

0.1 Introdução

Será considerado o problema usual de regressão linear com ruído

$$b = Ax + \epsilon, \quad (\text{P})$$

em que $b \in \mathbb{R}^n$ é a resposta, $\epsilon \in \mathbb{R}^n$ é o ruido, $A \in \mathbb{R}^{n \times p}$ é a matriz do modelo e $x \in \mathbb{R}^p$ é o vetor de coeficientes buscado. Com o aumento da dimensão dos dados, técnicas de aprendizado esparso ganharam enfoque recentemente pela compacticidade e interpretabilidade dos modelos obtidos Hastie et al. [2015], Bhlmann and van de Geer [2011].

Assumindo essa esparsidade na solução, algo comumente desejado ou teoricamente esperado no caso $p \gg n$ Hastie et al. [2015], Bhlmann and van de Geer [2011], uma estratégia natural é considerar o problema de quadrados mínimos regularizado

$$\min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \lambda_0 \|x\|_0 + \lambda_q \|x\|_q^q, \quad q \in \{1, 2\}. \quad (\text{R})$$

Aqui, a norma ℓ_0 refere-se à pseudo-norma definida como $\|x\|_0 = |\{i \mid x_i \neq 0, i = 1, \dots, p\}| \forall x \in \mathbb{R}^p$, $\lambda_0 \geq 0$ é seu parâmetro de penalização que regula o balanço entre elementos não nulos e recuperação da resposta, e $\lambda_q \geq 0$ controla a regularização ℓ_q .

Apesar da escolha $q = 1$ também induzir esparsidade, na realidade a regularização ℓ_q é necessária para evitar *overfitting* em cenários com baixa Relação Sinal-Ruído (RSR) Mazumder et al. [2017]. Esse efeito se dá pelo fenômeno de *shrinkage* induzido por essas regularizações. Denotaremos (R) com $q = 1$ de problema $\ell_0\ell_1$ e $q = 2$ de $\ell_0\ell_2$.

Alternativas de abordagem

O problema (R) é NP-difícil Natarajan [1995], tornando a solução computacional custosa. Três abordagens gerais são empregadas na prática. A primeira é utilizar *proxys* da norma ℓ_0 , como a norma ℓ_1 no modelo LASSO Tibshirani [1996], a *Minimax Concave Penalty* (MCP) Zhang [2010] e a *Smoothly Clipped Absolute Deviation* (SCAD) Fan and Li [2001]. Porém, em muitos regimes, estimadores obtidos de (R) sob parâmetros de penalização adequados exibem características estatísticas superiores (predição, estimativa e seleção de variáveis) comparado com essas alternativas menos computacionalmente desafiadoras (veja Hazimeh and Mazumder [2020a] e referências ali presentes).

A segunda alternativa são algoritmos que abordam a norma ℓ_0 diretamente e buscam soluções exatas. Trabalhos como Hazimeh et al. [2021] usam *Mixed Integer Programming* (MIP) para resolver em otimalidade global problemas com $p \sim 10^7$ em tempos de minutos a horas quando soluções altamente esparsas são desejadas.

A terceira é utilizar heurísticas ou algoritmos que solucionam aproximadamente (R) (ou uma formulação com restrição de cardinalidade) com $\lambda_q = 0$. Métodos populares incluem (*greedy*) *stepwise regression* Hastie et al. [2015], *Iterative Hard Thresholding* (IHT) Blumensath and Davies [2009], **abess** Zhu et al. [2020], e *greedy* e *randomized Coordinate Descent* (CD) Beck and Eldar [2013], Patrascu and Necoara [2015].

Em Hazimeh and Mazumder [2020a], é proposta uma abordagem que busca conciliar a velocidade de modelagens *proxys* dessa segunda estratégia e a otimalidade obtida pelos métodos de otimização inteira mista. O algoritmo proposto busca soluções quase ótimas que satisfaçam uma propriedade de exatidão combinatória local tal que pequenas perturbações no suporte não melhoram a função objetivo.

Os estimadores obtidos demonstraram superioridade em comparação com outros algoritmos de aprendizado esparso em quesitos como predição, estimativa e seleção de variáveis. Além disso, a implementação *open-source* dos autores, **fastselect**, provou-se significativamente mais rápida que implementações amplamente usadas como **ncvreg** Breheny and Huang [2011] e **glmnet** Friedman et al. [2010] que usam modelos *proxy*.

Notação

Para um conjunto de índices $T \subset \{1, 2, \dots, p\}$, $U_T \in \mathbb{R}^{p \times |T|}$ denota a submatriz composta pelas colunas da matriz identidade I_p correspondentes aos elementos de T . Além disso, $x_T \in \mathbb{R}^{|T|}$ denota o subvetor de $x \in \mathbb{R}^p$ composto pelos índices em T .

O conjunto das funções L -suaves, $\mathcal{C}_L^{1,1}$, contém todas as funções $\mathcal{C}^1 \ni g : \mathbb{R}^k \rightarrow \mathbb{R}$ (continuamente diferenciáveis) tais que ∇g é Lipschitz contínuo com fator L , isto é, $\|\nabla g(x) - \nabla g(y)\| \leq L\|x - y\| \forall x, y \in \mathbb{R}^k$.

Para uma matriz $B \in \mathbb{R}^{k \times k}$ com autovalores reais, $\lambda_{\max}(B)$ é o seu maior autovalor.

Denotamos por fim o conjunto $[p] := \{1, 2, \dots, p\}$.

0.2 Operador proximal e condições de otimalidade

Definindo $f(x) := \frac{1}{2}\|Ax - b\|_2^2 + \lambda_2\|x\|_2^2$ a parte suave de (R) e $h(x) := \lambda_0\|x\|_0 + \lambda_1\|x\|_1$ a parte não suave, é possível aplicar a vasta teoria de otimização compósita ao problema

$$\min_{x \in \mathbb{R}^p} F(x) := f(x) + h(x). \quad (\text{C})$$

Para isso, introduzimos o chamado operador proximal Moreau [1962] de $g : \mathbb{R}^p \rightarrow \mathbb{R}$ que, em sua formulação mais clássica, é definido como

$$\text{prox}_g(x) := \arg \min_{u \in \mathbb{R}^p} g(u) + \frac{1}{2}\|u - x\|_2^2. \quad (1)$$

Tomando $g = ch$, $c \geq 0$, o problema proximal acima é separável. Seja $h_i(u_i) := \lambda_0 \mathbf{1}_{\{0\}}(u_i) + \lambda_1 |u_i| + \frac{(u_i - x_i)^2}{2c} \forall i \in [p]$. Como no *soft-thresholding operator*, $\text{sign}(x_i)(|x_i| - c\lambda_1)$ é o minimizador não nulo de h_i quando $|x_i| > c\lambda_1$ Hastie et al. [2015]. Se $|x_i| - c\lambda_1 < \sqrt{2c\lambda_0}$, então $h_i(0) < h_i(\text{sign}(x_i)(|x_i| - c\lambda_1))$. Já se $|x_i| - c\lambda_1 = \sqrt{2c\lambda_0}$, tem-se $h_i(0) = h_i(\text{sign}(x_i)(|x_i| - c\lambda_1))$. Por fim, $h_i(0) > h_i(\text{sign}(x_i)(|x_i| - c\lambda_1))$ se $|x_i| - c\lambda_1 > \sqrt{2c\lambda_0}$. Dessa maneira, temos uma fórmula fechada para o operador proximal de ch ,

$$[\text{prox}_{ch}(x)]_i = \arg \min_{u_i \in \mathbb{R}} h_i(u_i) = \begin{cases} \{0\}, & |x_i| - c\lambda_1 < \sqrt{2c\lambda_0} \\ \{0, \text{sign}(x_i)(|x_i| - c\lambda_1)\}, & |x_i| - c\lambda_1 = \sqrt{2c\lambda_0} \\ \{\text{sign}(x_i)(|x_i| - c\lambda_1)\}, & |x_i| - c\lambda_1 > \sqrt{2c\lambda_0} \end{cases}.$$

Soluções estacionárias

Para $G : \mathbb{R}^p \rightarrow \mathbb{R}$ e $d \in \mathbb{R}^p$, definimos a derivada direcional (inferior)

$$G'(x, d) := \liminf_{\alpha \downarrow 0} \frac{G(x + \alpha d) - G(x)}{\alpha}.$$

Definição 1 (Solução estacionária) Um vetor $x^* \in \mathbb{R}^p$ é uma solução estacionária de (C) se, $\forall d \in \mathbb{R}^p$, $F'(x^*, d) \geq 0$.

Para $x \in \mathbb{R}^p$, definimos a i -ésima variável minimizadora da parte suave de F

$$\tilde{x}_i := \arg \min_{u_i \in \mathbb{R}} f(x + (u_i - x_i)e_i) = \frac{\left\langle b - \sum_{j \neq i} x_j A_j, A_i \right\rangle}{\|A_i\|_2^2 + 2\lambda_2} \quad \forall i \in [p], \quad (2)$$

em que A_k denota a k -ésima coluna de A .

Lema 1 Um vetor $x^* \in \mathbb{R}^p$ com suporte S é uma solução estacionária de (C) se, e somente se,

$$x_i^* = \text{sign}(\tilde{x}_i^*) \left(|\tilde{x}_i^*| - \frac{\lambda_1}{\|A_i\|_2^2 + 2\lambda_2} \right) \text{ e } |\tilde{x}_i^*| > \frac{\lambda_1}{\|A_i\|_2^2 + 2\lambda_2} \quad \forall i \in S. \quad (3)$$

Demonstração. [Hazimeh and Mazumder, 2020a, Lemma 1]. \square

Mínimo coordenada a coordenada

A seguinte classe de minimizadores é inspirada em pontos estacionários de algoritmos coordenados Hazimeh and Mazumder [2020a].

Definição 2 (Mínimo CW) Um ponto $x^* \in \mathbb{R}^p$ é um mínimo Coordinate-Wise (CW) de (C) se, $\forall i \in [p]$, x_i^* minimiza F com respeito à x_i mantendo as demais coordenadas fixas, ou seja,

$$x_i^* \in \arg \min_{u_i \in \mathbb{R}} F(x^* + (u_i - x_i^*)e_i) \quad \forall i \in [p].$$

Seja $x^* \in \mathbb{R}^p$ mínimo CW. É possível caracterizar mínimos CW a partir do operador proximal através da relação

$$\begin{aligned} x_i^* &\in \arg \min_{u_i \in \mathbb{R}} F(x^* + (u_i - x_i^*)e_i) \\ &= \arg \min_{u_i \in \mathbb{R}} f(x^* + (u_i - x_i^*)e_i) + h(x^* + (u_i - x_i^*)e_i) \\ &= \arg \min_{u_i \in \mathbb{R}} \frac{1}{2} \left\| b - \sum_{j \neq i} x_j^* A_j - u_i A_i \right\|_2^2 + \lambda_2 u_i^2 + \lambda_0 \mathbf{1}_{\{0\}}(u_i) + \lambda_1 |u_i| \quad (4) \\ &= \arg \min_{u_i \in \mathbb{R}} \frac{\|A_i\|_2^2 + 2\lambda_2}{2} (u_i - \tilde{x}_i^*)^2 + \lambda_0 \mathbf{1}_{\{0\}}(u_i) + \lambda_1 |u_i| \\ &= \text{prox}_{\frac{1}{\|A_i\|_2^2 + 2\lambda_2} (\lambda_0 \mathbf{1}_{\{0\}}(\cdot) + \lambda_1 |\cdot|)}(\tilde{x}_i^*) \quad \forall i \in [p], \end{aligned}$$

onde foi usado que a transladação e multiplicação por constantes não alteram o minimizador. Comparando (4) e (3), pelo Lema 1 x^* é solução estacionária.

Mínimo inescapável por troca

As próximas definições de otimalidade seguem de conceitos de análise combinatorial e refinam a ideia de mínimo CW. Dado um ponto mínimo CW x^* , podemos tentar melhorá-lo em valor objetivo performando uma operação de troca que consiste em desativar (definir como 0) algumas coordenadas do suporte de x^* e permitir que outras entrem no suporte. Após isso, é realizada uma otimização para o novo suporte com respeito as novas coordenadas inseridas (otimização parcial) ou todas as coordenadas (otimização total).

Definição 3 (Mínimo PSI) Seja $k \in \mathbb{N}^+$. Um ponto $x^* \in \mathbb{R}^p$ com suporte S é um mínimo Partial Swap Inescapable (PSI) de (C) de ordem k , denotado $\text{PSI}(k)$, se x^* é uma solução estacionária e $\forall S_1 \subset S, S_2 \subset S^c$ tais que $|S_1|, |S_2| \leq k$ vale

$$F(x^*) \leq \min_{u \in \mathbb{R}^{|S_2|}} F(x^* - U_{S_1}x_{S_1}^* + U_{S_2}u). \quad (5)$$

Seja $x^* \in \mathbb{R}^p$ mínimo $\text{PSI}(k)$ para algum $k \in \mathbb{N}^+$. De (5), $\forall i \in S$, tomado $S_1 = \{i\}$ e $S_2 = \emptyset$, tem-se

$$F(x^*) \leq F(x^* - U_{S_1}x_{S_1}^*) = F(x^* - x_i^*e_i).$$

Por (3) (estacionariedade de x^*) e a relação acima, tem-se então que x^* satisfaz (4). Assim, x^* é mínimo CW.

