.1 Testes utilizando busca linear inexata

	Frank-Wolfe				Gradiente Projetado			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	3	24	0.005	0.0787	14	43	0.074	0.0787
3	22	185	0.010	-0.1315	17	51	0.006	-0.1315
5	38	371	0.020	-0.0891	17	51	0.006	-0.0891
100	171	1760	0.192	-0.8203	16	48	0.012	-0.8203
500	303	2866	0.528	-0.9284	15	45	0.015	-0.9284
1000	267	2386	1.104	-0.9475	15	45	0.029	-0.9476

Tabela 1 – Todos os autovalores iguais a π.

Na Tabela 1 vemos que o número de iterações realizadas por FW foi menor que no método GP apenas para $n = 2$. O número de iterações de GP se manteve estável com o aumento da dimensão, enquanto que o número de iterações de FW parece aumentar com a dimensão.

Em termos de tempo computacional, fora o primeiro problema, o GP foi mais rápido para os problemas das de mais dimensões. Embora a iteração de GP seja mais cara que a de FW, já que o problema (20) é, em geral, mais fácil que (23), notamos que o tempo por iteração de GP foi menor que o tempo por iteração de FW. Uma possível explicação é dada pelo alto número de avaliações de função de FW se comparado a GP. Aqui vale ressaltar que as matrizes G geradas são, em geral, densas e com isso o produto matriz-vetor Gx custa $O(n^2)$.

Os valores funcionais do ponto encontrado coincidem até a quarta casa decimal para todas as dimensões exceto a última. Isto nos mostra que, apesar da diferença de velocidade, os métodos apresentaram precisões praticamente idênticas no caso que todos os autovalores são iguais.

Destacamos que este é o cenário "mais fácil", uma vez que o número de condicionamento[1] de G é 1 (o menor possível).

	Frank-Wolfe				Gradiente Projetado			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	9	122	0.005	-0.4627	8578	25735	1.931	-0.4627
3	23	213	0.010	0.0428	14	54	0.004	0.0428
5	42	414	0.018	-0.0090	13	58	0.005	-0.0090
100	10001	147702	13.403	-0.2633	1077	9002	1.031	-0.2649
500	10001	175907	31.924	-0.3130	3310	35493	6.110	-0.3131
1000	10001	153245	73.441	-0.2632	10001	116570	62.334	-0.2693

Tabela 2 – Todos os autovalores variando linearmente ($\lambda_i = 2i$).

[1] Número de condicionamento de uma matriz não singular G é igual ao maior autovalor dividido pelo menor autovalor da matriz.

Na Tabela 2 vemos que o número de iterações de FW foi superior para todas as dimensões exceto duas: $n = 2$ e $n = 1000$. A Figura 19 ilustra o que aconteceu com o GP para $n = 2$: ela mostra a trajetória dos iterados.

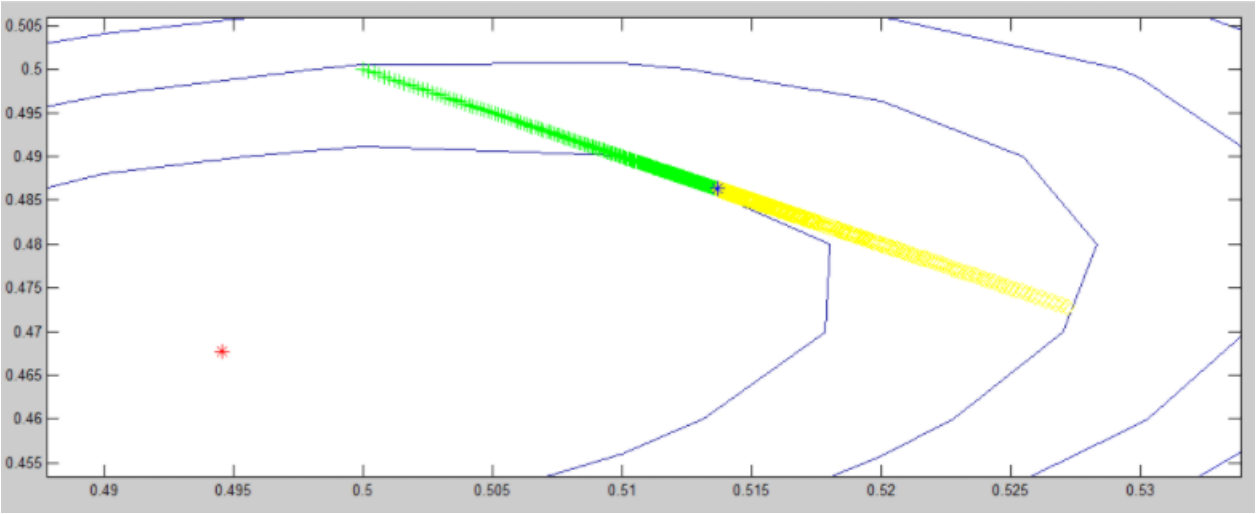

Figura 19 – Temos a trajetória dos últimos iterados em $\mathbb{R}^2$ (iterados pares em verde e os iterados ímpares em amarelo), o minimizador irrestrito representado pelo "$*$" vermelho e o minimizador restrito representado pelo "$*$" azul (além das curvas de nível da função), tudo numa vizinhança de x^*.

Com exceção da dimensão $n = 2$, o número de iterações de GP foi menor do que o de FW. É importante destacar que FW atingiu o número máximo de iterações para $n = 100, 500, 1000$, enquanto que GP também falhou para $n = 1000$.

Para os problemas nos quais ambos funcionaram, o valor funcional obtido foi praticamente o mesmo.

