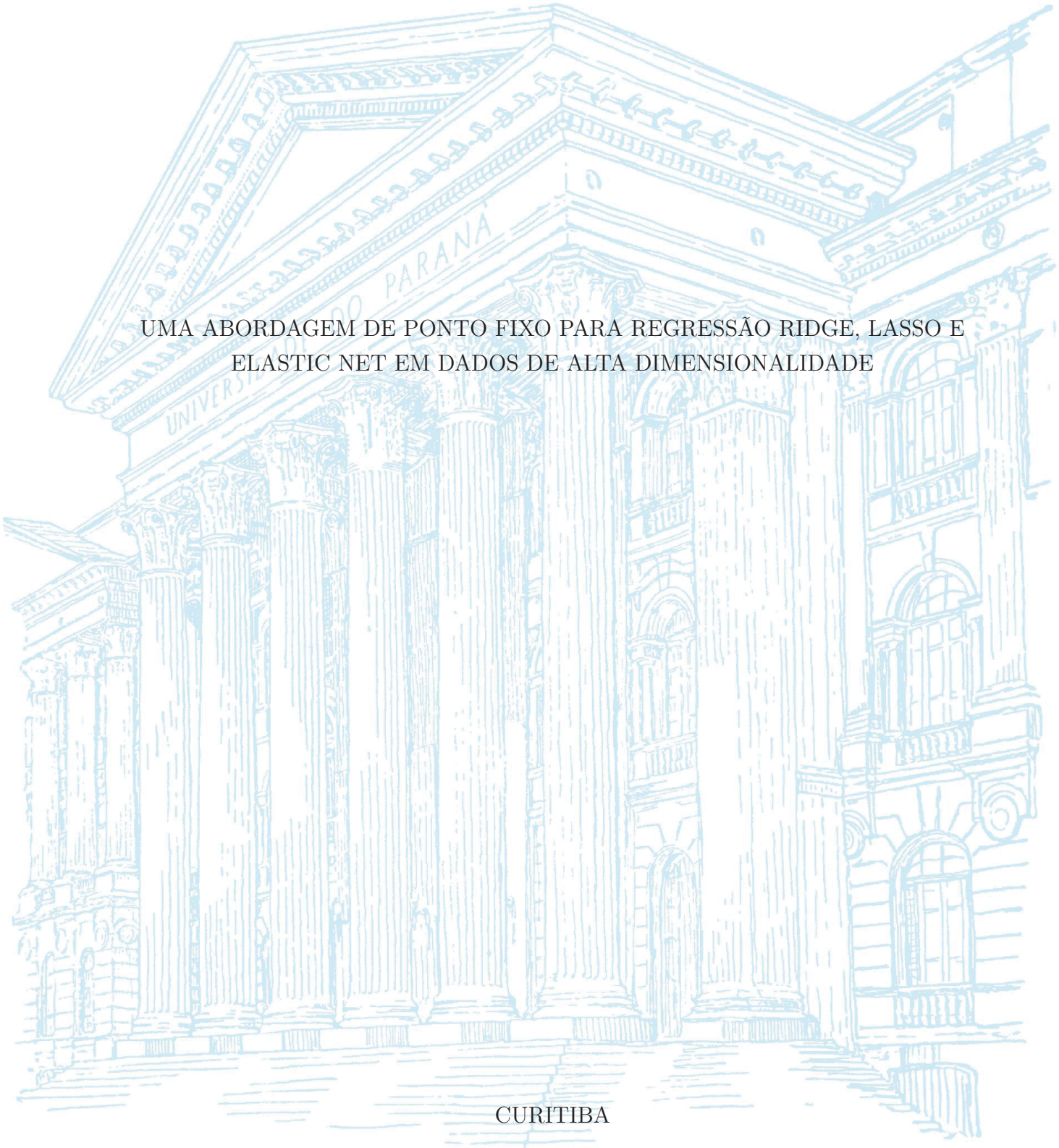
UNIVERSIDADE FEDERAL DO PARANÁ

DANIELA MIRAY IGARASHI



UMA ABORDAGEM DE PONTO FIXO PARA REGRESSÃO RIDGE, LASSO E ELASTIC NET EM DADOS DE ALTA DIMENSIONALIDADE

CURITIBA

2024

DANIELA MIRAY IGARASHI

UMA ABORDAGEM DE PONTO FIXO PARA REGRESSÃO RIDGE, LASSO E ELASTIC NET EM DADOS DE ALTA DIMENSIONALIDADE

Tese apresentada ao curso de Pós-Graduação em Métodos Numéricos em Engenharia, Setor de Ciências Exatas e de Tecnologia, Universidade Federal do Paraná, como requisito parcial para a obtenção do título de Doutor em Métodos Numéricos.

Orientador: Prof. Dr. Luiz Carlos Matioli

CURITIBA

2024

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Igarashi, Daniela Miray
Uma abordagem de ponto fixo para regressão Ridge, Lasso e Elastic Net Em dados de alta dimensionalidade / Daniela Miray Igarashi. – Curitiba, 2024.
1 recurso on-line : PDF.

Tese (Doutorado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Métodos Numéricos em Engenharia.

Orientador: Luiz Carlos Matioli

1. Teoria do ponto fixo. 2. Regressão de cumeeira (Estatística). 3. Mínimos quadrados. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Métodos Numéricos em Engenharia. III. Matioli, Luiz Carlos. IV . Título.

Bibliotecário: Leticia Priscila Azevedo de Sousa CRB-9/2029

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **DANIELA MIRAY IGARASHI** intitulada: **Uma abordagem de ponto fixo para regressão ridge, lasso e elastic net em dados de alta dimensionalidade**, sob orientação do Prof. Dr. LUIZ CARLOS MATIOLI, que após terem inquirido a aluna e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de doutora está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 20 de Março de 2024.