Definição 4 (Mínimo FSI) Seja $k \in \mathbb{N}^+$. Um ponto $x^* \in \mathbb{R}^p$ com suporte S é um mínimo Full Swap Inescapable (FSI) de (C) de ordem k , denotado $\text{FSI}(k)$, se $\forall S_1 \subset S, S_2 \subset S^c$ tais que $|S_1|, |S_2| \leq k$ vale

$$F(x^*) \leq \min_{u \in \mathbb{R}^{|(S \cup S_2) \setminus S_1|}} F(x^* - U_{S_1}x_{S_1}^* + U_{(S \cup S_2) \setminus S_1}u).$$

Seja $x^* \in \mathbb{R}^p$ mínimo $\text{FSI}(k)$ para algum $k \in \mathbb{N}^+$. Tomando $S_1 = S_2 = \emptyset$ na definição acima, temos

$$F(x^*) \leq \min_{u \in \mathbb{R}^{|S|}} F(x^* + U_S u) \leq \min_{u_i \in \mathbb{R}} F(x^* + (u_i - x_i^*)e_i) \quad \forall i \in [p],$$

logo x^* é mínimo CW, e, portanto, solução estacionária. Além disso, como $S_2 \subset (S \cup S_2) \setminus S_1$, x^* satisfaz (5). Dessa forma, x^* é $\text{PSI}(k)$.

Ponto estacionário

Os conceitos a seguir são padrão em análise variacional (consulte [Beck, 2017, Chapter 3] para mais detalhes). Seja $g : \mathbb{R}^p \rightarrow \mathbb{R}$ e $x \in \text{dom } g$ fixo. Então

$$\widehat{\partial}g(x) := \left\{ \eta \in \mathbb{R}^p \mid \liminf_{u \rightarrow x} \frac{g(u) - g(x) - \langle \eta, u - x \rangle}{\|u - x\|} \geq 0 \right\}$$

é denominado o subdiferencial regular (ou Fréchet) de h em x . Além disso, o conjunto $\partial g(x)$, definido pela relação

$$\eta \in \partial g(x) \iff \exists \{x^k\}, \{\eta^k\} \subset \mathbb{R}^p : x^k \rightarrow_g x, \eta^k \rightarrow \eta, \eta^k \in \widehat{\partial}g(x^k) \quad \forall k \in \mathbb{N},$$

é conhecido como o subdiferencial limite (ou de Mordukhovich) de g em x . Claramente, sempre vale que $\widehat{\partial}g(x) \subset \partial g(x)$ por construção.

Como trabalhamos com uma função descontínua, h , o subdiferencial de Mordukhovich se prova mais apropriado para a seguinte definição de estacionariedade. Como $f \in C^1$, vale a regra da soma [Mordukhovich, 2018, Proposition 1.30] $\partial F(x) = \nabla f(x) + \partial h(x)$.

Definição 5 (Ponto M-estacionário) Um ponto $x^* \in \mathbb{R}^p$ é um ponto M-estacionário de (C) se

$$0 \in \partial F(x^*) = \nabla f(x^*) + \partial h(x).$$

Note que

$$\begin{aligned} 0 \in \widehat{\partial}F(x^*) &\iff \liminf_{u \rightarrow x^*} \frac{F(u) - F(x^*)}{\|u - x^*\|} \geq 0 \\ &\iff \liminf_{\alpha \downarrow 0} \frac{F(x^* + \alpha d) - F(x^*)}{\alpha \|d\|} \geq 0 \quad \forall d \neq 0 \\ &\iff \liminf_{\alpha \downarrow 0} \frac{F(x^* + \alpha d) - F(x^*)}{\alpha} \geq 0 \quad \forall d \in \mathbb{R}^p. \end{aligned}$$

A última condição é exatamente a definição de x^* ser uma solução estacionária. Assim, como $\widehat{\partial}F(x) \subset \partial F(x)$, temos que soluções estacionárias são pontos M-estacionários.

Ponto estacionário do gradiente proximal

A seguinte definição de estacionariedade é baseada no passo do gradiente proximal.

Definição 6 (Ponto estacionário PG) Um ponto $x^* \in \mathbb{R}^p$ é um ponto estacionário Proximal Gradient (PG) de (C) se, $\forall 0 < \tau < \frac{1}{L_f}$, $f \in \mathcal{C}_{L_f}^{1,1}$,

$$x^* \in \text{prox}_{\tau h}(x^* - \tau \nabla f(x^*)).$$

Seja x^* estacionário PG de suporte S . Note que, $\forall 0 < \tau < \frac{1}{L_f}$,

$$\begin{aligned} x^* \in \text{prox}_{\tau h}(x^* - \tau \nabla f(x^*)) &\implies 0 \in \tau \widehat{\partial}h(x^*) + (x^* - (x^* - \tau \nabla f(x^*))) \\ &\iff 0 \in \tau \nabla f(x^*) + \tau \widehat{\partial}h(x^*) \\ &\iff 0 \in \nabla f(x^*) + \widehat{\partial}h(x^*) \\ &\iff 0 \in \widehat{\partial}F(x^*), \end{aligned}$$

onde na primeira implicação foi usada a regra de Fermat (condição necessária de primeira ordem). Dessa forma, x^* é solução estacionária.

Além disso, substituindo

$$\begin{aligned} [\nabla f(x^*)]_i &= \langle Ax^* - b, A_i \rangle + 2\lambda_2 x_i^* \\ &= \left\langle \sum_{j \neq i} x_j^* A_j - b, A_i \right\rangle + (\|A_i\|_2^2 + 2\lambda_2)x_i^* \stackrel{(2)}{=} (\|A_i\|_2^2 + 2\lambda_2)(x_i^* - \tilde{x}_i^*) \end{aligned}$$

e usando a separabilidade do PG, tem-se

$$\begin{aligned} x^* &\in \text{prox}_{\tau h}(x^* - \tau \nabla f(x^*)) \\ \iff x_i^* &\in \text{prox}_{\tau(\lambda_0 \mathbf{1}_{\{0\}}(\cdot) + \lambda_1 |\cdot|)}(x_i^* - \underbrace{\tau(\|A_i\|_2^2 + 2\lambda_2)(x_i^* - \tilde{x}_i^*)}_{:=y_i^*}) \quad \forall i \in [p] \\ \iff \begin{cases} x_i^* = \text{sign}(y_i^*)(|y_i^*| - \tau\lambda_1) \text{ e } |x_i^*| \geq \sqrt{2\tau\lambda_0}, & i \in S \\ |y_i^*| - \tau\lambda_1 \leq \sqrt{2\tau\lambda_0}, & i \notin S \end{cases} \end{aligned}$$

Se $i \in S$ e $y_i^* > 0$, tem-se

$$x_i^* = x_i^* - \tau(\|A_i\|_2^2 + 2\lambda_2)(x_i^* - \tilde{x}_i^*) - \tau\lambda_1 \iff x_i^* = \tilde{x}_i^* - \frac{\lambda_1}{\|A_i\|_2^2 + 2\lambda_2},$$

e como $\text{sign}(x_i^*) = \text{sign}(y_i^*)$ (*soft-thresholding* preserva sinal), devemos ter $\tilde{x}_i^* > 0$. Já se $y_i^* < 0$,

$$x_i^* = x_i^* - \tau(\|A_i\|_2^2 + 2\lambda_2)(x_i^* - \tilde{x}_i^*) + \tau\lambda_1 \iff x_i^* = \tilde{x}_i^* + \frac{\lambda_1}{\|A_i\|_2^2 + 2\lambda_2},$$

e pelo mesmo argumento anterior $\tilde{x}_i^* < 0$. Dessa forma

$$\begin{cases} x_i^* = \text{sign}(\tilde{x}_i^*) \left(|\tilde{x}_i^*| - \frac{\lambda_1}{\|A_i\|_2^2 + 2\lambda_2} \right) \text{ e } |x_i^*| \geq \sqrt{2\tau\lambda_0}, & i \in S \\ (\|A_i\|_2^2 + 2\lambda_2)|\tilde{x}_i^*| - \lambda_1 \leq \sqrt{\frac{2\lambda_0}{\tau}}, & i \notin S \end{cases}.$$

Comparando a relação acima com (4), conclui-se que, se $L_f \geq 1$, mínimos CW não estacionários PG.

0.3 Hierarquia das condições de otimalidade

Pelo que foi discutido na seção anterior, temos a seguinte hierarquia das condições de otimalidade:

$$\begin{aligned} \text{Mínimos FSI}(k) &\subset \text{Mínimos PSI}(k) \subset \text{Mínimos CW} \stackrel{L_f \geq 1}{\subset} \text{Estacionários PG} \\ &\subset \text{Soluções estacionárias} \subset \text{M-estacionários}. \end{aligned} \tag{6}$$

Temos que $f \in \mathcal{C}_{L_f}^{1,1}$ com $L_f = \lambda_{\max}(A^T A) + 2\lambda_2$, lembrando que $A^T A$ é simétrica e, portanto, dotada de autovalores reais. A condição $L_f \geq 1$ é garantida, por exemplo, se $\|A_i\|_2 = 1 \ \forall i \in [p]$. Nesse caso, $\lambda_{\max}(A^T A) \geq \max_{i \in [p]} e_i^T A^T A e_i = \max_{i \in [p]} (A^T A)_{ii} = \max_{i \in [p]} \|A_i\|_2^2 = 1$.

0.4 Algoritmos

Partially greedy cyclic CD

O algoritmo presente em Hazimeh and Mazumder [2020a] basicamente consiste em obter x^{k+1} minimizando com respeito somente a i -ésima coordenada via

$$x_i^{k+1} \in \arg \min_{u_i \in \mathbb{R}} F(x^k + (u_i - x_i^k)e_i) = \text{prox}_{\frac{1}{\|A_i\|_2^2 + 2\lambda_2}(\lambda_0 \mathbf{1}_{\{0\}}(\cdot) + \lambda_1 |\cdot|)}(x_i^k).$$

O algoritmo é iniciado com x^0 e calcula $r^0 = b - Ax^0$. Antes de executado, as coordenadas são reordenadas em ordem decrescente de $|\langle r^0, A_i \rangle| = |[\nabla f(x^0)]_i - 2\lambda_2 x_i^0|$. Na prática, é usada ordenação parcial, ou seja, somente as maiores t coordenadas nesse critério (com $t \ll p$) são ordenadas enquanto as demais mantêm a ordem original, e as atualizações então usam uma ordem cíclica. Isso é mais rápido e igualmente eficaz que a ordenação completa Hazimeh and Mazumder [2020a].

Por esse aspecto denominaremos o método de *Partially Greedy Cyclic Coordinate Descent* (PGCCD). A reordenação é feita uma única vez antes da execução, diferentemente do método *greedy* CD que seleciona a melhor coordenada para cada atualização. Outras alternativas de ordem de atualização podem ser empregadas, como randômica ou cíclica convencional, embora experimentos numéricos demonstrem a superioridade da estratégia em questão Hazimeh and Mazumder [2020a].

Os passos de Spacer, cuja inclusão tem o propósito de garantir convergência teórica, serão omitidos da nossa implementação. Sob hipóteses fracas, o artigo demonstrou convergência desse algoritmo para mínimos CW.

NPG

O algoritmo NPG (Algorithm 1) para solucionar problemas compósitos como (C) foi apresentado inicialmente em Kanzow and Mehlitz [2021]. A cada iteração, ele executa uma busca linear com condição de decréscimo GLL Grippo et al. [1986]. Para parâmetros $m > 1$, essa busca é não monótona, comparando o valor de função de um candidato com o maior valor de função dos últimos m iterados da sequência principal. A escolha do tamanho de passo inicial é livre com salvaguardas, o que fornece flexibilidade para implementações de passos obtidos na literatura (oriundos de problemas não compósitos ou até de minimização restrita), em especial aqueles com características espetrais Barzilai and Borwein [1988].

Algorithm 1 Nonmonotone Proximal Gradient method (NPG)

Input: $x^0 \in \mathbb{R}^p$, $\gamma_{\max} > \gamma_{\min} > 0$, $m \in \mathbb{N}$, $\delta \in (0, 1)$ and $\tau \in (0, 1)$

Output: Last x^{k+1} computed or x^{best}

- 1: Initialize $k := 0$
- 2: **repeat**
- 3: Choose $\gamma_{k,0} \in [\gamma_{\min}, \gamma_{\max}]$
- 4: Initialize $i := 0$
- 5: **while**

$$F(x^{k,i}) > \max_{0 \leq j \leq \min\{k, m-1\}} F(x^{k-j}) - \frac{\delta}{2\gamma_{k,i}} \|x^{k,i} - x^k\|^2 \quad (7)$$

where

$$x^{k,i} \in \text{prox}_{\gamma_{k,i} h}(x^k - \gamma_{k,i} \nabla f(x^k)) \quad (8)$$

do

- 6: $\gamma_{k,i+1} \leftarrow \tau \gamma_{k,i}$
 - 7: $i \leftarrow i + 1$
 - 8: **end while**
 - 9: $x^{k+1} \leftarrow x^{k,i}$
 - 10: $k \leftarrow k + 1$
 - 11: **until** A suitable termination criterion is violated at iteration k
-

Uma possível modificação simples e de amplo emprego em outros esquemas não monótonos é retornar

$$x^{best} \in \arg \min_{x^k \in \{x^k\}} F(x^k), \quad (9)$$

em que $\{x^k\}$ é a sequência gerada pelo NPG, ao invés do último iterado calculado.

A teoria de convergência presente em Kanzow and Mehlitz [2021], apesar de mais fraca que a de métodos semelhantes, garante convergência do Algorithm 1 para um ponto M-estacionário quando h é somente semicontínua inferior, como a nossa função de interesse.