	Frank-Wolfe				**Gradiente Projetado**			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	17	201	0.009	-0.3013	260	2822	0.142	-0.3013
3	10001	140415	6.076	2.3946	7259	77846	4.892	2.3848
5	9103	126960	5.633	2.8136	418	3164	0.242	2.8126
100	10001	151606	14.225	-0.5096	4085	44381	6.342	-0.5099
500	4822	69423	10.198	-0.8090	2696	29048	7.510	-0.8093
1000	1344	15714	7.784	-0.8624	45	325	0.168	-0.8625

Tabela 3 – Todos os autovalores iguais a 22 exceto dois, o menor igual a 0.05 e o maior igual a 1024 com busca linear inexata

Na Tabela 3 mais uma vez vemos que o número de iterações, tempo e avaliações do GP são menores do que os do FW, exceto na dimensão dois. Em destaque ficam a dimensão $n = 5$ e dimensão $n = 1000$ onde o GP precisou de muito menos iterações, avaliações e tempo computacional para alcançar uma aproximação do mínimo. Em praticamente todas as dimensões o GP obteve valores funcionais menores.

No caso $n = 1000$ nos surpreende a pequena quantidade de iterações necessárias do GP, mas analisando o minimizador do problema e o ponto inicial utilizado observamos que, por acaso, começamos o método muito próximo da solução.

Para todas as distribuições de autovalores e busca linear inexata o método GP obteve melhores resultados que o método FW, exceto para a dimensão $n = 2$. Isto nos leva a preferir o método GP para dimensões maiores quando utilizamos a busca linear inexata.

4.3.2 Testes utilizando busca linear exata

Antes de iniciar a análise dos testes utilizando busca linear exata será feito o estudo da resolução do problema (15) no caso da função quadrática do problema (24):

$$\min_{t\in[0,1]} f(x_k + td_k) = \frac{1}{2}(x_k + td_k)^T G(x_k + td_k) - (x_k + td_k)^T \mu =: g(t). \tag{25}$$

Calculando a derivada de $g(t)$ obtemos

$$g'(t) = (G(x_k + td_k) - \mu)^T d_k. \tag{26}$$

Para obter t tal que $g'(t) = 0$, resolvemos

$$\begin{aligned} 0 &= (G(x_k + td_k) - \mu)^T d_k \\ &= ((x_k + td_k)^T G - \mu^T) d_k \\ &= (x_k^T G + td_k^T G - \mu^T) d_k \\ &= x_k^T G d_k + td_k^T G d_k - \mu^T dk \end{aligned}$$

ou seja,

$$t = \frac{\mu^T d_k - x_k^T G d_k}{d_k^T G d_k}. \tag{27}$$

E dessa forma, escolhemos $t_k = \min\{t, 1\}$, que também assegura que $x_k + t_k d_k \in \Omega$.

	Frank-Wolfe				**Gradiente Projetado**			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	1	2	0.007	0.0787	1	2	0.027	0.0787
3	13	14	0.002	-0.1315	2	3	0.001	-0.1315
5	168	169	0.020	-0.0891	2	3	0.001	-0.0891
100	4551	4552	1.047	-0.8200	3	4	0.004	-0.8204
500	1467	1468	0.505	-0.9278	3	4	0.005	-0.9284
1000	907	908	0.983	-0.9467	3	4	0.007	-0.9476

Tabela 4 – Todos os autovalores iguais a π.

Na Tabela 4 observamos que o método GP precisou de pouquíssimas iterações para convergir em todas as dimensões. Já o FW precisou de mais, principalmente

para a dimensão $n = 100$ na qual foram necessárias aproximadamente 4500 iterações. Apenas para a dimensão $n = 2$ o FW foi mais rápido que o GP, e a maior diferença de tempo entre os métodos foi para a dimensão $n = 100$, onde o FW precisou de mais iterações do que todos os outros casos. Apesar disso, ambos os métodos resolveram todos os problemas em menos de dois segundos. Os valores funcionais encontrados são praticamente idênticos nas dimensões menores, já nas maiores notamos que o GP foi mais preciso.

	Frank-Wolfe				**Gradiente Projetado**			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	1	2	0.001	-0.4627	1	2	0.001	-0.4627
3	15	16	0.003	0.0428	13	14	0.004	0.0428
5	60	61	0.009	-0.0090	17	18	0.005	-0.0090
100	10001	10001	2.392	-0.2601	626	627	0.360	-0.2649
500	10001	10001	4.760	-0.3131	796	797	0.612	-0.3131
1000	10001	10001	13.744	-0.2417	6090	6091	10.966	-0.2693

Tabela 5 – Todos os autovalores variando linearmente ($\lambda_i = 2i$).

Na Tabela 5 os dois métodos apresentaram número de iterações e tempo muito parecidos nas primeiras duas dimensões. Já na dimensão $n = 5$ vemos uma notável diferença e, para dimensões maiores FW falhou, pois atingiu o número máximo de iterações. O GP apresentou mais dificuldade na dimensão $n = 1000$.

Apesar do FW estourar o limite de iterações na dimensão 1000 e o GP não, o GP não foi muito mais rápido que o FW. Porém, quando analisamos o valor *fmin* o FW ainda estava "longe" do minimizador comparado ao GP.

Apenas para as dimensões 100 e 1000 os valores *fmin* são diferentes, nas outras dimensões os métodos tiveram a mesma precisão, considerando quatro casas decimais, até mesmo para a dimensão 500 na qual, apesar de FW ter estourado o número de iterações, a solução deste estava "próxima" a solução de GP.

	Frank-Wolfe				**Gradiente Projetado**			
n	**it.**	**av.**	**tempo(s)**	**fmin**	**it.**	**av.**	**tempo(s)**	**fmin**
2	1	2	0.009	-0.3013	1	2	0.001	-0.3013
3	10001	10001	1.384	2.406	1111	1112	0.316	2.3848
5	19	20	0.003	2.8126	9	10	0.003	2.8126
100	10001	10001	2.787	-0.5085	171	172	0.092	-0.5099
500	10001	10001	3.848	-0.8082	215	216	0.200	-0.8093
1000	10001	10001	13.646	-0.8614	116	117	0.192	-0.8625

Tabela 6 – Todos os autovalores iguais exceto dois, um deles igual a 0.05 e outro igual a 1024.