Assinatura Eletrônica
25/03/2024 13:04:38.0
LUIZ CARLOS MATIOLI
Presidente da Banca Examinadora

Assinatura Eletrônica
25/03/2024 22:35:06.0
PAULO SERGIO MARQUES DOS SANTOS
Avaliador Externo (UNIVERSIDADE FEDERAL DO PIAUí)

Assinatura Eletrônica
25/03/2024 22:02:58.0
GISLAINE APARECIDA PERIÇARO
Avaliador Externo (UNIVERSIDADE ESTADUAL DO PARANÁ)

Assinatura Eletrônica
25/03/2024 13:06:49.0
DIRCEU SCALDELAI
Avaliador Externo

Assinatura Eletrônica
11/04/2024 11:54:40.0
CESAR AUGUSTO TACONELI
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Para Terezinha, Masami, Felipe e Eduardo.

AGRADECIMENTOS

Gostaria de agradecer sinceramente à todos que direta ou indiretamente contribuíram para a realização deste trabalho. Mesmo que a jornada seja difícil em muitos momentos, este trabalho jamais seria finalizado não fosse por estas pessoas e sou imensamente grata.

Ao meu orientador professor Dr. Luiz Carlos Matioli, pelo apoio, confiança, e paciência, em todo o período de orientação. Obrigada por fazer parte da minha formação e construção da pessoa que hoje sou.

Aos amigos que fiz durante esses últimos anos, companheiros do PPGMNE, todas as pessoas que conheci ao morar em Curitiba desde o mestrado. Uma citação especial para o Tiago, ao compartilhar muitos momentos nessa jornada em comum. Obrigada por tornarem esta caminhada mais leve e agradável.

À minha família e amigos, pelo apoio incondicional por todo este tempo. Obrigada por compreender mesmo sem entender, por relevar minha a ausência, e serem a minha maior motivação em momentos difíceis.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio para realização deste estudo.

À Universidade Federal do Paraná pela oportunidade de realização deste curso.