NSPG

A primeira variante do Algorithm 1, o *Nonmonotone Spectral Proximal Gradient method* (NSPG), utiliza o passo espectral primal Barzilai and Borwein [1988], RAYDAN [1993]. Várias abordagens foram propostas para lidar com funções objetivo não convexas, nas quais o tamanho do passo espectral pode se tornar negativo. Para o tamanho de passo inicial em cada iteração é usada a proposta em Dai et al. [2015], que consiste em

$$\gamma_{k,0} \leftarrow \gamma_{k,+}^{BBR} := \begin{cases} \gamma_k^{BBR}, & \gamma_k^{BBR} > 0 \\ \frac{\|s^k\|}{\|r^k\|}, & \text{c.c.} \end{cases}, \quad (10)$$

em que γ_k^{BBR} é o passo espectral definido como

$$\gamma_k^{BBR} := \frac{(s^k)^T s^k}{(s^k)^T r^k},$$

sendo $s^k := x^k - x^{k-1}$ e $r^k := \nabla f(x^k) - \nabla f(x^{k-1})$.

O passo espectral pode ser entendido como um passo quasi-Newton em que H_k , a aproximação da Hessiana em x^k , é restrita a uma estrutura simplificada com somente uma variável de decisão, $H_k := \mu_k I$, $\mu_k \in \mathbb{R}$. A equação secante,

$$H_k s^k = r^k, \quad (11)$$

então se reduz à $\mu_k s^k = r^k$. Em geral, essa equação não possui solução exata, sendo então aceita a solução de quadrados mínimos, isso é,

$$\mu_k = \arg \min_{\mu \in \mathbb{R}} \| \mu s^k - r^k \|^2 = \frac{(s^k)^T r^k}{(s^k)^T s^k} = \frac{1}{\gamma_k^{BBR}}.$$

De fato, a interpretação acima aplicada em (8) para $i = 0$ implica

$$\begin{aligned} x^{k,0} &\in \arg \min_{u \in \mathbb{R}^p} \left\{ h(u) + \langle \nabla f(x^k), u - x^k \rangle + \frac{1}{2\gamma_k^{BBR}} \|u - x^k\|^2 \right\} \\ &= \arg \min_{u \in \mathbb{R}^p} \left\{ \langle \nabla f(x^k), u - x^k \rangle + \frac{\mu_k}{2} (u - x^k)^T I (u - x^k) + h(u) \right\} \\ &= \arg \min_{u \in \mathbb{R}^p} \left\{ \langle \nabla f(x^k), u - x^k \rangle + \frac{1}{2} (u - x^k)^T H_k (u - x^k) + h(u) \right\} \\ &= \arg \min_{u \in \mathbb{R}^p} \left\{ f(x^k) + \langle \nabla f(x^k), u - x^k \rangle + \frac{1}{2} (u - x^k)^T H_k (u - x^k) + h(u) \right\}, \end{aligned}$$

ou seja, o esquema proximal com o passo espectral consiste em minimizar uma aproximação de segunda ordem de f em x^k (termos à esquerda na última equação) e h original, o que é essencialmente uma aproximação da função objetivo F completa.

Em experimentos realizados em Barbosa and Silva [2024], foi averiguado que o NSPG superou outros métodos proximais no problema (R) com $\lambda_q = 0$ tanto na velocidade de convergência quanto na qualidade das soluções obtidas em valor objetivo. Esses resultados motivaram a tentativa de empregá-lo juntamente com técnicas de otimização local combinatorial Hazimeh and Mazumder [2020a].

NSPGH

Outra forma de escrever a equação secante (11) (também denominada forma primal) é na sua forma dual $B_k r^k = s^k$, na qual B_k agora aproxima localmente a inversa da Hessiana. Tomando $B_k = \mu_k I$ resulta agora no tamanho de passo espectral dual

$$\gamma_k^{BBR2} := \mu_k = \arg \min_{\mu \in \mathbb{R}} \| \mu r^k - s^k \|^2 = \frac{(s^k)^T r^k}{(r^k)^T r^k}.$$

Pela desigualdade de Cauchy-Schartz, $\gamma_k^{BBR} \geq \gamma_k^{BBR2}$. Analogamente,

$$\gamma_{k,+}^{BBR2} := \begin{cases} \gamma_k^{BBR2}, & \gamma_k^{BBR2} > 0 \\ \frac{\|r^k\|}{\|s^k\|}, & \text{c.c.} \end{cases}. \quad (12)$$

Os trabalhos Goldstein et al. [2014], Zhou et al. [2006] mostraram que o desempenho de algoritmos proximais pode ser melhorado usando uma escolha híbrida entre esses dois tamanhos de passo,

$$\gamma_{k,+}^{BBRH} = \begin{cases} \gamma_{k,+}^{BBR2}, & \gamma_{k,+}^{BBR} < \delta\gamma_{k,+}^{BBR2} \\ \gamma_{k,+}^{BBR} - \frac{1}{\delta}\gamma_{k,+}^{BBR2}, & \text{c.c.} \end{cases} \quad (13)$$

aqui, o hiperparâmetro $\delta \in \mathbb{R}$ é tipicamente escolhido como 2, como será na nossa implementação.

O Algorithm 1 tal que $\gamma_{k,0} \leftarrow \gamma_{k,+}^{BBRH}$ será chamado de NSPH, em que “H” representa *hybrid*.

VMNSPG

Com o intuito de permitir mais informação de segunda ordem, em Park et al. [2020] utiliza um operador proximal com métrica matricial. Seja $g : \mathbb{R}^p \rightarrow \mathbb{R}$, $U \in S_{++}^p$ (simétrica positiva definida) e $\|y\|_U := \sqrt{y^T U y}$ a norma U . O operador proximal com métrica variável é definido como

$$\text{prox}_{g,U}(x) := \arg \min_{u \in \mathbb{R}^k} g(u) + \frac{1}{2} \|u - x\|_U^2. \quad (14)$$

No geral, esse subproblema proximal é de difícil solução devido ao acoplamento de variáveis causado pela multiplicação por U . Por esse motivo, a utilização desse prox é geralmente restrita a matrizes U diagonais e funções g separáveis, garantindo, assim, separabilidade de (14).

Também nesse artigo, é introduzido um esquema semelhante ao Algorithm 1 que utiliza a atualização

$$x^{k,i} \in \text{prox}_{h,U_{k,i}}(x^k - U_{k,i}^{-1} \nabla f(x^k)) \quad (15)$$

em lugar de (8) e o critério

$$F(x^{k,i}) > \max_{0 \leq j \leq \min\{k,m-1\}} F(x^{k-j}) - \frac{\delta}{2} \|x^{k,i} - x^k\|_{U_{k,i}}^2$$

em lugar de (7), em que analogamente $U_{k,i} = \tau^i U_{k,0}$ e $U_{k,0}$ é diagonal positiva definida. A teoria de convergência presente é restrita à h convexa, f L -suave e fortemente convexa e parâmetro de não monotonicidade $m = 1$. Contudo, é possível que a teoria de Kanzow and Mehlitz [2021] seja aplicável para relaxar essas hipóteses.

Seja U_{k-1} a métrica que gerou o iterado x^{k-1} por (15). Para melhor capturar a geometria Hessiana de f , os autores de Park et al. [2020] propõe a métrica diagonal dada pelo problema

$$\begin{aligned} \min_{U \in \mathbb{R}^p} & \|Us^k - y^k\|_2^2 + \mu\|U - U_{k-1}\|_F^2 \\ \text{s.a. } & \frac{1}{\gamma_{k,+}^{BBR}} I \preceq U \preceq \frac{1}{\gamma_{k,+}^{BBR2}} I, U = \text{Diag}(u). \end{aligned} \quad (16)$$

Aqui, o hiperparâmetro $\mu > 0$ controla o *trade-off* entre a satisfação da equação secante (11) e a proximidade com a métrica anterior U_{k-1} . Um μ grande é usado

se é esperado que a aproximação do Hessiano não mude muito entre iterações. Já um μ serve como salvaguarda e evita operações indefinidas. Os elementos diagonais são limitados pelos passos espetrais primal e dual, conferindo garantia de não negatividade ou passos muito grandes em uma coordenada.

O problema (16) possui solução em forma fechada dada por

$$(U_{k,+}^{DBBR})_{ii} := \max \left\{ \frac{1}{\gamma_{k,+}^{BBR}}, \min \left\{ \frac{1}{\gamma_{k,+}^{BBR2}}, \frac{(s_i^k)^2 + \mu(U_{k-1})_{ii}^2}{(s_i^k)^2 + \mu} \right\} \right\} \quad \forall i \in [p].$$

Essa versão do Algorithm 1 em que $U_{k,0} \leftarrow U_{k,+}^{DBBR}$ será denominada *Variable Metric Nonmonotone Spectral Proximal Gradient method* (VMNSPG).

Otimização local combinatorial

Um algoritmo iterativo para encontrar um mínimo PSI(k) através de perturbações no suporte é apresentado em Hazimeh and Mazumder [2020a]. Em cada iteração ℓ , é executado algum dos algoritmos anteriores para obter um mínimo CW ou ponto M-estacionário x^ℓ . Esse algoritmo será chamado de interior. Após isso é buscado um movimento descendente para o problema combinatório

$$\begin{aligned} \min_{u, S_1, S_2} & F(x^\ell - U_{S_1}x^\ell + U_{S_2}u) \\ \text{s.a } & S_1 \subseteq S, S_2 \subseteq S^c, |S_1|, |S_2| \leq k, u \in \mathbb{R}^{|S_2|} \end{aligned} \tag{17}$$

onde S é o suporte de x^ℓ . Se existir solução viável \hat{x} para o problema acima com $F(\hat{x}) < F(x^\ell)$, então \hat{x} pode não ser um mínimo CW/ponto M-estacionário. Neste caso, reinicializa-se o algoritmo escolhido com \hat{x} . Caso contrário, e ademais x^ℓ é solução estacionária (por exemplo, mínimo CW, gerado pelo PGCCD), então ele é PSI(o). O algoritmo a seguir resume o procedimento.

Algorithm 2 Composite Partial Swap Inescapable of order o method (CPSI(o))

Input: $\hat{x}^0 \in \mathbb{R}^p$, $o \in \mathbb{N}^+$ and an inner algorithm

Output: Last $\hat{x}^{\ell+1}$ computed (which is a PSI(o) minimum)

```

1: Initialize  $\ell := 0$ 
2: while true do
3:    $\hat{x}^{\ell+1} \leftarrow$  output of inner algorithm initialized with  $\hat{x}^\ell$ 
4:   if problem (17) has feasible solution  $\hat{x}$  with  $F(\hat{x}) < F(\hat{x}^{\ell+1})$  then
5:      $\hat{x}^{\ell+1} \leftarrow \hat{x}$ 
6:      $\ell \leftarrow \ell + 1$ 
7:   else
8:     break
9:   end if
10: end while

```

Para o caso $k = 1$, o seguinte algoritmo busca uma solução factível melhorada do problema (17).

Algorithm 3 Search for improved solution of problem (17) with $k = 1$

Input: x^ℓ
Output: \hat{x} (improved solution) if found, otherwise x^ℓ

- 1: $S \leftarrow \text{Supp}(x^\ell)$
- 2: **for** $i \in S$ **do**
- 3: **for** $j \in S^c$ **do**
- 4: $v_j^* \leftarrow \arg \min_{v_j \in \mathbb{R}} F(x^\ell - x_i^\ell e_i + v_j e_j)$ (18)
- 5: $F_j^* \leftarrow F(x^\ell - x_i^\ell e_i + v_j^* e_j)$
- 6: **end for**
- 7: $\vartheta \leftarrow \arg \min_{j \in S^c} F_j^*$
- 8: **if** $F_\vartheta^* < F(x^\ell)$ **then**
- 9: $\hat{x} \leftarrow x^\ell - x_i^\ell e_i + v_\vartheta^* e_\vartheta$
- 10: **break**
- 11: **end if**
- 12: **end for**

A solução de (18) é dada por

$$v_j^* \in \text{prox}_{\frac{1}{\|A_i\|_2^2 + 2\lambda_2}(\lambda_0 \mathbf{1}_{\{0\}}(\cdot) + \lambda_1 |\cdot|)} \left(x_{ij}^\ell \right), \quad (19)$$

em que x_{ij}^ℓ é a j -ésima variável minimizadora da parte suave de F a partir de $x^\ell - x_i^\ell e_i$

$$x_{ij}^\ell := \arg \min_{u_j \in \mathbb{R}} f(x^\ell - x_i^\ell e_i + u_j e_j) = \frac{\langle b - \sum_{l \neq i} x_l^\ell A_l, A_j \rangle}{\|A_j\|_2^2 + 2\lambda_2} = \frac{\langle r^\ell + x_i^\ell A_i, A_j \rangle}{\|A_j\|_2^2 + 2\lambda_2}.$$

Além disso, valem as seguintes equivalências

$$\begin{aligned} \arg \min_{j \in S^c} F_j^* &\iff \arg \max_{j \in S^c} |v_j^*| \\ F_\vartheta^* < F(x^\ell) &\iff |v_\vartheta^*| > |x_i^\ell|. \end{aligned}$$

Essas propriedades dão origem a seguinte versão eficiente do Algorithm 3.