Na Tabela 6 logo nos chama a atenção o caso $n = 3$ no qual FW estourou o limite de iterações. Para o GP o número também foi elevado. No entanto, ressaltamos

que estes problemas possuem a distribuição de autovalores mais desafiadora, uma vez que ela implica num elevado número de condição de G. Neste cenário é esperado que métodos de primeira ordem (que usam apenas informação de derivadas primeiras) encontrem dificuldades.

Fora isso, assim como no caso anterior, para as dimensões 2 e 5 os métodos apresentaram resultados parecidos e as iterações nos casos $n = 100$, $n = 500$ e $n = 1000$ estouraram o limite para o método FW. Para as dimensões maiores o GP foi muito superior ao FW em número de iterações, tempo e precisão.

4.3.3 Conclusão preliminar

Em todos os casos vimos que, exceto para a dimensão dois, o GP foi mais eficiente que o FW usando busca linear inexata e, melhor ainda com a busca linear exata. Assim, em vista destes resultados numéricos, decidimos empregar o GP com busca linear exata para resolver os problemas POP considerados no próximo capítulo.

5 OTIMIZAÇÃO DE PORTFÓLIO DE INVESTIMENTOS

A teoria de portfólio busca modelar os riscos associados a uma coleção de investimentos i com retornos r_i, $i = 1, 2, \ldots, n$, de modo a auxiliar na escolha de um portfólio "ótimo" no sentido de minimizar riscos e maximizar o retorno esperado (NOCEDAL; WRIGHT, 1999).

Os retornos r_i são variáveis aleatórias que podem ser caracterizadas por seu valor esperado $\mu_i = E[r_i]$ e sua variância $\sigma_i^2 = E[(r_i - \mu_i)^2]$. Existe sempre uma flutuação do retorno sobre sua média, que indica quais os investimentos com maior risco.

Um portfólio tem um percentual do seu valor total investido em fundos disponíveis e, supondo que todos os n fundos estão disponíveis e a venda a descoberto[1] não seja possível, as restrições são: $\sum_{i=1}^{n} x_i = 1$ e $x \geq 0$. Como já mencionado anteriormente neste trabalho, tais restrições definem o simplex unitário Δ_n (veja Seção 4.1.1).

O retorno de um portfólio é dado pela variável aleatória $R = \sum_{i=1}^{n} x_i r_i$. Deste modo o *retorno esperado* do portfólio é dado por:

$$E(R) = E\left[\sum_{i=1}^{n} x_i r_i\right] = \sum_{i=1}^{n} x_i E[r_i] = x^T \mu. \tag{28}$$

A variância pode ser obtida por leis elementares da estatística. A covariância entre cada par de investimentos é dada pela expressão

$$\sigma_{ij} = E[(r_i - \mu_i)(r_j - \mu_j)], \tag{29}$$

e a correlação por

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j},$$

em que σ_i denota o desvio padrão da variável aleatória r_i.

A correlação de investimentos mede a tendência dos investimentos i e j de se movimentarem na mesma direção (ou não). Quando há investimentos com retornos que sobem ou caem juntos, há covariância positiva e quanto mais perto 1 estiver ρ_{ij}, mais os investimentos são correlacionados. É claro que a correlação pode ser casual, eventual, isto é, não persistente. Já quando se movem em direções contrárias mais próxima de –1 fica ρ_{ij}. Com isso, a variância do retorno R do portfólio pode ser escrita como

$$E[(R - E[R])^2] = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i x_j \sigma_i \sigma_j \rho_{ij} = x^T G x, \tag{30}$$

em que a matriz $G \in \mathbb{R}^{n \times n}$ é simétrica com entradas $G_{ij} = \rho_{ij} \sigma_i \sigma_j$. Verifica-se que G é positiva semidefinida (NOCEDAL; WRIGHT, 1999, Página 441).

[1] A venda a descoberto é uma estratégia que consiste na venda de uma ação que você não possui em carteira. Para fazer isso você realiza duas operações: aluguel da ação que não tem e venda deste mesmo ativo.

Os portfólios mais interessantes têm o retorno esperado $x^T\mu$ maior e com menor risco $x^T Gx$. (MARKOWITZ, 1952) em seu modelo une em um único problema os objetivos de maximizar o retorno esperado e de minimizar o risco do portfólio com o auxílio de um "parâmetro de aversão ao risco" $\kappa > 0$, para determinar o portfólio ideal:

$$\begin{aligned} \underset{x}{\text{minimizar}} \quad & \frac{\kappa}{2} x^T Gx - x^T \mu \\ \text{sujeito a} \quad & x \in \Delta_n. \end{aligned} \tag{31}$$

A escolha de $\kappa > 0$ depende da preferência do investidor. Os investidores mais conservadores procuram sempre minimizar os riscos e devem escolher um valor alto de κ e para os investidores mais agressivos, que desejam um retorno mais alto (ao preço de um risco maior), o valor de κ deve estar mais próximo de zero.

5.1 ESTIMATIVAS PARA μ, σ E ρ

Na vida real, a dificuldade no uso do modelo de Markowitz está na definição dos retornos, variâncias e covariâncias esperados nos investimentos. Uma opção é o uso de dados históricos, por exemplo, dos últimos 5 anos, para estimar as quantidades μ_i, σ_i e ρ_{ij}. Esta expectativa não é sinal de que estes dados sejam idênticos aos que ocorrerão no futuro, por este motivo, muitos profissionais também utilizam suas observações e expectativas do mercado para chegar a estes valores.

Nas próximas subseções, com base em (ROSS STEPHEN, 2015) e (BARBETTA, 2010), descreveremos uma maneira de estimar μ_i, σ_i e ρ_{ij} com base em dados históricos.

5.1.1 Taxa de retorno de um ativo

Dado um investimento, a taxa de retorno simples (T_r) será dada por

$$T_r = \frac{P_f - P_i}{P_i},$$

em que P_f é o preço final do investimento no período analisado, P_i o preço inicial do investimento no mesmo período.