Muito obrigada.

Ever tried. Ever failed. No matter. Try again. Fail again. Fail better.
Samuel Beckett

RESUMO

Em problemas de regressão linear múltipla, quando o número de características é muito maior que o número de observações, tem-se dados de alta dimensão, sendo um tipo de problema relevante dado que é comum em dados genômicos, análise de imagens, finanças e biologia molecular. Dados de alta dimensão podem ser afetados pela multicolinearidade, que ocorre quando duas ou mais variáveis independentes estão correlacionadas, e esse fenômeno pode levar a inferências não confiáveis. Uma abordagem de regularização, como regressão ridge, lasso ou elastic net, pode ser usada neste caso. Este trabalho apresenta um novo algoritmo baseado em ponto fixo para resolver o problema de regressão ridge. O algoritmo é baseado em reescrever a condição de otimização necessária de primeira ordem como uma iteração de ponto fixo e é um algoritmo de fácil implementação. Experimentos numéricos foram executados em problemas mal condicionados e de alta dimensão a fim de avaliar sua viabilidade. O algoritmo proposto foi também aplicado na solução das técnicas de lasso ou elastic net, nesse caso sendo utilizado para solução do subproblema do método de direções alternadas dos multiplicadores. Além disso, o algoritmo proposto foi utilizado na solução do problema de dados genômicos reais de alta dimensão sobre a produção de riboflavina (vitamina B_2) com *Bacillus subtilis* para fins ilustrativos. Os resultados mostram que o algoritmo proposto é competitivo em tempo de execução quando comparado ao método do gradiente conjugado, à rotina *mldivide* do MATLAB® e ao método do resíduo mínimo.

Palavras-chaves: Regressão Ridge; Ponto Fixo; Mínimos Quadrados; Lasso; Elastic Net.

ABSTRACT

In multiple linear regression, the challenge of high-dimensional data arises when the number of features far exceeds the number of observations. This issue is encountered across various fields, including genomics, image analysis, finance, and molecular biology. High-dimensional datasets often suffer from multicollinearity, where correlated independent variables lead to unreliable inferences. A regularization approach, such as ridge regression, lasso, or elastic net, can be used in this case. This work presents a new fixed-point-based algorithm to solve the ridge regression problem. The algorithm rewrites the necessary first-order optimization condition as a fixed-point iteration and is straightforward to implement. Numerical experiments were conducted on ill-conditioned and high-dimensional problems to evaluate its feasibility. The proposed algorithm was also applied to solve lasso or elastic net problems, in which case it was used to solve the subproblem of the alternating directions method of multipliers. Additionally, the proposed algorithm was applied to solve a high-dimensional real genomic data problem regarding riboflavin (vitamin B_2) production with *Bacillus subtilis* for illustrative purposes. The results show that the proposed algorithm is competitive in terms of execution time when compared to the conjugate gradient method, MATLAB®*mldivide* routine, and the minimum residual method.

Key-words: Ridge Regression; Fixed Point; Least Squares; Lasso; Elastic Net.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Um dos objetivos da análise de dados é levantar hipóteses e tirar conclusões dos dados. Uma forma de extrair informações que podem ser compreendidas por humanos é por meio de modelos matemáticos, fazendo aproximações razoáveis da realidade. Desta forma, é possível entender relações, fazer previsões, classificar informações e tomar decisões baseadas no conhecimento extraído dos dados.

Nesse contexto, considere uma matriz de dados $X \in \mathbb{R}^{n \times p}$ que descreve as observações de um certo evento ou experimento. Denota-se por n o número de pontos no conjunto dos dados ou observações, enquanto p denota o número de variáveis disponíveis para explicar o evento, conhecido também por atributos, características ou variáveis explicativas. Assim, o elemento x_{ij} da matriz X representa o valor da i-ésima observação da j-ésima característica.