Algorithm 4 Efficient search for improved solution of problem (17) with $k = 1$

Input: x^ℓ
Output: \hat{x} (improved solution) if found, otherwise x^ℓ

- 1: $S \leftarrow \text{Supp}(x^\ell)$
- 2: **for** $i \in S$ **do**
- 3: **for** $j \in S^c$ **do**
- 4: Compute v_j^* using (19)
- 5: **end for**
- 6: $\vartheta \leftarrow \arg \max_{j \in S^c} |v_j^*|$
- 7: **if** $|v_\vartheta^*| > |x_i^\ell|$ **then**
- 8: $\hat{x} \leftarrow x^\ell - e_i x_i^\ell + e_\vartheta v_\vartheta^*$
- 9: **break**
- 10: **end if**
- 11: **end for**

0.5 Experimentos numéricos

Nessa seção serão efetuados experimentos com o intuito de comparar os algoritmos apresentados.

Setup experimental

Geração de dados

Assim como em Hazimeh and Mazumder [2020a], utilizamos conjuntos de dados sintéticos para uma ampla variedade de tamanhos e configurações de problemas. A matriz de dados é gerada com distribuição gaussiana multivariada $A_{n \times p} \sim \text{MVN}(0, \Sigma)$, e então suas colunas são normalizadas tal que $\|A_i\|_2 = 1 \forall i$ para simplificar as atualizações dos algoritmos. Como verdade base, é utilizado um vetor de coeficientes esparsos $x^\dagger \in \mathbb{R}^p$ com k^\dagger entradas não nulas igualmente espaçadas e com valor 1. É escolhido também um valor aproximado para o SNR, definido como

$$\text{SNR} := \frac{\mathbb{E}[\|Ax^\dagger\|_2^2]}{\mathbb{E}[\|b - Ax^\dagger\|_2^2]},$$

e com base nessa escolha é tomado

$$\sigma = \sqrt{\frac{\|Ax^\dagger\|_2^2}{n \cdot \text{SNR}}}.$$

Em seguida, o vetor de resposta é gerado como $b = Ax^\dagger + \epsilon$, onde $\epsilon_i \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$ é independente de A . Dessa forma, garantimos que

$$\frac{\|Ax^\dagger\|_2^2}{\|b - Ax^\dagger\|_2^2} = \frac{\|Ax^\dagger\|_2^2}{\|\epsilon\|_2^2} \approx \frac{\|Ax^\dagger\|_2^2}{n\sigma^2} = \text{SNR},$$

ou seja, o SNR verdadeiro seja próximo do SNR desejado.

Foram consideradas as seguintes instâncias de $\Sigma := ((\sigma_{ij}))$:

- *Correlação constante:* É definido $\sigma_{ij} = \rho \forall i \neq j$ e $\sigma_{ii} = 1 \forall i$;

- *Correlação exponencial:* É definido $\sigma_{ij} = \rho^{|i-j|}$ $\forall i, j$, com a convenção $0^0 = 1$.

Parâmetros dos algoritmos

Após experimentos iniciais de ajuste de hiperparâmetros, foram escolhidos $m = 15$, $\delta = 0.01$ e $\tau = 0.25$ para o NPG. As salvaguardas γ_{min} e γ_{max} são usualmente tomadas como valores arbitrariamente pequenos e grandes, respectivamente. Os tamanhos de passo espectral iniciais, $\gamma_{0,+}^{BBR}$ e $\gamma_{0,+}^{BBR2}$, foram tomados via (10) e (12) com a convenção de $x^{-1} = x^0 - 10^{-5}\nabla f(x^0)$. Já a métrica diagonal espectral inicial, $U_{0,+}^{DBBR}$, foi definida com esses tamanhos de passo e a convenção de $U_{-1,+}^{DBBR}$ nula. Para o VMNSPG, $\mu = 10^{-3}$ apresentou os melhores resultados dentre os valores testados. Quanto ao PGCCD, somente foram ordenadas um quarto das coordenadas.

Comparação entre PGCCD e NPG

Consideramos a eficiência dos algoritmos propostos para o problema (R) com $\lambda_q = 0$ (problema ℓ_0 puro). Foi utilizado um conjunto de dados com correlação exponencial, $\rho = 0.5$, $n = 500$, $p = 2000$, $k^\dagger = 100$, SNR = 10. Foi tomado $\lambda_0 = 0.5 \frac{\|A^T b\|_\infty^2}{2L_f}$ para obter uma penalização razoável (veja [Barbosa and Silva, 2024, Exemplo 3] para uma explicação dessa escolha). Foram geradas 50 inicializações com suporte de tamanho k^\dagger com índices uniformemente escolhidos de $[p]$ e valores com distribuição uniforme em $(0, 1)$. Para cada inicialização é rodado cada algoritmo. A medida de recuperação do suporte é definida como

$$\frac{|Supp(x) \cap Supp(x^\dagger)|}{\max\{|Supp(x)|, k^\dagger\}},$$

que quantifica a fração de variáveis verdadeiramente ativas recuperadas, normalizada pelo máximo entre o tamanho do suporte estimado e o verdadeiro k^\dagger . Essa medida penaliza tanto falsos positivos quanto falsos negativos e valores próximos de 1 indicam recuperação alta. O critério de parada utilizada em todos os algoritmos foi uma mudança relativa no objetivo menor que 10^{-7} .

A Subfigura 0.5 contém o *box plot* do número de iterações, em que uma passagem pelas p coordenadas corresponde a uma iteração do PGCCD. Essa métrica não reflete precisamente a velocidade de cada algoritmo uma vez que não considera, por exemplo, o número de sub iterações na busca linear do NPG e a reordenação efetuada no PGCCD. Mesmo assim, é possível ter uma ideia de que a convergência do PGCCD é significativamente mais rápida que a dos métodos proximais não monótonos apresentados. O VMSPG demonstra convergência mais rápida entre os métodos NPG, provavelmente devido à sua incorporação de mais informação de segunda ordem. Seu número de iterações, com mediana em torno de 30, fica o mais próximo do PGCCD, com mediana em torno de 20. O passo espectral híbrido do NSPGH parece acelerar levemente a convergência em comparação com o passo primal usado no NSPG, e ambos performam cerca de duas a três vezes pior que o PGCCD.

Contudo, uma análise da qualidade das soluções obtidas através da Seção 0.5 e Subfigura 0.5 demonstra que, apesar de mais lentos, ambos o NSPG e NSPGH

obtém melhores resultados em valor objetivo e recuperação de suporte. Em particular nesse segundo aspecto, a diferença é bastante significativa. O VMNSPG obteve a pior desempenho em ambos os quesitos, apesar da sua convergência mais rápida.

A capacidade do NPG com estratégias de passo espectral encontrar soluções de qualidade está diretamente ligada a não monotonia da sequência de iterados e a permissão de tamanhos de passo maiores que permitem maior liberdade de movimento Barbosa and Silva [2024]. No caso do VMNSPG, é possível que uma convergência local mais rápida restrinja essa movimentação e a busca por soluções melhores. A excelente recuperação do suporte indica que o NSPG e NSPGH produzem candidatos mais adequados para a aplicação da otimização combinatorial local, como será visto nos experimentos a seguir. Esse resultado é curioso visto que esses algoritmos só possuem garantia de convergência a pontos M-estacionários, enquanto o PGCCD busca mínimos CW, cuja definição é mais restritiva (veja (6)).

Para melhor analisar as características dos limites das sequências dos algoritmos NPG, também foi executado o PGCCD a partir desses pontos para determinar se eles também eram mínimos CW. Consideramos que caso o suporte não seja alterado durante essa passagem pelo PGCCD e o número de iterações seja pequeno (2 ou menos) antes do critério de parada, então o ponto pode razoavelmente ser considerado (numericamente) CW. Para o NSPG, 60% das inicializações resultaram em pontos CW, enquanto essa porcentagem foi 44% para o NSPGH. O VMNSPG não resultou em nenhum ponto CW seguindo essa definição.

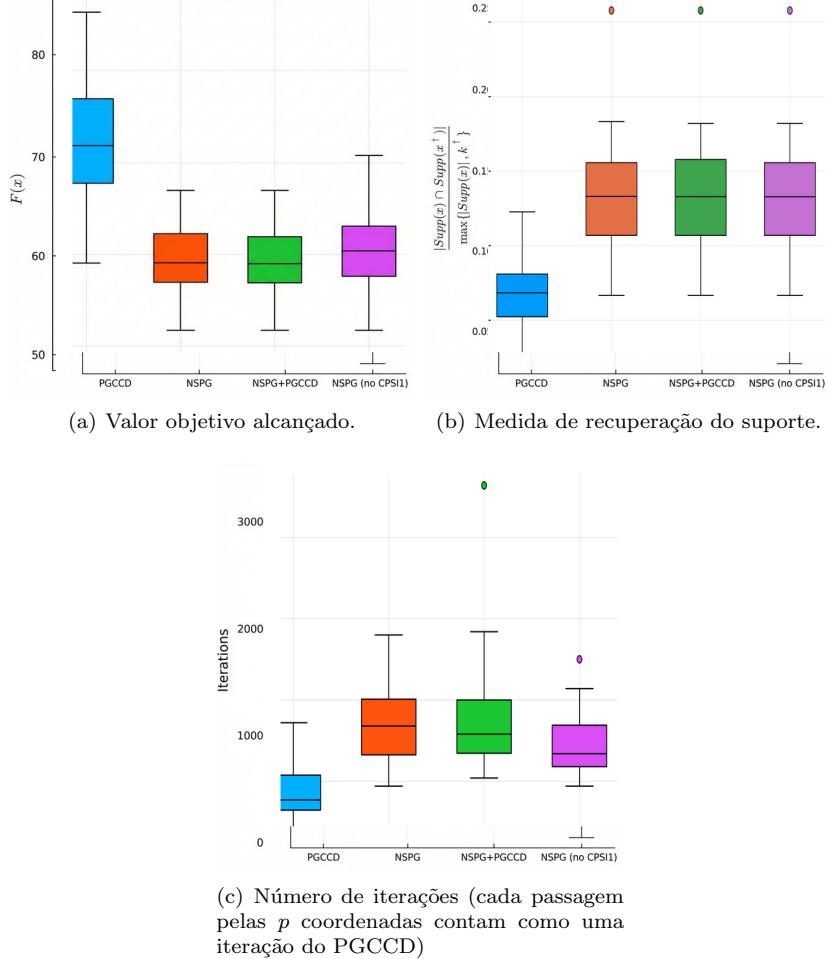


Figura 1: *Box plots* dos resultados dos algoritmos testados para 50 iniciizações. Correlação exponencial, $\rho = 0.5$, $n = 500$, $p = 2000$, $k^\dagger = 100$, SNR = 10.

Comparação com otimização combinatorial

Nesse experimento, visamos identificar o efeito que cada algoritmo tem quando empregado como *inner solver* do algoritmo CPSI(1) (Algorithm 2). Serão testados o PGCCD, o NSPG e o NSPG seguido do PGCCD. Como comparação da qualidade das soluções, também foi incluído o NSPG sem estratégia combinatorial.

O setup experimental é idêntico ao da seção anterior. Além das medidas apresentadas ali, incluímos o número de iterações externas (iterações do Algorithm 2). O número de iterações total se refere ao somatório de iterações de todas as chamadas do *inner solver* durante a execução.

A Subfigure 0.5 mostra que a diferença de iterações entre o PGCCD e o NSPG se tornou menos pronunciada que na Subfigure 0.5. Isso se deve ao fato

de que o PGCCD executa mais iterações externas em média, ou seja, precisa de mais perturbações para melhorar suas soluções, como visto na Subfigure 0.5. A otimização combinatorial causa um grande aumento no número de iterações do NSPG.

A qualidade das soluções obtidas por métodos NSPG (incluindo sem otimização combinatorial) continua bastante superior àquelas do PGCCD nos critérios analisados (Seção 0.5 e Subfigure 0.5). De fato, a otimização combinatorial melhora as soluções do NSPG, mas o ganho é relativamente pequeno nesse experimento.

A utilização conjunta do NSPG e PGCCD se mostrou vantajosa em todos os aspectos se comparado com o NSPG puro.

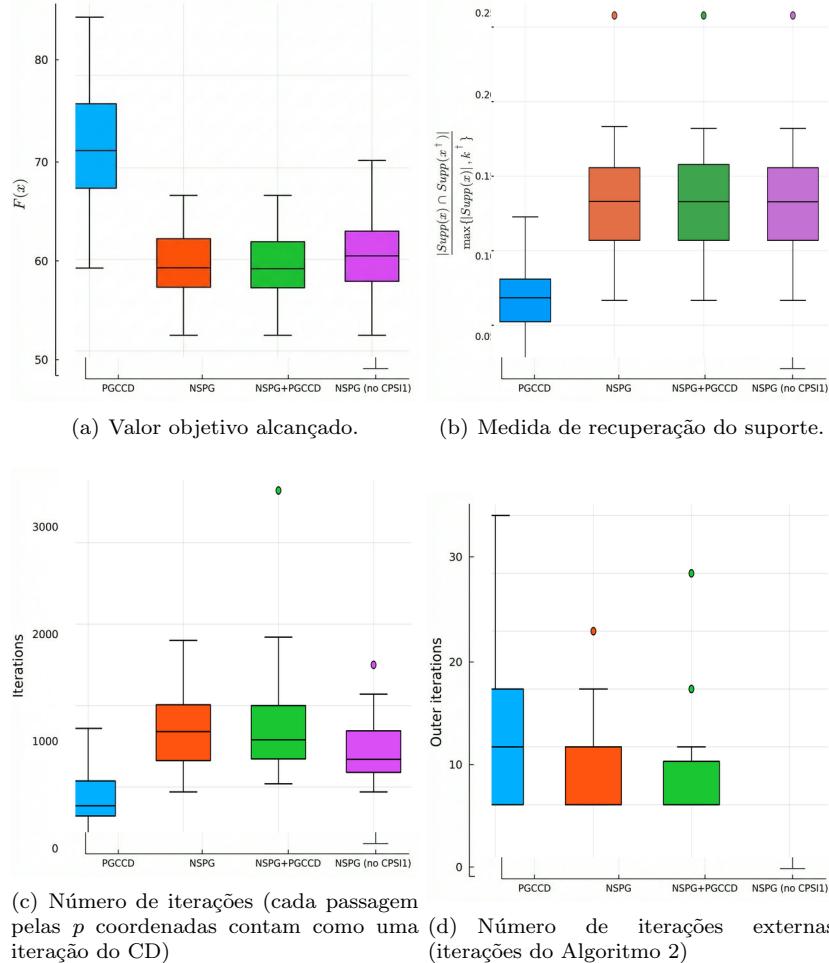


Figura 2: *Box plots* dos resultados dos algoritmos testados para 50 iniciizações. Correlação exponencial, $\rho = 0.5$, $n = 500$, $p = 2000$, $k^\dagger = 100$, SNR = 10.