Essa taxa é preferível quando lidamos com vários ativos ao longo do mesmo período de tempo. Para obtermos a taxa histórica de retorno, que utilizaremos como nossa taxa de retorno esperada, ou seja, μ, faremos a média das taxas de retorno simples em um determinado período de tempo (taxa diária, mensal, trimestral, anual).

5.1.2 Risco de um ativo

Como vimos, calcularemos o vetor μ através da média da taxa histórica de retorno e usaremos estimativas para a variância, σ^2, que mede a dispersão de um

conjunto de dados em torno da média, para quantificarmos o risco de um ativo. Dado um conjunto de n dados y_i com $i = 1, \ldots, n$, uma estimativa para a variância é dada por

$$\sigma_y^2 = \frac{\sum_i^n (y_i - \bar{y})^2}{n-1} \tag{32}$$

em que $\bar{y}$ é a média dos dados y_i.

Lembramos que quanto maiores os valores de variância de um ativo, maior o risco desse ativo.

5.1.3 Medindo a relação entre ativos

Para quantificarmos a correlação entre dois ativos nós calcularemos a covariância entre eles. Dados dois conjuntos de n dados y_i e z_i, com $i = 1, \ldots, n$ temos que uma estimativa para a covariância será dada por

$$\sigma_{yz} = \frac{\sum_{i=1}^n (y_i - \bar{y})(z_i - \bar{z})}{n-1} \tag{33}$$

em que $\bar{y}$ e $\bar{z}$ são as médias dos dados y_i e z_i, respectivamente.

A correlação entre os dois conjuntos de dados é então estimada por

$$\rho_{yz} = \frac{\sigma_{yz}}{\sigma_y \sigma_z}. \tag{34}$$

Exemplo 5.1.1. Veja que para dois ativos i e j podemos obter $\mu_i = \mu_j = 15\%$ porém a média do ativo i pode ter sido composta de taxas anuais iguais a 14%, 16%, 13%, 17% ao passo que a média do ativo j seja composta de taxas anuais iguais a 50%, –20%, –20%, 50%. No caso do ativo i vemos que caso tivéssemos investido nosso dinheiro nele nós obteríamos uma rentabilidade mais estável ao longo do tempo ao passo que no ativo j obteríamos uma grande variabilidade na rentabilidade no mesmo período. Intuitivamente vemos que o ativo j é mais arriscado que o ativo i, então calculemos a variância dos dois ativos:

$$\sigma_i^2 = \frac{(14\% - 15\%)^2 + (16\% - 15\%)^2 + (13\% - 15\%)^2 + (17\% - 15\%)^2}{3}$$

$$\sigma_j^2 = \frac{(50\% - 15\%)^2 + (-20\% - 15\%)^2 + (-20\% - 15\%)^2 + (50\% - 15\%)^2}{3}$$

Ou seja,

$$\sigma_i^2 = \frac{0.01\% + 0.01\% + 0.04\% + 0.04\%}{3} = \frac{0.1\%}{3} = 0.033\% = 0.00033$$

$$\sigma_j^2 = \frac{12\% + 12\% + 12\% + 12\%}{3} = \frac{48\%}{3} = 16\% = 0.16$$

Por fim obtemos

$$\sigma_i = \sqrt{0.033\%} = 1.8\% = 0.018$$

$$\sigma_j = \sqrt{16\%} = 40\% = 0.4$$

Como $\sigma_i < \sigma_j$ podemos confirmar a nossa intuição. Agora calculemos a covariância entre esses dois ativos:

$$\sigma_{ij} = \frac{(-1\%)(35\%) + (1\%)(-35\%) + (-2\%)(-35\%) + (2\%)(35\%)}{3} = \frac{0.021}{3} = 0.007$$

E, por fim, a correlação entre eles será dada por:

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j} = \frac{0.007}{(0.018)(0.4)} \approx 0,972$$

5.2 UM ESTUDO DE CASO

Para realizar um estudo de caso utilizamos sete ações da bolsa brasileira indicadas pelo economista Cristiano Martins Caetano[2] que são VIVT3.SA, QUAL3.SA, ALUP11.SA, DIRR3.SA, ENBR3.SA, HYPE3.SA, ODPV3.SA (esta será a ordem das variáveis x_i). Os dados das ações utilizadas são das datas 03/08/2016 até 02/08/2021, foram extraídos do *Yahoo! Finance* e passaram por um pré-processamento cujos detalhes são descritos no Apêndice A.

O vetor de retornos μ obtido é

$$\mu = (0.127120, 0.209334, 0.146460, 0.295909, 0.134735, 0.150473, 0.073672)^T, \tag{35}$$

e a matriz G

$$G = \begin{pmatrix} 0.086632 & 0.033705 & 0.022811 & 0.041212 & 0.033084 & 0.034601 & 0.023447 \\ 0.033705 & 0.218631 & 0.041263 & 0.076657 & 0.056033 & 0.063715 & 0.046171 \\ 0.022811 & 0.041263 & 0.057125 & 0.042257 & 0.033444 & 0.028971 & 0.021727 \\ 0.041212 & 0.076657 & 0.042257 & 0.187278 & 0.058139 & 0.056374 & 0.043403 \\ 0.033084 & 0.056033 & 0.033444 & 0.058139 & 0.087081 & 0.037045 & 0.028224 \\ 0.034601 & 0.063715 & 0.028971 & 0.056374 & 0.037045 & 0.111035 & 0.034724 \\ 0.023447 & 0.046171 & 0.021727 & 0.043403 & 0.028224 & 0.034724 & 0.097109 \end{pmatrix} \tag{36}$$

Nos experimentos foram utilizados diversos valores de κ e analisados os vetores x^*, soluções de (31) que correspondem às porcentagens das ações. Como discutido no fim do Capítulo 4, foi utilizado o método GP com busca linear exata para resolver os problemas já que este apresentou o melhor desempenho nos experimentos daquele capítulo. A escolha do ponto inicial x_0 foi

$$x_0 = \left(\frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}\right)^T,$$

[2] Amigo pessoal do autor, graduado em Ciências Ecônomicas pela Universidade Estadual de Ponta Grossa

e tolerância utilizada no critério de parada foi $\varepsilon = 10^{-4}$. Em todos os testes feitos o algoritmo fez no máximo 1685 iterações.