Suponha ainda que existam informações adicionais referentes ao resultado do evento ou experimento, sendo representadas por $y \in \mathbb{R}^n$, conhecida como variável resposta. Assim, o objetivo é explicar os resultados y por meio das variáveis explicativas X_j com $i = 1 \dots p$, ou seja, encontrar uma aproximação que defina a relação

$$y = \mathfrak{f}(X) + \epsilon$$

em que $\mathfrak{f}$ é uma função desconhecida e ϵ é um termo de erro aleatório, independente de X.

Existem diversos métodos e técnicas com o intuito de estimar a relação entre as variáveis. As formulações e hipóteses naturalmente dependem das características dos dados. Geralmente, estes métodos são voltados para a função $\mathfrak{f}$, estimando a função em si, ou determinando parâmetros de um modelo construído para explicar a relação dos dados (JAMES et al., 2014). Para tanto, normalmente é feito uma otimização de uma função que representa o erro entre o valor real de y e a aproximação computada.

Uma das formas mais simples de construir um modelo é supor que a relação entre as variáveis é linear, o que se traduz no método de regressão linear múltipla $y = X\beta + \epsilon$, em que o vetor $\beta \in \mathbb{R}^p$ representa os parâmetros do modelo. Por sua simplicidade e interpretação compreensível, é um método aplicado à diversas áreas do conhecimento. Conforme é discutido no decorrer deste trabalho, a regressão linear encontra dificuldades em problemas com alta dimensão por conta da multicolinearidade entre atributos, e também em casos em que a matriz dos dados é mal condicionada, o que na prática indica que a matriz é quase singular. Em ambos os casos, a estimação dos parâmetros pode ser prejudicada.

Quando $p > n$, diz-se que a matriz X é de alta dimensão. Exemplos dessa característica surgem em problemas da medicina e biologia molecular (BÜHLMANN;

KALISCH; MEIER, 2014): ao investigar as causas de uma doença em um paciente, a expressão dos genes do paciente pode ser usada para explicar o evento de doença. Neste caso, o número de atributos é muito maior que o número de pacientes no estudo. Outra situação envolve usar termos de pesquisa para entender o padrão de compras de clientes, o número de pesquisas feitas e o número de palavras utilizadas pelos indivíduos pode fornecer uma grande quantidade de dados. Para os exemplos apresentados anteriormente, a variável resposta y poderia ser binária e indicar se um paciente tem ou não uma doença, enquanto que no segundo exemplo, a variável resposta poderia representar o que o cliente comprou após várias pesquisas. Ainda, segundo Fan e Lv (2008) os problemas surgem frequentemente na área genômica, como estudos de expressão gênica, imagens biomédicas, ressonância magnética funcional, tomografia, classificações de tumores, processamento de sinais, análise de imagens e finanças, onde o número de variáveis ou parâmetros pode ser muito maior que o tamanho de indivíduos observados n.

Segundo James et al. (2014), as técnicas desenvolvidas para mitigar o problema da alta dimensionalidade podem ser separadas em três classes. A primeira classe são técnicas de *seleção de atributos*, esta abordagem envolve identificar as características que estão mais relacionadas com a resposta e aplicar a regressão linear neste conjunto reduzido de variáveis explicativas. Um exemplo de técnica dessa classe é a regressão *stepwise*, em que as variáveis preditoras são adicionadas ou removidas sequencialmente de acordo com a sua contribuição no ajuste do modelo. A segunda classe de métodos envolve a *redução de dimensão* e uma técnica desta classe é a regressão por análise de componentes principais (PCA) (JOLLIFFE, 2002). O objetivo deste método é criar "novas" M variáveis explicativas que são chamadas de componentes principais. As componentes são calculadas como combinações lineares dos p atributos originais, são ortogonais entre si e são utilizadas para ajustar um modelo de regressão linear. Pelo fato de $M < p$ e nem todas as componentes principais serem selecionadas, tem-se a redução da dimensão do problema. A seleção de características relevantes ou redução de dimensionalidade é uma estratégia eficaz para lidar com dados de alta dimensionalidade. Ao transformar os dados de alta dimensionalidade em um espaço de menor dimensão, a carga computacional é significativamente reduzida. Ao mesmo tempo, é possível também obter estimativas precisas utilizando técnicas bem desenvolvidas para dados de menor dimensão (FAN; LV, 2008). Por fim, outros métodos se baseiam no encolhimento dos coeficientes da regressão, quando um termo de regularização é adicionado de forma que os coeficientes da regressão são encurtados e se aproximam de zero como no método de regressão ridge (HOERL; KENNARD, 1970), ou sejam exatamente zero e tem-se a técnica de lasso (TIBSHIRANI, R., 1996).