Aumentando o SNR para 300 para permitir recuperação completa do su-

porte, vemos uma diferença drástica na qualidade das soluções, com somente um aumento pequeno no número de iterações (Figura 3). A estratégia NSPG+PGCCD continua sendo a mais atrativa, com menos soluções *outliers* com um custo levemente superior em iterações, porém todas as estratégias NSPG apresentam recuperação perfeita da solução verdadeira com probabilidade alta.

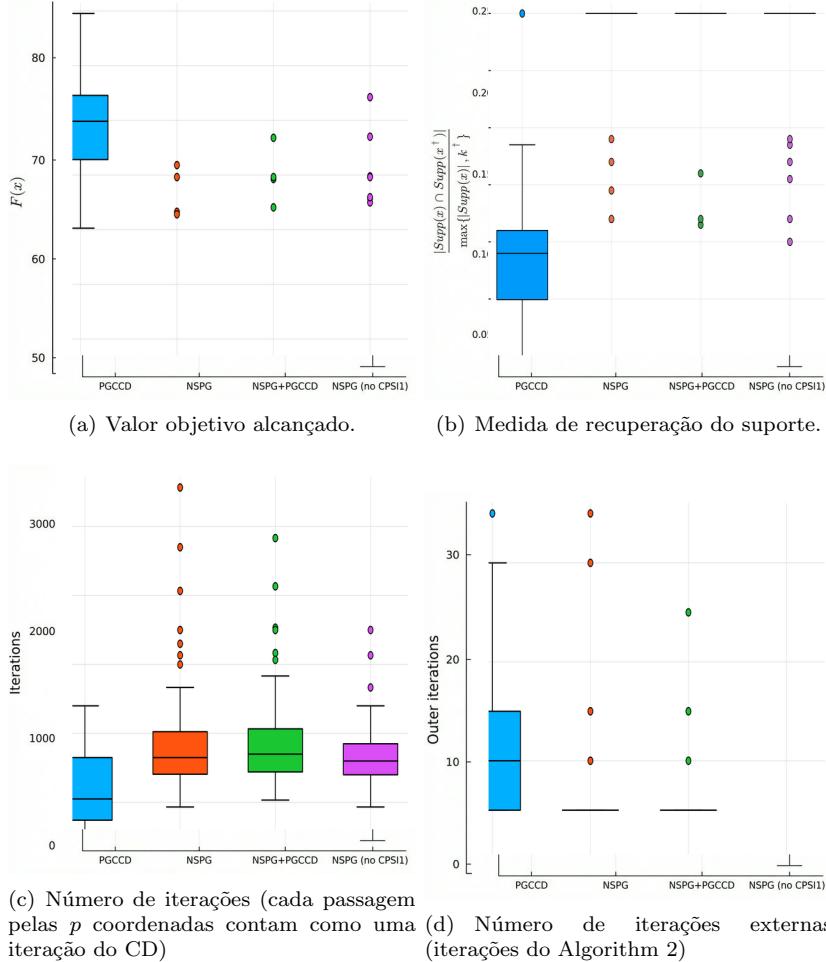


Figura 3: *Box plots* dos resultados dos algoritmos testados para 50 iniciizações. Correlação exponencial, $\rho = 0.5$, $n = 500$, $p = 2000$, $k^\dagger = 100$, SNR = 300.

Para o próximo experimento, mantivemos os parâmetros anteriores e trocando o tipo de correlação por constante, um cenário significativamente mais difícil. A Figura 4 apresenta resultados bastante drásticos em favor de estratégias NSPG com busca combinatorial. Em comparação com o CD, o ganho em iterações (tanto internas quanto externas) é substancial. Já em comparação com o NSPG sem busca combinatorial, a qualidade das soluções é muito superior. Entre o NSPG puro e o NSPG+PGCCD, esse primeiro se provou quase

tão bom em qualidade de soluções, mas com mais iterações externas, as quais são a parte mais expressiva no tempo de execução total.

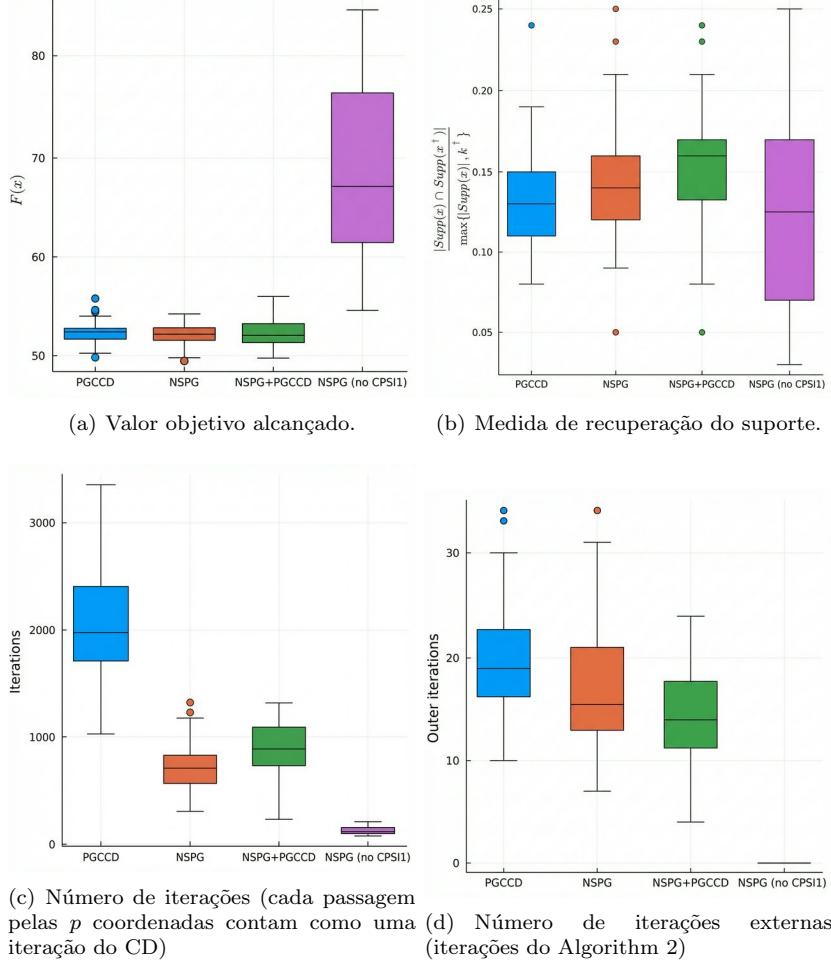


Figura 4: *Box plots* dos resultados dos algoritmos testados para 50 inicializações. Correlação constante, $\rho = 0.5$, $n = 500$, $p = 2000$, $k^\dagger = 100$, SNR = 300.

Por fim, com correlação exponencial, dividimos todas as dimensões por 2, tomamos $k^\dagger = 25$, e aumentamos a correlação para $\rho = 0.9$ (Figura 5). Todos os métodos com otimização combinatorial obtiveram 100% de recuperação do suporte original, enquanto o NSPG sem essa estratégia não. O número de iterações (internas e externas) do PGCCD foi muito mais alto, indicando novamente que as soluções obtidas desse método exigem mais perturbações do suporte antes de estabilizar.

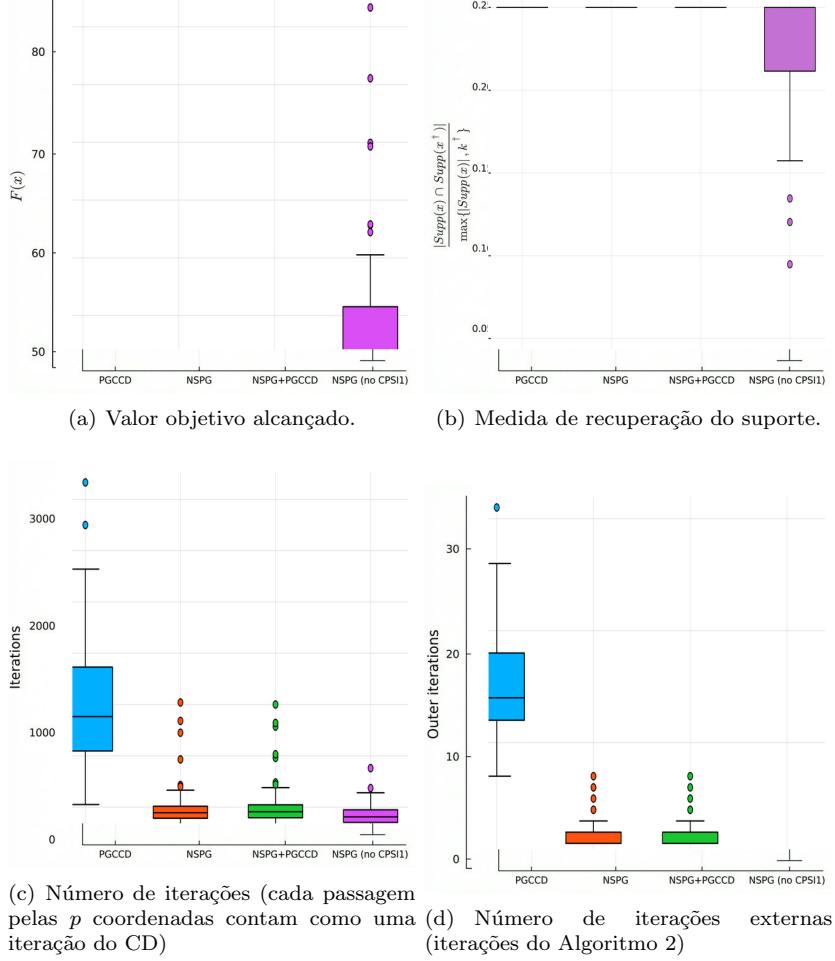


Figura 5: *Box plots* dos resultados dos algoritmos testados para 50 inicializações. Correlação exponencial, $\rho = 0.9$, $n = 250$, $p = 1000$, $k^\dagger = 25$, SNR = 300.

Estratégias de Validação Cruzada

Para a seleção do parâmetro de regularização λ , implementamos e comparamos três estratégias de validação cruzada (CV).

Para métodos inicializados com passo espectral (como o NSPG), o cálculo de λ em todas as estratégias foi ajustado para compensar a escala introduzida pelo passo $\gamma_{j,0}$. Para métodos que não utilizam o passo espectral inicial (como o PGCCD), assume-se $\gamma_{j,0} = 1$.

Adicionalmente, implementamos uma etapa de *Refinamento Final* pós-CV. Independentemente da estratégia de validação cruzada utilizada, o algoritmo retém o melhor λ^* encontrado. Em seguida, compara-se o erro de validação (MSE) de duas soluções finais geradas com este λ^* :

- **Candidato do Caminho (*Path-based*)**: A solução obtida seguindo o caminho de regularização (seja decrescente, crescente ou adaptativo).
- **Candidato do Zero (*Zero-based*)**: Uma solução obtida reinicializando o solver a partir do vetor nulo (*cold-start*) diretamente em λ^* .

A solução final escolhida é aquela que apresenta o menor MSE de validação. Esta etapa é crucial pois o *warm-start* pode, por vezes, prender o algoritmo em ótimos locais sub-ótimos herdados de λ anteriores, enquanto o *cold-start* pode encontrar uma bacia de atração melhor para o mesmo parâmetro de regularização.

O impacto deste refinamento é analisado detalhadamente no Apêndice C. As métricas apresentadas incluem: (i) *Wins/Ties/Losses*: Classificação qualitativa da escolha pelo zero em relação à recuperação do suporte, calculada apenas quando o candidato do zero é escolhido (por menor erro de validação). *Wins* indica que $J_{zero} > J_{path}$, *Ties* que $J_{zero} = J_{path}$, e *Losses* que $J_{zero} < J_{path}$; e (ii) *Sim Improv (Similarity Improvement)*: Diferença média na similaridade do suporte ($J_{zero} - J_{path}$) nestes mesmos casos.