Os valores de κ utilizados em experimentos preliminares pertencem ao conjunto

$$\{0.05, 1, 5, 10, 50, 100, 500, 2000, 5000\},$$

mas discutiremos os resultados apenas para alguns destes valores.

Para os valores $\kappa = 0.05$ e $\kappa = 1$ obtivemos as soluções dadas por

$$x^* = (0, 0, 0, 1, 0, 0, 0)^T \quad \text{para} \quad \kappa = 0.05,$$

e

$$x^* = (0, 0.0957, 0, 0.9043, 0, 0, 0)^T \quad \text{para} \quad \kappa = 1.$$

Como para valores pequenos de κ o investidor tolera mais o risco, o peso maior fica com o retorno esperado, o que justifica escolher os investimentos com maior retorno (ainda que este seja o mais arriscado). Assim, investidores mais agressivos escolheriam, dentre as sete ações indicadas, colocar quase todo seu montante na ação DIRR3.SA. Isto era de se esperar, já que esta ação apresenta um retorno muito maior que o das outras ações.

Para $\kappa = 5$ obtivemos a solução

$$x^* = (0.1811, 0.0386, 0.4495, 0.1952, 0.0375, 0.0913, 0.0068)^T.$$

Neste valor de κ temos uma aversão ao risco baixa à moderada. O primeiro fato que chama atenção nesta solução é o peso dado às ações VIVT3.SA e ALUP11.SA cujos os retornos esperados são *intermediários* entre as ações consideradas. Este fato pode ser esclarecido comparando as colunas 1 e 3 da matriz G com as outras. Percebemos que seus valores são menores, em geral, que o das outras ações. Ainda neste nível de risco a ação DIRR3.SA tem uma porcentagem considerável, o que nos mostra que ainda há um bom peso dado aos retornos esperados já que esta ação tem o segundo maior risco entre as sete consideradas.

Para $\kappa = 10$ obtivemos a solução

$$x^* = (0.1976, 0.0047, 0.4400, 0.0742, 0.0847, 0.0930, 0.1058).$$

Neste valor de κ já temos uma aversão ao risco média. Nota-se que a ação DIRR3.SA perdeu bastante peso em comparação aos outros dois resultados analisados. Analisando as colunas $1, 3, 7$ da matriz G percebe-se que as três ações que apresentaram maiores porcentagens na solução obtida apresentam valores menores, em geral, que as colunas das outras ações. Vemos também que as correlações das ações VIVT3.SA e ALUP11.SA com a ODPV3.SA são as menores (ver última linha da matriz *CORR* na Figura 29, Apêndice A), ou seja, a alta (ou a baixa) no valor das ações VIVT3.SA

e ALUP11.SA está pouco relacionada com da ação ODPV3.SA que apresenta menor retorno esperado entre todas as ações do portfólio.

Já nos valores de $\kappa \geq 50$ considerados, obtivemos soluções muito parecidas (até mesmo para $\kappa = 5000$. Para $\kappa = 2000$ a solução obtida foi

$$x^* = (0.2105, 0, 0.4137, 0, 0.1071, 0.0767, 0.1921). \tag{37}$$

Neste caso temos uma aversão ao risco alta. Dessa forma, era de se esperar que as ações que apresentassem menor covariância (comparada entre si e com as outras) teriam maior porcentagem na solução obtida. E de fato é isso que ocorreu neste caso (e para os outros citados no parágrafo anterior), as ações VIVT3.SA, ALUP11.SA e ODPV3.SA que apresentaram maiores porcentagens. Para a solução (37) o retorno esperado da carteira é igual a $12,74\%$ que é um valor alto, ainda mais para uma carteira de aversão alta ao risco.

5.2.1 Fronteira Eficiente

A base do conteúdo apresentado nesta subseção está na referência (ROSS STEPHEN, 2015, Capítulo 11)

Na função objetivo de (31) nós consideramos tanto o risco dos investimentos, modelado pelo fator $\frac{1}{2}x^T Gx$, quanto o retorno esperado, representado por $\mu^T x$. No entanto, estes dois objetivos são, em geral, conflitantes: normalmente os investimentos com maior retorno esperado são também aqueles com maior risco.

Neste seção tentaremos responder duas perguntas:

1. Fixado um retorno esperado r, o portfólio obtido de (31) é o de menor risco para algum dos valores de κ considerados?

2. Fixado um risco γ, o portfólio obtido de (31) é o de maior retorno para algum dos valores de κ considerados?

Matematicamente, a primeira questão corresponde a verificar se a solução do problema

$$\begin{aligned} \min_x \quad & \frac{1}{2}x^T Gx \\ \text{sujeito a} \quad & x \in \Delta_n \\ & \mu^T x = r, \end{aligned} \tag{38}$$

em que r é o retorno da solução de (31), possui risco maior ou igual ao da solução de (31), para algum valor de κ considerado.

Por outro lado, a segunda questão consiste em verificar se a solução do problema

$$\begin{aligned} \min_{x} \quad & -\mu^T x, \\ \text{sujeito a} \quad & x \in \Delta_n \\ & \frac{1}{2} x^T G x = \gamma, \end{aligned} \tag{39}$$

em que γ é o risco da solução de (31), possui retorno menor ou igual ao da solução de (31), para algum valor de κ considerado.

No caso em que a resposta a estas questões seja afirmativa, diremos que a solução de (31) está na *fronteira eficiente* do problema.

Para resolver o problema (38) utilizaremos a função *quadprog* do *Octave* e a função *fmincon* para resolver o problema (39). Através de um código implementado, em *Octave* [3] verificou-se que para todos os valores de κ utilizados na Seção 5.2 a resposta às duas questões acima foi afirmativa. Então todas as soluções encontradas para o problema (31), para os valores de κ utilizados anteriormente, estão na fronteira eficiente ou muito próximas da mesma.