Regressão ridge, uma extensão da regressão linear, é um método que introduz um termo de regularização que penaliza os coeficientes do modelo por meio de uma norma ℓ_2, a norma euclidiana, ao quadrado. Esta penalização auxilia a mitigar algumas das dificuldades da regressão linear, particularmente, ao lidar com conjuntos de dados que

contêm preditores com alta correlação entre si ou possuem alta dimensão. Dessa forma, a regressão ridge resolve o problema de alta dimensionalidade acrescentando um termo de penalidade quadrática aos coeficientes da regressão linear, o que afasta o sistema da singularidade (CASAGRANDE, 2016). O uso da penalidade quadrática ℓ_2 tem o efeito de restringir os valores dos coeficientes, tornando-os menores e mais estáveis, reduzindo a sensibilidade a pequenas variações nos dados de treinamento. Neste caso, a penalidade funciona como uma regularização do problema, no sentido de que, ao adicionar este termo, obtém-se propriedades para o problema que o torna também menos difícil de ser resolvido.

O lasso é outro método utilizado em problemas de regressão linear e é uma alternativa à regressão ridge. Assim como a ridge, o lasso lida com problemas de multicolinearidade, mas de outra forma. Enquanto a ridge usa uma penalidade com a norma ℓ_2, o lasso utiliza uma penalidade com a norma ℓ_1, conhecida também como a norma do valor absoluto ou distância de Manhattan. Essa diferença na penalidade tem um efeito significativo no comportamento do modelo, o lasso tem a capacidade de forçar alguns dos coeficientes a se tornarem exatamente zero, o que significa que ele realiza uma seleção automática de variáveis (JAMES et al., 2014). Em outras palavras, o lasso pode eliminar variáveis menos relevantes do modelo, tornando-o mais simples e interpretável. Essa propriedade é especialmente útil em contextos de alta dimensionalidade.

Elastic net é uma abordagem híbrida que combina as vantagens tanto da regressão ridge quanto do lasso. Ao combinar as regularizações ℓ_1 e ℓ_2, o elastic net pode lidar efetivamente com problemas de multicolinearidade e seleção de variáveis ao mesmo tempo, encontrando um equilíbrio entre seleção de características e encolhimento de parâmetros.

Regressão ridge, lasso, e elastic net são técnicas de regularização usadas para melhorar o desempenho e generalização de modelos de regressão linear, especialmente em cenários que envolvem numerosas variáveis preditoras. Sendo que a escolha de cada uma delas envolve considerar a natureza do problema e suas características particulares. Dado que cada uma das abordagens possui propriedades diferentes.

Diferentes artigos e métodos foram apresentados para a solução do referido problema, assim como para questões de uma formulação geral em que a regressão ridge é considerada um caso particular. Até onde temos conhecimento, alguns dos trabalhos que utilizam uma abordagem de ponto fixo especificamente para a solução de um problema como a regressão ridge são os de Silva, Ribeiro e Periçaro (2018) e Ribeiro e Richtárik (2018). Nestes artigos, os autores propõem algoritmos para resolver uma formulação primal-dual, minimizando o *gap* de dualidade dos problemas, fundamentado no método do gradiente. No primeiro caso, os autores abordam a solução de problemas mal condicionados. Em Liu e Xu (2023), os autores apresentam formulações de algoritmos de ponto fixo primal-dual para outras normas como o lasso e elastic net, utilizando um algoritmo de gradiente proximal para solução dos problemas.