Validação Cruzada Padrão (Decrescente)

Esta estratégia segue a abordagem clássica de caminhos de regularização, similar ao proposto em Hazimeh and Mazumder [2020a]. O algoritmo inicia com um valor de λ_0 suficientemente grande para garantir que a solução inicial seja o vetor nulo (ou esparsa). Em seguida, λ é reduzido monotonicamente. A re-inicialização (*warm-start*) é utilizada, onde a solução otimizada para λ_j serve como ponto de partida para a iteração $j + 1$ com $\lambda_{j+1} < \lambda_j$. A atualização de λ é dada por:

$$\lambda_{j+1} = 0.9 \cdot \min \left(\lambda_j, \gamma_{j,0} \cdot \frac{\max_{l \in S^c} |\nabla f(x^{j,0})_l|^2}{2} \right),$$

onde $\gamma_{j,0}$ é o tamanho de passo inicial da iteração j e S^c é o complemento do suporte de $x^{j,0}$. Esta fórmula busca o próximo valor que permitiria a entrada de uma nova variável no suporte. Para o valor inicial, utilizamos $\lambda_0 = 1.01 \cdot \gamma_{0,0} \cdot \frac{\max_l |\nabla f(x^0)_l|^2}{2}$, garantindo que o ponto de partida seja o vetor nulo.

Validação Cruzada Inversa (Crescente)

Inspirada pela necessidade de explorar regiões de soluções mais densas primeiramente, esta estratégia percorre o caminho na direção oposta. O processo inicia com um $\lambda_0 = 1.01^{-1} \cdot \gamma_{0,0} \cdot \frac{\min_l |\nabla f(x^0)_l|^2}{2}$ pequeno para gerar um suporte relativamente cheio. Para garantir um ponto de partida válido com suporte não nulo, o que ainda pode acontecer com correlação constante usando PGCCD, executa-se uma fase de *warmup* onde λ é decrescido geometricamente por 0.9 até que $\|x^{j,1}\|_0 > 0$. A partir deste ponto, λ é incrementado monotonicamente:

$$\lambda_{j+1} = 0.9^{-1} \cdot \max \left(\lambda_j, \frac{\min_{l \in S} |x_l^{j,0} - \gamma_{j,0} \nabla f(x^{j,0})_l|^2}{2\gamma_{j,0}} \right).$$

Esta atualização visa encontrar o próximo valor que forçaria a remoção de uma variável do suporte. Esta estratégia tende a ser mais rápida em alguns

cenários por iniciar já com o suporte correto ou próximo dele, evitando o custo de adicionar variáveis uma a uma.

Validação Cruzada Adaptativa Inteligente (*Smart Adaptive*)

Esta estratégia híbrida visa combinar a robustez da busca decrescente com a eficiência da busca crescente, mitigando o risco de convergir para mínimos locais ruins dependendo do ponto de partida. O procedimento consiste em duas fases:

1. **Sondagem (*Probe Phase*):** Avalia-se o erro de validação (MSE) nos dois extremos possíveis do caminho (λ_{low} da estratégia inversa e λ_{high} da padrão). Notavelmente, o cálculo de λ_{low} também emprega a fase de *warmup* descrita na estratégia inversa, garantindo que a sondagem inferior inicie de um suporte não nulo válido. A direção inicial de varredura (crescente ou decrescente) é escolhida baseada no extremo que apresentar menor erro de validação.
2. **Varredura com Reversão:** Inicia-se a varredura na direção escolhida. A cada passo, o algoritmo verifica se uma mudança de direção resultaria em uma redução significativa no erro de validação (melhora > 1%). Se tal melhora for detectada, o algoritmo permite uma única inversão de direção. Isso possibilita encontrar vales de mínimos locais que uma varredura monotônica rígida não conseguiria explorar em detalhes e, em certos casos, uma parada mais rápida do algoritmo.

O critério de parada da estratégia adaptativa inclui: (i) alcançar o limite de λ oposto ao de partida (definido por uma razão fixa do extremo não escolhido); (ii) detecção de oscilação (dupla inversão de direção); ou (iii) colapso do suporte para vazio (no caso de direção crescente) ou cheio (decrescente).

A escolha da estratégia de validação cruzada para cada algoritmo foi baseada nos resultados apresentados na Figura 6 e nas Figuras 1–3 do apêndice. A primeira linha de cada figura apresenta a medida de recuperação do suporte (Jaccard modificado¹), enquanto a segunda linha apresenta o tempo médio de execução por valor de λ testado.

Observa-se que métodos baseados em NSSP (incluindo PGCCD) beneficiam-se significativamente de caminhos de regularização bem definidos. Para o NSPG, observa-se que a estratégia inversa é altamente competitiva e robusta. Já para o PGCCD, a estratégia padrão (*Regular*) mostrou-se a mais estável para garantir boa recuperação em todos os cenários, enquanto a abordagem adaptativa (*Smart Adaptive*) mostrou-se ideal para a combinação NSPG+PGCCD, equilibrando eficiência e evasão de mínimos locais inadequados. Assim, os resultados reportados nas seções anteriores utilizam CV Inversa para NSPG, CV Adaptativa para NSPG+PGCCD e CV Padrão para PGCCD.

¹Utilizamos uma variação do índice de Jaccard dada por $J(S, S^\dagger) = |S \cap S^\dagger| / \max(|S|, |S^\dagger|)$, onde S e S^\dagger são os suportes estimado e verdadeiro, respectivamente. Diferente do Jaccard padrão, esta medida penaliza desbalanços de cardinalidade com base no maior conjunto.

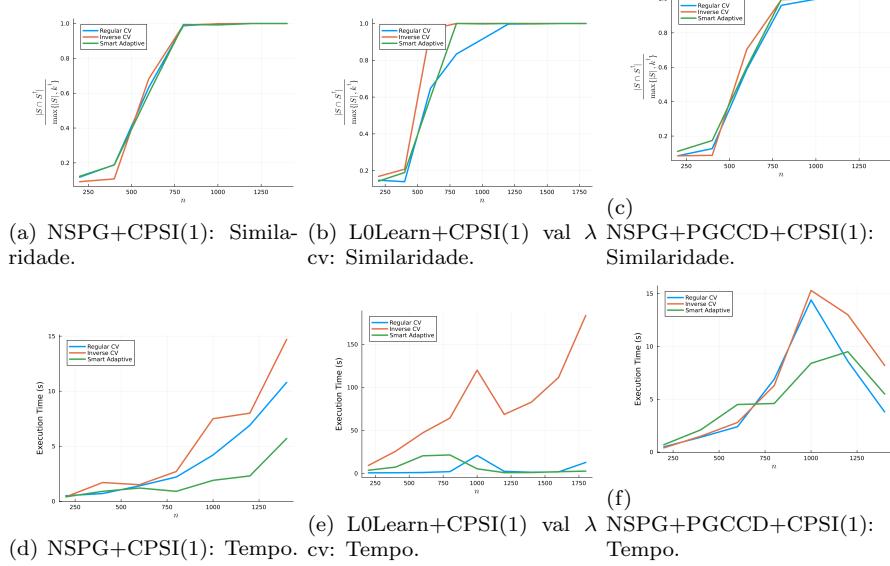


Figura 6: Comparação das estratégias de validação cruzada para NSPG, PGCCD e NSPG+PGCCD (Correlação exponencial, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10).

Comparação com L0Learn

Para contextualizar o desempenho do PGCCD em relação ao estado da arte, realizamos uma comparação com o pacote **L0Learn** Hazimeh and Mazumder [2020b] (versão de desenvolvimento do repositório git <https://github.com/fbkns/L0Learn>). O **L0Learn** é altamente otimizado e implementa algoritmos de descida coordenada com estratégias sofisticadas de *active set*.

Nesta comparação, distinguimos três variantes: (i) **L0Learn+CPSI1 val**: utiliza o algoritmo nativo CDPSI da versão Git; (ii) **L0Learn val (CD only)**: utiliza o algoritmo nativo CD; (iii) **L0Learn+CPSI1 val path**: utiliza o motor CD do **L0Learn** como base em um *wrapper Julia* que permite chamadas para um par específico (λ, x) , aplicando nossa própria estratégia de validação cruzada padrão. Cabe destacar que as versões nativas do **L0Learn** no repositório Git operam computando o caminho de regularização completo para um *grid* de λ 's, apesar de o artigo original sugerir uma abordagem de caminho flexível. Assim, para as variantes (i) e (ii), recebemos o conjunto completo de soluções gerado pelo pacote e selecionamos a melhor baseada no erro do conjunto de validação. Já na variante (iii), o solver é invocado iterativamente seguindo nossa lógica de busca, garantindo consistência com o processo de refinamento local.

A Figura 7 apresenta os resultados para o cenário de correlação exponencial com $\rho = 0.5$ e $p = 2000$. Neste cenário, todos os métodos baseados em NSSP atingem recuperação quase perfeita do suporte a partir de $n \approx 800$. A combinação NSPG+PGCCD+CPSI1 atinge similaridade 1.0 ligeiramente antes dos demais, porém ao custo de um tempo de execução significativamente superior, crescendo de forma acentuada com n (atingindo ~ 245 s para $n = 2000$) — enquanto os demais permanecem abaixo de 15s. Esse cenário ilustra

o principal compromisso da abordagem combinada: a otimização coordenada do PGCCD melhora a qualidade do suporte, mas a um custo computacional que escala supralinearmente. Quanto ao erro L_∞ nos parâmetros, os métodos NSPG+CPSI1 e NSPG+PGCCD+CPSI1 apresentam os menores valores em todo o intervalo de n , indicando que a melhor recuperação do suporte traduz-se diretamente em melhor estimativa dos parâmetros. O método L0Learn val (CD only) é consistentemente o pior tanto em similaridade quanto em L_∞ .

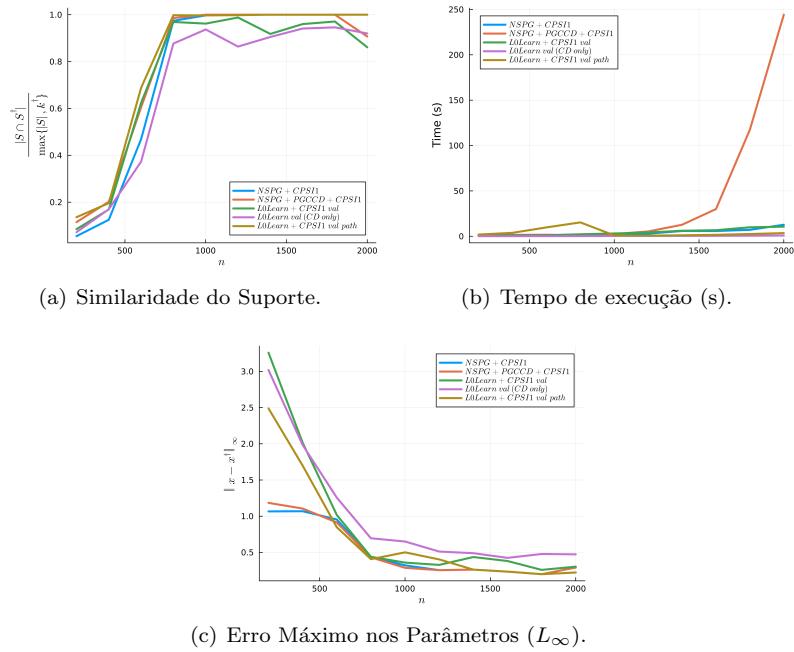


Figura 7: Comparaçao entre os métodos propostos e variantes do L0Learn. Correlação exponencial, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10.

A vantagem do PGCCD torna-se mais pronunciada nos cenários de correlação constante, como ilustrado na Figura 8. Neste cenário com $\rho = 0.9$ e $p = 1000$, a alta correlação dificulta a seleção de variáveis para todos os métodos, porém tanto o NSPG+CPSI1 quanto o NSPG+PGCCD+CPSI1 apresentam superioridade consistente. O NSPG+PGCCD+CPSI1 atinge similaridade ~ 0.95 já em $n \approx 250$, enquanto as variantes nativas do L0Learn necessitam de $n \geq 400$ para atingir desempenho comparável. O L0Learn val (CD only), em particular, apresenta degradação significativa neste cenário de alta correlação. Em termos de erro L_∞ , os métodos NSSP mantêm vantagem em toda a faixa de n , convergindo para ~ 0.1 a partir de $n \approx 800$.

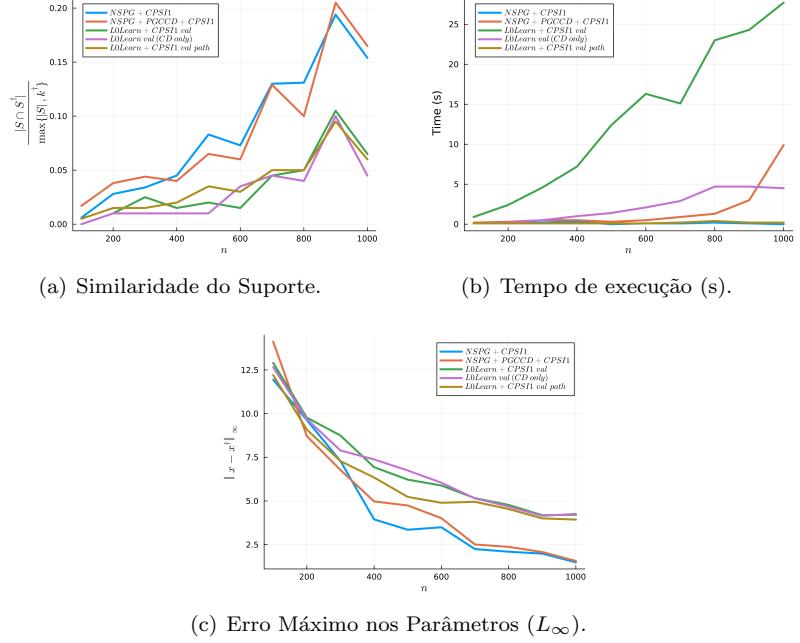


Figura 8: Comparação entre os métodos propostos e variantes do L0Learn. Correlação constante, $\rho = 0.9$, $p = 1000$, $k^{\dagger} = 20$, SNR = 5.