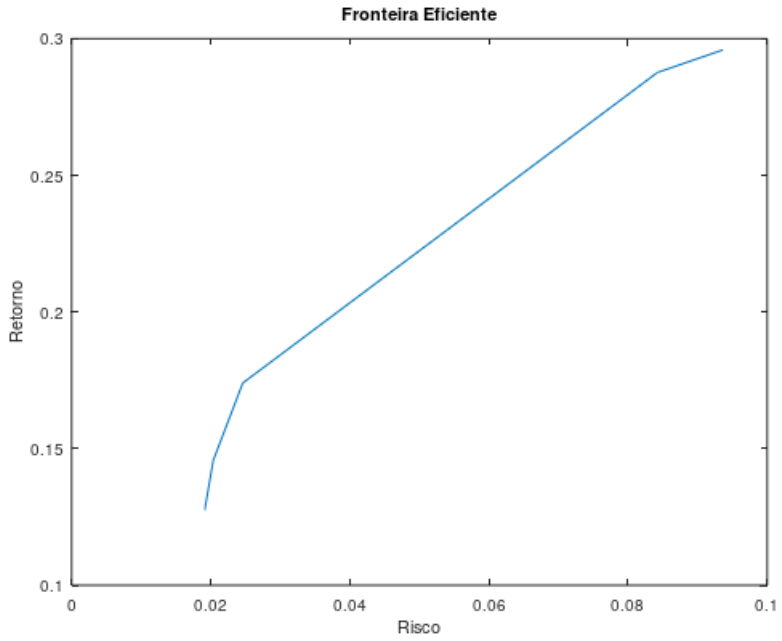

Figura 20 – Uma aproximação (grosseira) da fronteira eficiente, feita a partir dos retornos e riscos correspondentes as soluções encontradas para $\kappa = 0.05, 1, 5, 10, 50, 100$ das ações analisadas.

[3] Função “fronteira.m” descrita no Apêndice A e disponível em `https://drive.google.com/drive/folders/1NXOjm8v6xE-gHi4V1zzU4F7dYaFSUcWe?usp=sharing`

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Os métodos de primeira ordem estudados, Gradiente Projetado (GP) e Frank-Wolfe (FW), apresentaram um desempenho esperado para os testes aleatórios: os métodos com busca linear exata foram melhores, mas ainda encontraram dificuldades para problemas com número de condição elevado.

No entanto, no estudo de caso realizado, no qual consideramos o problema (31) para as ações indicadas na Seção 5.2, o GP apresentou resultados satisfatórios e que coincidiram (dentro da precisão requerida) com as soluções fornecidas pelas rotinas *quadprog* e *fmincon* do Octave.

O modelo de Markowitz, (MARKOWITZ, 1952), para seleção de portfólio é muito simples, o que pode ajudar, principalmente, investidores iniciantes a escolher uma melhor distribuição dos investimentos de seu portfólio.

Alguns tópicos que pretendemos estudar em trabalhos futuros são:

1. a possibilidade de escolher κ "ideal" e automatizar a coleta, processamento dos dados das ações e obtenção do portfólio ótimo para níveis de risco baixo, médio e alto de um portfólio dado;

2. melhorias na teoria de Markowitz como CAPM (SHARPE, 1964) e simulações de Monte Carlo (WANG, 2012) aplicadas a finanças;

3. melhorias no GP para acelerar a convergência como Gradiente Projetado Espectral (BIRGIN E. G., 2000) e Aceleração de Nesterov (NESTEROV., 1983).

REFERÊNCIAS

BARBETTA, P. A. et al. **Estátisca para cursos de engenharia e informática**. São Paulo: Atlas, 2010.

BERTSEKAS, D. P. **Nonlinear Programming**. Belmont: Athena Scientific, 1999.

BIRGIN E. G., et al. Nonmonotone Spectral Projected Gradient Methods on Convex Sets. **SIAM Journal on Optimization**, v. 10, n. 4, p. 1196–1211, 2000.

GONÇALVES, D. S. et al. A projected gradient method for optimization over density matrices. **Optimization Methods and Software**, v. 31, p. 328–341, 2016.

MARKOWITZ, H. Portfolio Selection. **The Journal of Finance**, v. 26, n. 1, p. 77–91, 1952.

MARTÍNEZ, J. M.; SANTOS, S. A. **Métodos Computacionais de Otimização**. Rio de Janeiro: IMPA/SBM, 1995.

MICHELOT, C. A finite algorithm for finding the projection of a point onto the canonical simplex of $\mathbb{R}^n$. **Journal of Optimization Theory and Applications**, v. 50, p. 195–200, 1986.

NESTEROV., Y. A method of solving a convex programming problem with convergence rate $O(\frac{1}{k^2})$. **Doklady Akademiia Nauk SSSR**, v. 27, n. 2, p. 372–376, 1983.

NOCEDAL, J.; WRIGHT, S. J. **Numerical Optimization**. Nova Iorque: Springer, 1999.

ROSS STEPHEN, A. et al. **Administração financeira**. [*S.l.*]: Grupo A, 2015.

SHARPE, W. F. Capital asset prices: a theory of market equilibrium under conditions of risk. **The Journal of Finance**, v. 19, n. 3, p. 425–442, 1964.

WANG, Hui. **Monte Carlo simulation with applications to finance**. [*S.l.*]: CRC Press, 2012.

APÊNDICE A – OBTENÇÃO E PRÉ-PROCESSAMENTO DE DADOS REAIS

Foi utilizado o *Yahoo! Finance* para a coleta de dados das ações indicadas. Primeiro foi feita pesquisa pelo código das ações no site como, por exemplo, mostra-se na figura a seguir.