O presente trabalho concentra-se na resolução do problema de otimização do modelo de regressão ridge, por meio da aplicação de um algoritmo de ponto fixo. O algoritmo é fundamentado na reescrita da condição necessária de otimalidade de primeira ordem como uma iteração de ponto fixo. Trata-se de um algoritmo que envolve apenas operações de matriz em suas iterações, de simples implementação e fácil compreensão. Experimentos numéricos foram executados no *software* MATLAB® para avaliar o desempenho do algoritmo proposto para problemas mal condicionados e de alta dimensionalidade.

O algoritmo proposto neste trabalho encontra aplicações para o problema de otimização do lasso e elastic net. Conforme delineado nas seções subsequentes, quando o Método de Direções Alternadas de Multiplicadores (ADMM) é utilizado para resolver os problemas das regularizações lasso e elastic net, as formulações de suas iterações são semelhantes a um problema de otimização como a regressão ridge. O ADMM é uma técnica de otimização para resolver problemas de otimização convexa, especialmente aqueles com restrições e estruturas separáveis. É reconhecida por lidar com problemas envolvendo múltiplas variáveis e restrições, decompondo o problema em subproblemas mais simples passíveis de uma solução iterativa. A iteração do ADMM alterna entre atualizar as variáveis primais e duais, e cada uma dessas atualizações é composta por subproblemas mais simples de serem resolvidos. Neste caso, o algoritmo proposto neste trabalho é utilizado para solução da atualização da variável primal.

1.1 OBJETIVOS DA PESQUISA

Neste trabalho, o enfoque será dado a métodos que empregam a abordagem de regularização, como a regressão ridge, lasso e elastic net. Esses métodos têm sido utilizados em problemas de regressão e aprendizado de máquina para lidar com desafios relacionados à multicolinearidade, alta dimensionalidade e seleção de variáveis. Um dos objetivos desta pesquisa é explorar e aprofundar o entendimento desses métodos de regularização, concentrando-se na resolução do problema de otimização que permite estimar os coeficientes do modelo de regressão ridge

Um dos objetivos principais e contribuição do presente trabalho é o desenvolvimento de um algoritmo de ponto fixo para tratar justamente da resolução de problemas no formato do subproblema de regressão ridge. O algoritmo desenvolvido envolve somente operações básicas com vetores e matrizes, sendo de fácil compreensão e implementação. O intuito é demonstrar resultados de convergência e também a avaliação do algoritmo por meio de experimentos numéricos de comparação a fim de analisar a sua performance.

Além disso, outra contribuição desta pesquisa será a aplicação do algoritmo desenvolvido para resolver o subproblema do método ADMM para lasso e elastic net. O ADMM é uma técnica de otimização utilizada para resolver problemas de otimização convexa com restrições, especificamente quando a função objetivo é separável. Ao utilizar

o algoritmo de ponto fixo desenvolvido neste trabalho, a resolução do subproblema do ADMM para lasso e elastic net será modificada, proporcionando uma abordagem diferente para a obtenção dos coeficientes de regressão nesses métodos de regularização.

1.2 ESTRUTURA DO TRABALHO

Os demais capítulos do presente trabalho estão estruturados da seguinte maneira: No capítulo 2, é discutido o conceito de regularização, apresentados diversos métodos com esta abordagem, e demais conceitos relacionados. No capítulo 3 é apresentada a proposta de algoritmo e foco principal deste trabalho. O capítulo 4 apresenta os resultados de experimentos numéricos realizados com o algoritmo proposto. Finalmente, o capítulo 5 apresenta as considerações finais do trabalho e encaminhamentos para prosseguimento da pesquisa.