Essa diferença substancial nos cenários de correlação constante ocorre porque a descida coordenada do PGCCD é particularmente eficaz quando as correlações entre variáveis são uniformes, permitindo uma exploração mais eficiente do espaço de soluções. O mesmo padrão é observado na Figura 1 ($\rho = 0.5$, $p = 2000$), onde o NSPG+PGCCD+CPSI1 alcança similaridade próxima a 1.0 já em $n \approx 400$ com custo temporal moderado (~22s para $n = 2000$). No cenário de correlação exponencial com $\rho = 0.9$ (Figura 3), as diferenças entre os métodos são menores e a convergência é mais uniforme, sugerindo que a estrutura exponencial decrescente das correlações facilita a separação das variáveis relevantes.

Em síntese, os resultados indicam que: (i) a combinação NSPG+PGCCD oferece a recuperação de suporte mais robusta, especialmente em cenários de correlação constante, ao custo de maior tempo de execução; (ii) o NSPG isolado é altamente competitivo em cenários de correlação exponencial, onde alcança excelente recuperação com custo computacional reduzido; (iii) a variante L0Learn+CPSI1 val path, que aplica nosso procedimento de validação cruzada ao motor do L0Learn, apresenta desempenho significativamente superior às variantes nativas, evidenciando a importância da estratégia de busca; e (iv) o erro L_{∞} confirma consistentemente que a qualidade da recuperação do suporte se traduz em melhor estimativa dos parâmetros verdadeiros.

Apêndice A

Comparação adicional de estratégias de validação cruzada

Neste apêndice, apresentamos comparações adicionais das estratégias de validação cruzada para diferentes cenários de correlação e dimensionalidade, complementando a discussão da Figura 6.

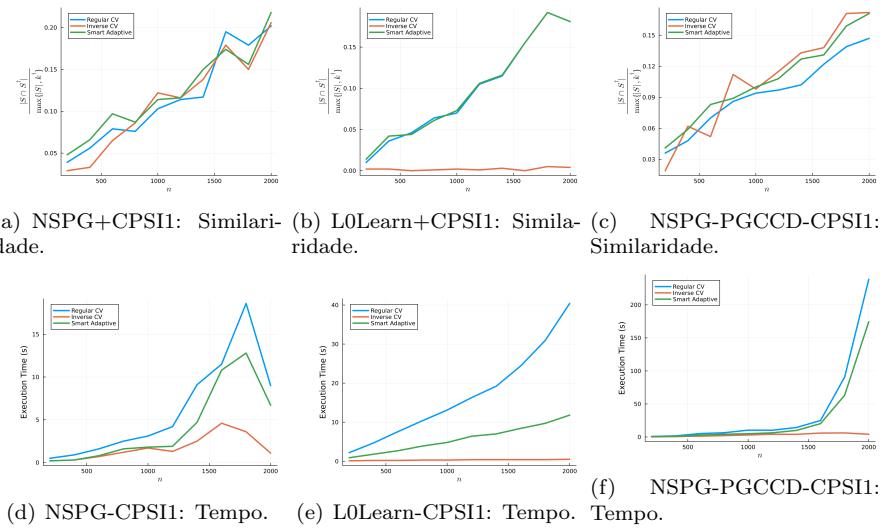


Figura 1: Comparação das estratégias de validação cruzada (Correlação constante, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10).

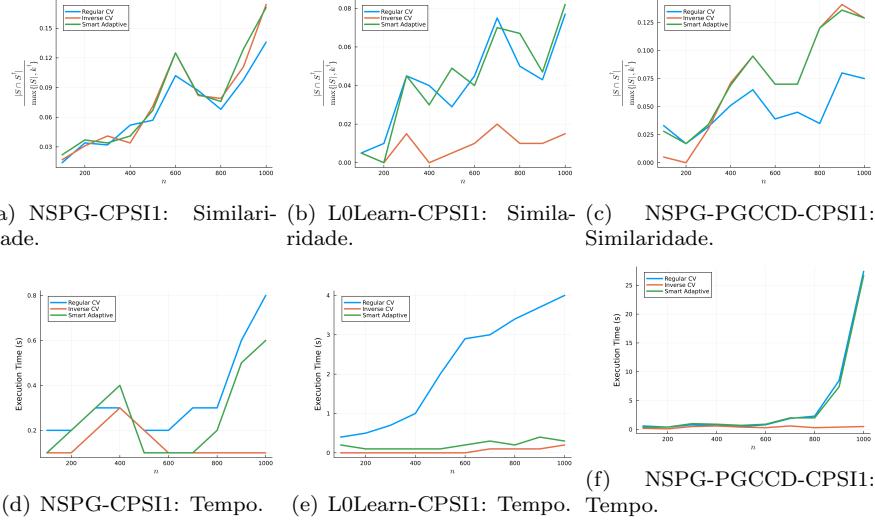


Figura 2: Comparação das estratégias de validação cruzada (Correlação constante, $\rho = 0.9$, $p = 1000$, $k^\dagger = 20$, SNR = 5).

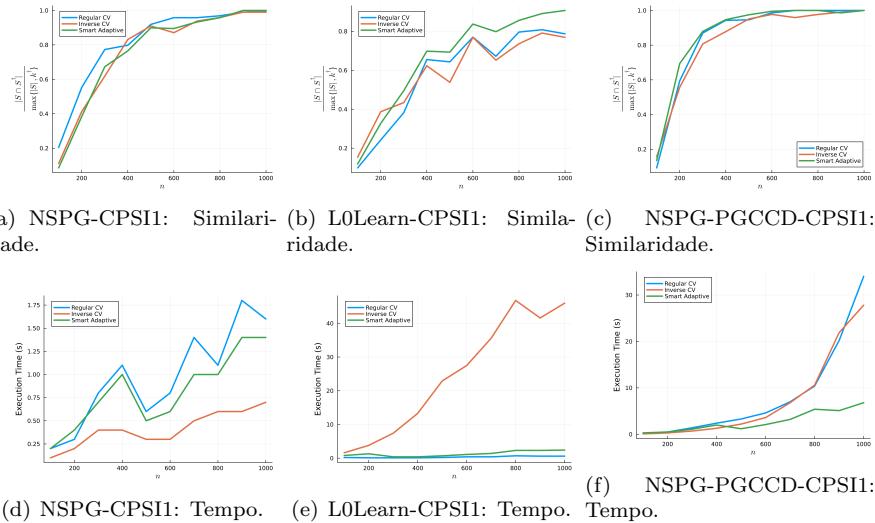


Figura 3: Comparação das estratégias de validação cruzada (Correlação exponencial, $\rho = 0.9$, $p = 1000$, $k^\dagger = 20$, SNR = 5).

Apêndice B

Comparação adicional com L0Learn

Neste apêndice, apresentamos comparações adicionais entre o PGCCD e o L0Learn para diferentes cenários.

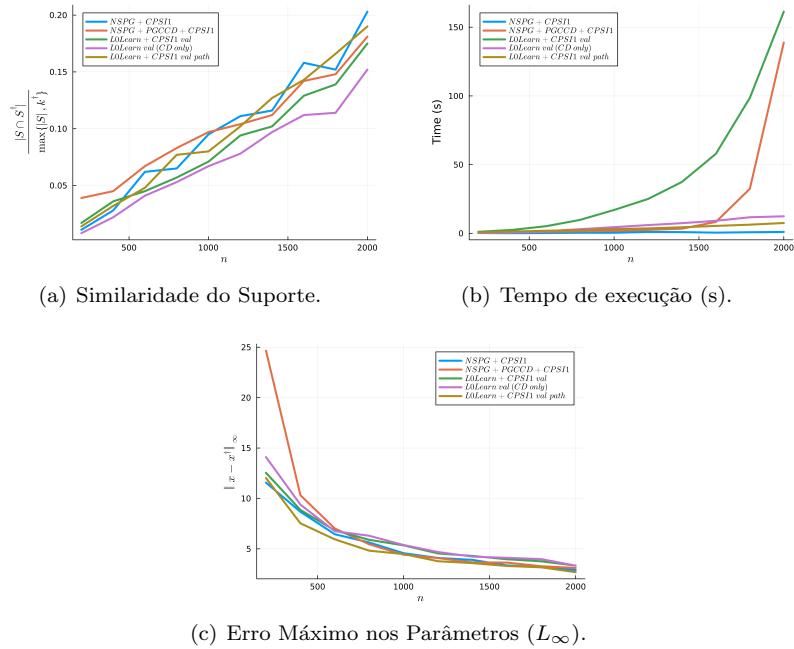


Figura 1: Comparação entre os métodos propostos e variantes do L0Learn. Correlação constante, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10.

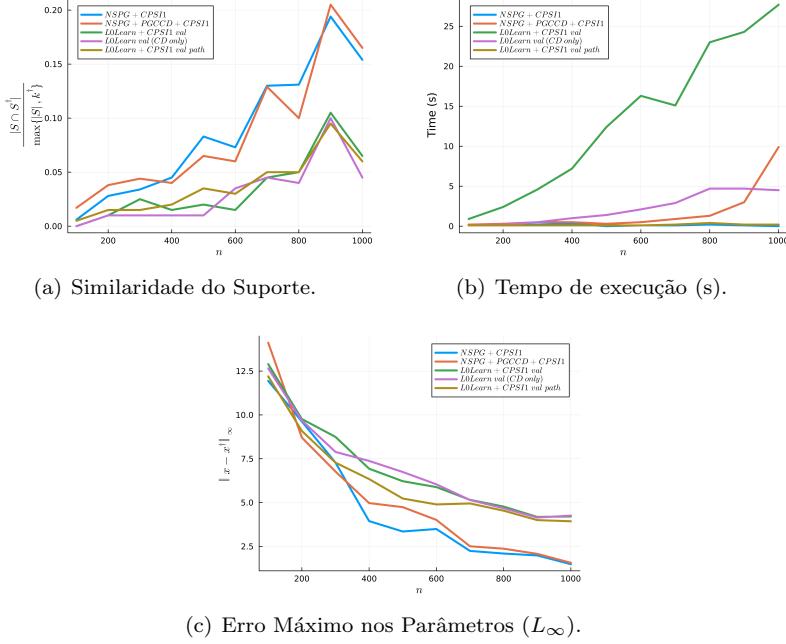


Figura 2: Comparação entre os métodos propostos e variantes do L0Learn. Correlação constante, $\rho = 0.9$, $p = 1000$, $k^\dagger = 20$, SNR = 5.

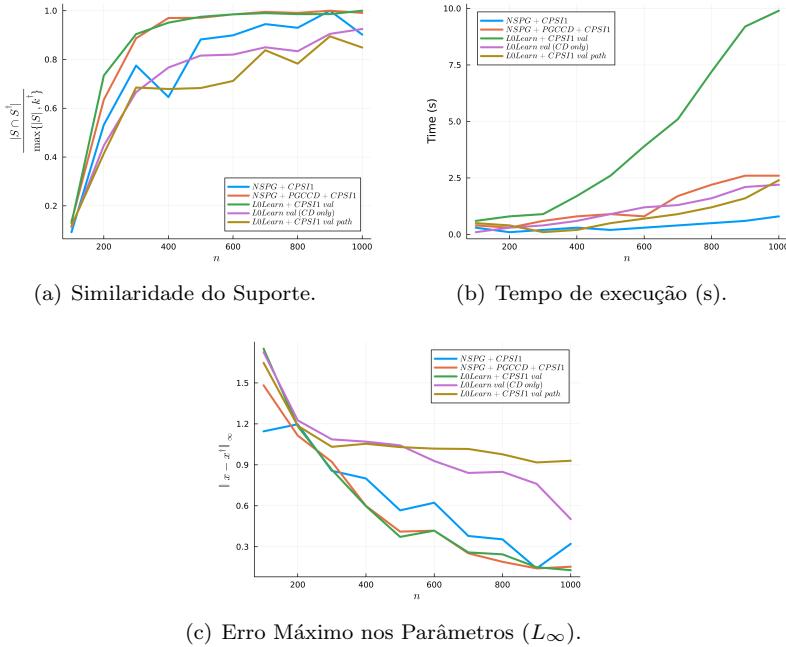


Figura 3: Comparação entre os métodos propostos e variantes do L0Learn. Correlação exponencial, $\rho = 0.9$, $p = 1000$, $k^\dagger = 20$, SNR = 5.

Apêndice C

Análise do Refinamento

Neste apêndice, detalhamos o impacto do refinamento final na qualidade da solução. Os gráficos abaixo mostram, para cada método e cenário, a frequência com que o candidato a partir do zero (*zero-based*) foi escolhido em detrimento do candidato do caminho (*path-based*) (barra “Wins”, que contabiliza vitórias, empates e derrotas na similaridade quando o zero é escolhido), e o impacto dessa escolha na similaridade do suporte (“Sim Improv”). A barra agrupa visualmente “Wins” (melhora no suporte), “Ties” (igualdade) e “Losses” (piora no suporte) condicionados à escolha pelo zero.