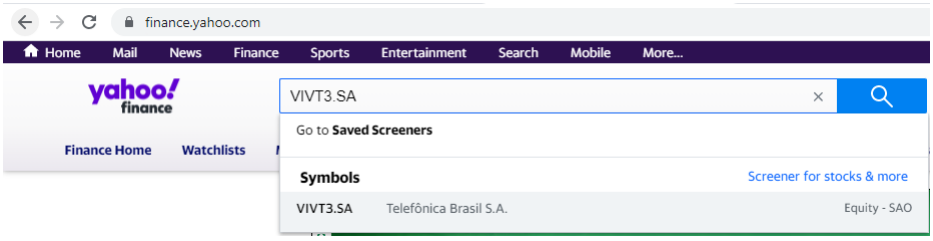

Figura 21 – Exemplo de busca de uma ação no site do Yahoo! Finance

Feito isso, vamos em *Historical Data* e no *Time Period* selecionamos a opção "5*Y*", que corresponde ao período de tempo de cinco anos, e selecionamos *Apply*. E para finalizar fizemos o *Download* dos dados em formato *csv*.

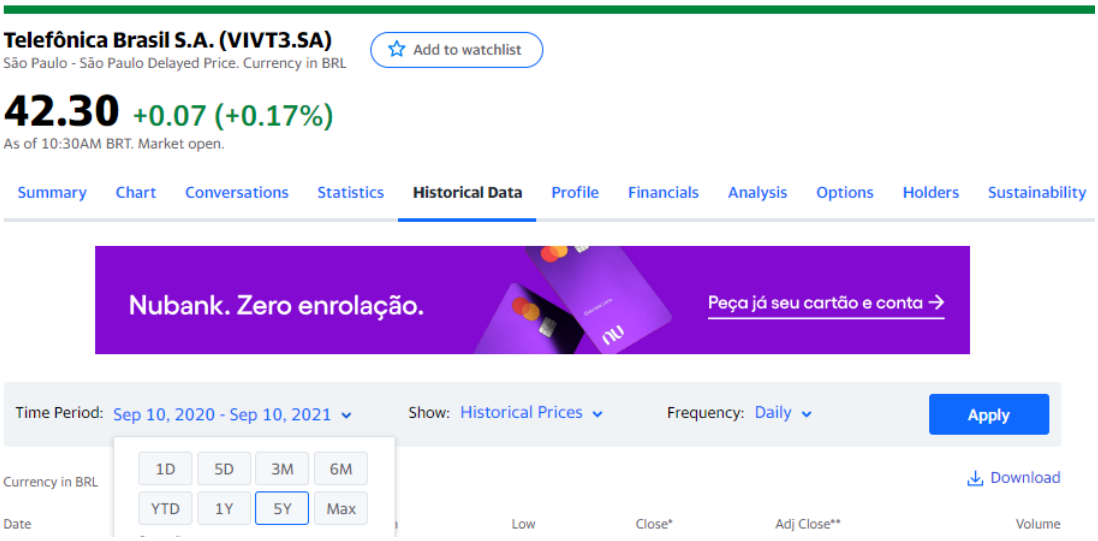

Figura 22 – Exemplo de coleta de dados históricos das ações

Para o tratamento dos dados, construção das estimativas de μ_i, σ_i, ρ_{ij}, bem como para a implementação dos algoritmos de Frank-Wolfe e Gradiente Projetado para resolução do problema (31) utilizamos o software *Octave* (códigos disponíveis em `https://drive.google.com/drive/folders/1NXOjm8v6xE-gHi4V1zzU4F7dYaFSUcWe?usp=sharing`).

No *Octave* podemos abrir o arquivo no editor de texto e você verá algo parecido com a figura a seguir.

Salvaremos este arquivo em uma matriz utilizando o código seguinte.

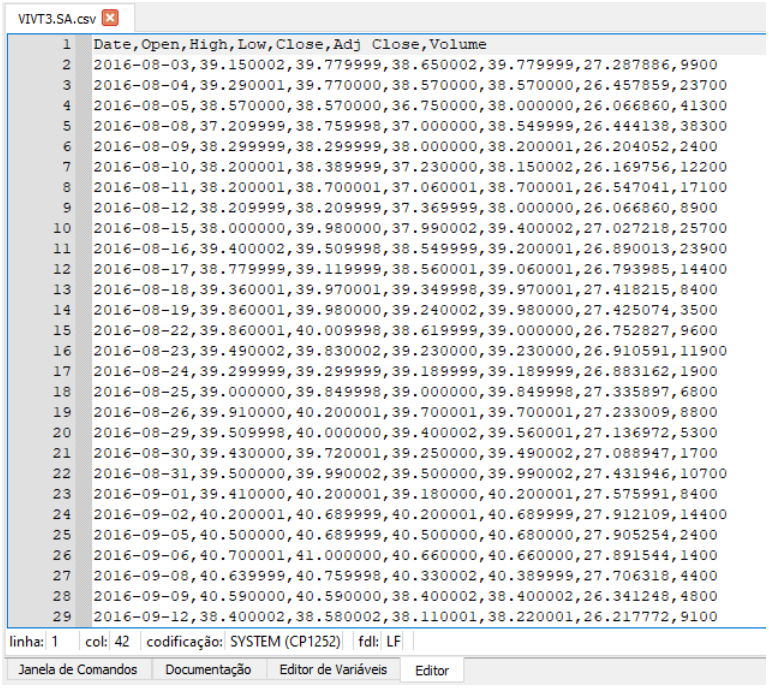

Figura 23 – Arquivo csv da ação VIVT3.SA no editor de texto do Octave.

```
>> A = csvread('VIVT3.SA.csv');
```

Figura 24 – Salvando arquivo csv da ação em matriz.

O arquivo, provavelmente, terá algum erro como apresentar o preço da ação em um determinado dia igual a zero, o que não é possível. Para “limparmos” esses dados foi desenvolvida a função “*precos.m*” para retirar esses zeros incorretos.