2 REGULARIZAÇÃO

Este capítulo visa proporcionar uma base de conceitos para compreensão do método proposto nesta tese e as demais contribuições abordadas nos capítulos subsequentes. Mais especificamente, são apresentadas três técnicas que desempenham papéis fundamentais na análise de regressão regularizada como a regressão ridge, o lasso e a elastic net.

As referências consultadas para escrita das Seções 2.1 e 2.2 foram Hastie, Tibshirani e Friedman (2009), Silva (2016) e Saleh, Arashi e Kibria (2019). Os conceitos apresentados na seção 2.3 são baseados nos trabalhos de Parikh e Boyd (2014) e Kumar (2015). É possível obter as referências e uma visão mais detalhada dos conceitos apresentados na Seção 2.4 em Robert Tibshirani (1996), James et al. (2014) e Hastie, Tibshirani e Wainwright (2015). Os conceitos descritos na seção 2.5 têm como base os trabalhos de Zou e Hastie (2005) e Hastie, Tibshirani e Wainwright (2015).

2.1 REGRESSÃO LINEAR

Uma das formas de modelar problemas reais por meio de modelos matemáticos é supor que a relação entre as variáveis explicativas X_j e a resposta y é linear. Assim a regressão linear usual tem como intuito modelar a relação entre uma variável dependente e uma ou mais variáveis independentes ao ajustar uma função linear aos dados. Assim, escreve-se o modelo de regressão linear múltipla na forma de:

$$y = X\beta + \epsilon,$$

em que $X \in \mathbb{R}^{n\times p}$ é a matriz de dados do problema, $\beta \in \mathbb{R}^p$ representa os parâmetros da regressão e $\epsilon \in \mathbb{R}^n$ é um termo aleatório. Supõe-se que ϵ é um vetor de n variáveis aleatórias independentes com média zero e variância σ^2 constante.

O estimador de mínimos quadrados de β, denotado por $\hat{\beta}$ pode ser obtido pela minimização da soma dos quadrados dos erros (SQE), definida como

$$f(\beta) = ||y - X\beta||^2, \tag{2.1}$$

em que $||\cdot||$ representa a norma euclidiana ou norma ℓ_2.

Em sua forma matricial, o problema de otimização é dado por

$$\underset{\beta}{\text{minimizar}} \quad (y - X\beta)^T(y - X\beta). \tag{2.2}$$

Pela condição necessária de otimalidade de primeira ordem, se $\hat{\beta}$ for um minimizador local de f, então este satisfaz $\nabla f(\hat{\beta}) = 0$,

$$\nabla f(\hat{\beta}) = -2X^T y + 2X^T X\hat{\beta} = 0,$$

solucionado com relação à $\hat{\beta}$ e, se X^TX for não singular, obtém-se

$$\hat{\beta} = (X^TX)^{-1}X^Ty, \tag{2.3}$$

que é o estimador de mínimos quadrados.

Note que $\nabla^2 f(\beta) = 2X^TX$ é semidefinida positiva, pois, dado um $a \in \mathbb{R}^n$ em que $a \neq 0$,

$$a^T(X^TX)a = (Xa)^TXa = ||Xa||^2 \geq 0.$$

Se X tem posto cheio, tem-se que X^TX é definida positiva. Portanto, f é uma função convexa. Desta forma, conclui-se que qualquer solução local é uma solução global do problema e, portanto, o estimador $\hat{\beta}$ é um minimizador global de f.

O estimador $\hat{\beta}$ em (2.3) depende da inversão da matriz X^TX. Por sua vez, em problemas em que X não possui posto cheio (e não tem colunas linearmente independentes), X^TX é singular, e não é possível obter uma solução única para os coeficientes de β. O mesmo ocorre quando X possui mais características que observações ($p > n$), pois nesse caso não é possível que a matriz X tenha colunas linearmente independentes (apenas linhas), o que também faz com que X^TX seja singular. Outro cenário desfavorável ocorre quando a matriz dos dados é mal-condicionada, obtendo X^TX quase singular, ou existe colinearidade entre observações e/ou atributos. Neste caso, tem-se estimativas imprecisas, com alta variação nos coeficientes, o que pode superestimar as medidas de erro padrão dos parâmetros, resultando em p-valores imprecisos ou falsos.