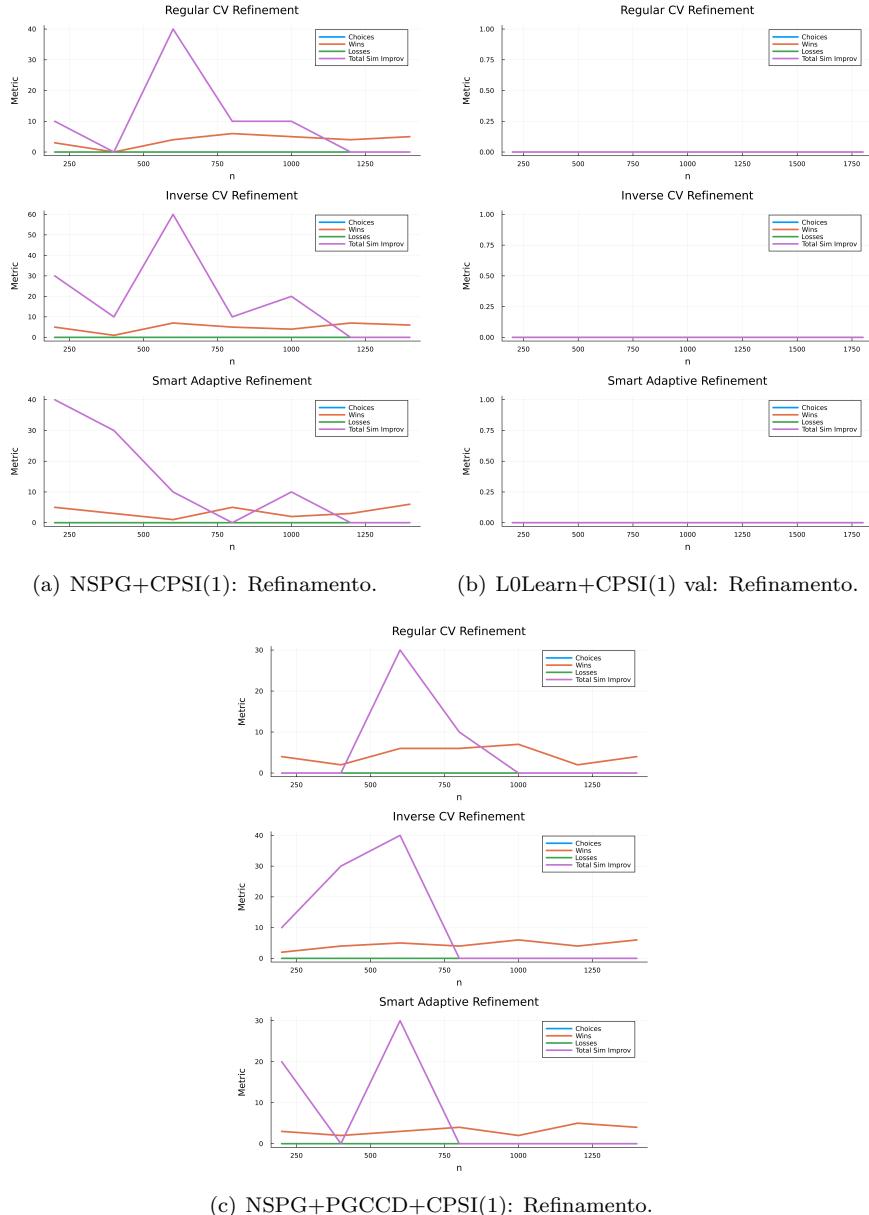


Figura 1: Impacto do refinamento final (Correlação exponencial, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10).

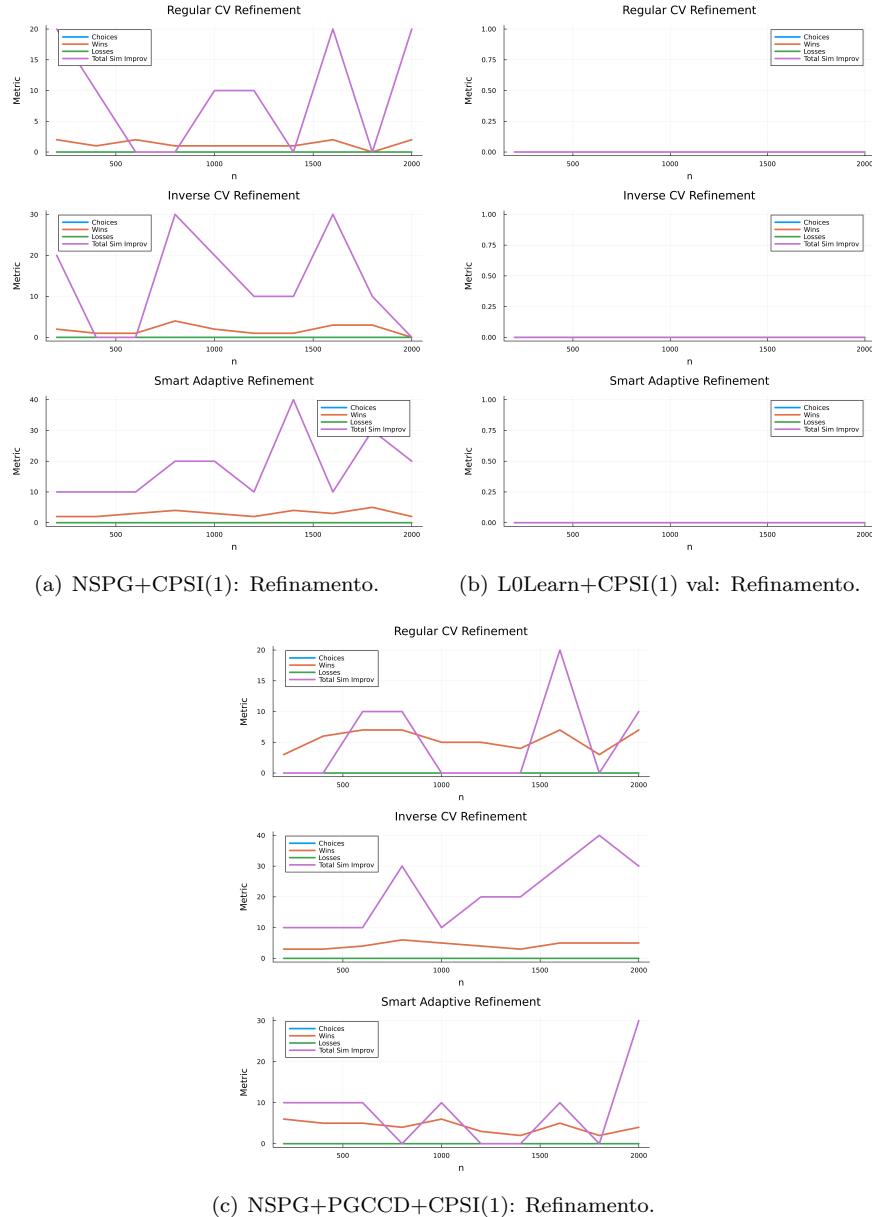


Figura 2: Impacto do refinamento final (Correlação constante, $\rho = 0.5$, $p = 2000$, $k^\dagger = 100$, SNR = 10).

Apêndice D

Detalhes de Implementação do PGCCD

O algoritmo PGCCD emprega a estratégia de *Active Set* conforme descrito em Hazimeh and Mazumder [2020a]. O método executa ciclos completos de des-cida coordenada até que o suporte da solução se mantenha inalterado por 10 iterações consecutivas (`ActiveSetNum=10`). A partir desse ponto, as iterações são restritas apenas às variáveis não-nulas (conjunto ativo) até a convergência. Por fim, uma verificação de otimalidade é realizada em todas as variáveis para garantir que nenhuma coordenada fora do suporte viola as condições de mínimo local. Adicionalmente, para reduzir o custo computacional nas fases iniciais, utilizamos uma estratégia de ordenação parcial (*partial greedy sort*), onde somente os 25% das variáveis com maior correlação com o resíduo são ordenadas para a varredura gulosa.

Referências Bibliográficas

- Gabriel Belém Barbosa and Paulo José da Silva Silva. Relaxing smoothness conditions for non-monotone optimization methods applied to discontinuous composite problems. Manuscript in progress, 2024.
- Jonathan Barzilai and Jonathan M. Borwein. Two-Point Step Size Gradient Methods. *IMA Journal of Numerical Analysis*, 8(1):141–148, 01 1988. ISSN 0272-4979. doi: 10.1093/imanum/8.1.141. URL <https://doi.org/10.1093/imanum/8.1.141>.
- Amir Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017. doi: 10.1137/1.9781611974997. URL <https://pubs.siam.org/doi/abs/10.1137/1.9781611974997>.
- Amir Beck and Yonina C. Eldar. Sparsity constrained nonlinear optimization: Optimality conditions and algorithms. *SIAM Journal on Optimization*, 23(3):1480–1509, 2013. doi: 10.1137/120869778. URL <https://doi.org/10.1137/120869778>.
- Peter Bühlmann and Sara van de Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2011. ISBN 3642201911.
- Thomas Blumensath and Mike E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009. ISSN 1063-5203. doi: <https://doi.org/10.1016/j.acha.2009.04.002>. URL <https://www.sciencedirect.com/science/article/pii/S1063520309000384>.
- Patrick Breheny and Jian Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, 5(1):232–253, 2011. doi: 10.1214/10-AOAS388. URL <https://doi.org/10.1214/10-AOAS388>.
- Yu-Hong Dai, Mehiddin Al-Baali, and Xiaoqi Yang. *A Positive Barzilai–Borwein-Like Stepsize and an Extension for Symmetric Linear Systems*, pages 59–75. na, 01 2015. ISBN 978-3-319-17688-8. doi: 10.1007/978-3-319-17689-5_3.
- Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001. ISSN 01621459. URL <http://www.jstor.org/stable/3085904>.

- Jerome H. Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010. doi: 10.18637/jss.v033.i01. URL <https://www.jstatsoft.org/index.php/jss/article/view/v033i01>.
- Tom Goldstein, Christoph Studer, and Richard Baraniuk. A field guide to forward-backward splitting with a fasta implementation. *ArXiv*, abs/1411.3406, 2014. URL <https://api.semanticscholar.org/CorpusID:9037325>.
- L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for newton’s method. *SIAM Journal on Numerical Analysis*, 23(4):707–716, 1986. ISSN 00361429. URL <http://www.jstor.org/stable/2157617>.
- Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC, 2015. ISBN 1498712169.
- Hussein Hazimeh and Rahul Mazumder. Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *Operations Research*, 68(5):1517–1537, September 2020a. ISSN 0030-364X. doi: 10.1287/opre.2019.1919. URL <https://doi.org/10.1287/opre.2019.1919>.
- Hussein Hazimeh and Rahul Mazumder. *L0Learn: Fast Algorithms for Best Subset Selection*, 2020b. URL <https://CRAN.R-project.org/package=L0Learn>. R package version 2.0.0.
- Hussein Hazimeh, Rahul Mazumder, and Ali Saab. Sparse regression at scale: branch-and-bound rooted in first-order optimization. *Mathematical Programming*, 196:1–42, 10 2021. doi: 10.1007/s10107-021-01712-4.
- Christian Kanzow and Patrick Mehlitz. Convergence properties of monotone and nonmonotone proximal gradient methods revisited. *Journal of Optimization Theory and Applications*, 195:624 – 646, 2021. URL <https://api.semanticscholar.org/CorpusID:244896538>.
- Rahul Mazumder, Peter Radchenko, and Antoine Dedieu. Subset selection with shrinkage: Sparse linear modeling when the snr is low. *Operations Research*, 71, 08 2017. doi: 10.1287/opre.2022.2276.
- B.S. Mordukhovich. *Variational analysis and applications*. Springer, Berlin, 2018. doi: 10.1007/978-3-319-92775-6.
- Jean Jacques Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. *Comptes rendus hebdomadaires des séances de l’Académie des sciences*, 255:2897–2899, 1962. URL <https://hal.science/hal-01867195>.
- B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995. doi: 10.1137/S0097539792240406. URL <https://doi.org/10.1137/S0097539792240406>.

- Youngsuk Park, Sauptik Dhar, Stephen Boyd, and Mohak Shah. Variable metric proximal gradient method with diagonal barzilai-borwein stepsize. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3597–3601, 05 2020. doi: 10.1109/ICASSP40776.2020.9054193.
- Andrei Patrascu and Ion Necoara. Random coordinate descent methods for ℓ_0 regularized convex optimization. *IEEE Transactions on Automatic Control*, 60(7):1811–1824, 2015. doi: 10.1109/TAC.2015.2390551.
- MARCOS RAYDAN. On the barzilai and borwein choice of steplength for the gradient method. *IMA Journal of Numerical Analysis*, 13(3):321–326, 1993. ISSN 1464-3642. doi: 10.1093/imanum/13.3.321. URL <http://dx.doi.org/10.1093/imanum/13.3.321>.
- R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.
- Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2), April 2010. ISSN 0090-5364. doi: 10.1214/09-aos729. URL <http://dx.doi.org/10.1214/09-AOS729>.
- Bin Zhou, Li Gao, and Yu-Hong Dai. Gradient methods with adaptive stepsizes. *Computational Optimization and Applications*, 35:69–86, 09 2006. doi: 10.1007/s10589-006-6446-0.
- Junxian Zhu, Canhong Wen, Jin Zhu, Heping Zhang, and Xueqin Wang. A polynomial algorithm for best-subset selection problem. *Proceedings of the National Academy of Science*, 117(52):33117–33123, December 2020. doi: 10.1073/pnas.2014241117.