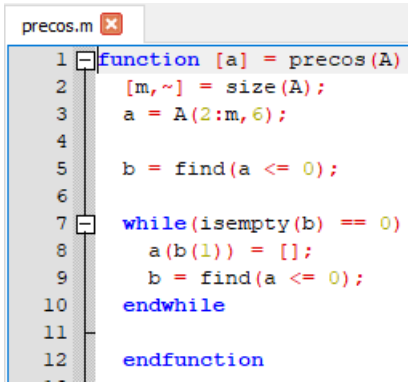

Figura 25 – Função *precos.m*

Estamos interessados na coluna “*Adj Close*”, veja Figura 23, que corresponde a sexta coluna da matriz *A* criada, Figura 24. Então o código analisa os números da sexta coluna da matriz A e caso alguma entrada seja igual a zero a entrada é retirada. E o código retorna um vetor, com nenhuma entrada nula, correspondentes aos preços

diários da ação. Com este vetor, utilizaremos a função *retornos.m* para calcular os retornos diários.

retornos.m

```
function [b] = retornos(a)
  n = length(a);
  b = zeros(n-1,1);

  for i = 1:n-1
    b(i) = (a(i + 1) - a(i))/a(i);
  endfor

  endfunction
```

Figura 26 – Função *retornos.m*

Então salvamos o vetor preços da ação VIVT3.SA em um vetor *a* e o vetor de retornos *aa*.

```
>> a = precos(A);
>> aa = retornos(a);
```

Figura 27 – Vetores de preços e de retornos

E dessa forma repetimos o processo para todas as outras ações. Tendo todos os vetores de retornos podemos calcular tanto o vetor μ de retornos esperados e a matriz *G* de covariância.

Para calcular o vetor μ cada entrada será igual a $248 \times md_A$ onde md_A corresponde a média dos retornos diários de uma ação *A* (das indicadas), que pode ser calculado pelo código *mean()*. Multiplicamos por 248 pois iremos anualizar os retornos (intuitivamente pensaríamos em multiplicar por 365, mas isto é incorreto já que as ações não são negociadas em todos os dias do ano). Obtemos o seguinte vetor.

$$\mu = (0.127120, 0.209334, 0.146460, 0.295909, 0.134735, 0.150473, 0.073672)^T . \tag{40}$$

Para calcular a matriz *G* primeiramente criaremos uma matriz auxiliar com as colunas correspondentes aos vetores de retornos diários das ações indicadas. Feito isso, utilizaremos o código *cov()* e *corr()* nesta matriz auxiliar e multiplicaremos por 248^2 o resultado do *cov()* (novamente multiplicaremos para anualizar os riscos e ficar coerente com o vetor μ, que também está anualizado), e estas são as matrizes *G* e *CORR*, respectivamente, como queríamos. Obtemos as seguintes matrizes.

```
G =

   0.086632   0.033705   0.022811   0.041212   0.033084   0.034601   0.023447
   0.033705   0.218631   0.041263   0.076657   0.056033   0.063715   0.046171
   0.022811   0.041263   0.057125   0.042257   0.033444   0.028971   0.021727
   0.041212   0.076657   0.042257   0.187278   0.058139   0.056374   0.043403
   0.033084   0.056033   0.033444   0.058139   0.087081   0.037045   0.028224
   0.034601   0.063715   0.028971   0.056374   0.037045   0.111035   0.034724
   0.023447   0.046171   0.021727   0.043403   0.028224   0.034724   0.097109
```

Figura 28 – Matriz de covariâncias *G*

```
CORR =

   1.0000   0.2449   0.3243   0.3235   0.3809   0.3528   0.2556
   0.2449   1.0000   0.3692   0.3788   0.4061   0.4089   0.3169
   0.3243   0.3692   1.0000   0.4085   0.4742   0.3638   0.2917
   0.3235   0.3788   0.4085   1.0000   0.4553   0.3909   0.3218
   0.3809   0.4061   0.4742   0.4553   1.0000   0.3767   0.3069
   0.3528   0.4089   0.3638   0.3909   0.3767   1.0000   0.3344
   0.2556   0.3169   0.2917   0.3218   0.3069   0.3344   1.0000
```

Figura 29 – Matriz de correlação *CORR*

Para responder as questões discutidas na Subseção 5.2.1, para os valores de κ discutidos na Seção 5.2, utilizamos a função "*fronteira.m*".

O vetor x_0 utilizado no teste foi

$$x_0 = \left(\frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}\right),$$

e os valores $\sigma = \varepsilon = 10^{-4}$. Obviamente, a *G* utilizada foi a matriz da figura 28 e o vetor μ da equação (40). Então, para as entradas do vetor "*eficiente*" iguais a zero temos que a solução do problema (31), para os valores de κ correspondentes a tais entradas, está na fronteira eficiente como discutido na Subseção 5.2.1.

fronteira.m

```
function [eficiente] = fronteira(x0,G,mu,sigma, epsilon)
  eficiente = []; n = length(mu); e = ones(n,1);
  for k = [1, 50, 100, 150, 200, 250, 300, 400, 600, 800, 1000, 1500, 2000, 3000]
    [xl, fmin,iteracao,avaliacoes,otimalidade,xks] = gradienteprojetado(x0,G,mu,k,sigma,epsilon);

    r = mu'*xl; gamma = 0.5*xl'*G*xl;

    A = [e'; mu']; b = [1; r];

    [x2, FVAL, EXITFLAG, OUTPUT, LAMBDA] = quadprog(G,[],[],[],A,b,zeros(7,1),[],x0);

    x3 = fmincon(@(x) -mu'*x, xl, [], [], e', 1, zeros(7,1), [],@(x) naolin(x,G,gamma));

    %comparando as soluções
    retorno = mu'*x2;
    risco = 0.5*x3'*G*x3;
    comparar = [r - retorno; risco - gamma];
    c = max(comparar);
    c = abs(c);

    if c > epsilon
      eficiente = [eficiente; 1]; %não está na fronteira eficiente
    else
      eficiente = [eficiente; 0]; %está na fronteira eficiente
    end
  end
```

Figura 30 – Função *fronteira.m*