Por conta dessas considerações, resolver o problema de mínimos quadrados com o intuito de ajustar dados e fazer previsões pode produzir um bom resultado com os dados utilizados no momento da estimação dos parâmetros, mas não garante que o modelo será capaz de generalizar bem para dados novos.

2.2 REGRESSÃO RIDGE

Uma alternativa para algumas dificuldades que a regressão linear enfrenta foi a publicação do método de regressão ridge por Hoerl e Kennard (1970). Nesta abordagem, adiciona-se um parâmetro de regularização com o intuito de limitar o crescimento de β, ou seja, adiciona-se uma penalidade à magnitude dos coeficientes. Os parâmetros do modelo podem ser encontrados ao solucionar o problema

$$\underset{\beta}{\text{minimizar}} \quad \frac{1}{2}||y - X\beta||^2 + \frac{\lambda}{2}||\beta||^2 \tag{2.4}$$

em que $\lambda > 0$ é um parâmetro de regularização, a ser determinado separadamente.

Assim como na regressão linear, a regressão ridge busca estimar parâmetros β que se ajustam melhor aos dados ao minimizar a SQE. O segundo termo de (2.4), $\frac{\lambda}{2}||\beta||^2$,

pode ser visto como uma penalização, ou seja, possuirá valor baixo quando os coeficientes $\beta_1, \beta_2, \ldots, \beta_p$ são próximos de zero. Dessa forma, o parâmetro λ serve para controlar o impacto relativo desses dois termos nas estimativas do coeficiente de regressão. Quanto maior o valor de λ, maior será o encurtamento nos coeficientes, e quanto menor o valor de λ, mais a regressão se aproxima do estimador de mínimos quadrados usual. Quando $\lambda = 0$, não há efeito da regularização, e tem-se o estimador de mínimos quadrados (2.3).

Cabe observar que a regressão ridge produz estimativas de parâmetros $\hat{\beta}_\lambda$ diferentes para cada valor de λ. No decorrer deste trabalho, discute-se como definir um valor para λ. Mais especificamente, a Seção 2.6 discorre uma das metodologias que podem ser utilizadas para a escolha do parâmetro, e a Seção 4.4 apresenta os valores adotados nos testes numéricos deste trabalho.

Observa-se que em (2.4), apenas os coeficientes de $\beta = (\beta_1, \cdots, \beta_p)^T$ são considerados na função objetivo, enquanto o intercepto β_0 é omitido. Como o intuito desta abordagem é regularizar os coeficientes e reduzir a relação estimada entre cada variável com a variável resposta, não é necessário reduzir o intercepto, que é simplesmente um valor médio da variável resposta quando dado um i, tem-se $x_{ij} = 0$ para todo $j = 1, \ldots, p$. Assumindo que as variáveis preditoras (colunas de X) foram centralizadas para ter média zero antes que o modelo seja ajustado ($\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0$), o intercepto poderá ser estimado simplesmente como a média dos elementos de y e $\beta_0 = \bar{y} = \sum_{i=1}^{n} \frac{y_i}{n}$ (JAMES et al., 2014). Por conveniência, durante os desenvolvimentos deste trabalho, assume-se que a matriz de dados X foi centralizada (possui colunas com média zero). Então o intercepto é omitido do problema, sendo calculado depois da estimação dos parâmetros.

A análise de existência de uma solução para (2.4) é análoga a do problema (2.2). Denota-se, por simplicidade, a função objetivo do problema de regressão ridge por $f_{\mathrm{R}}(\beta) = \frac{1}{2}||y - X\beta||^2 + \frac{\lambda}{2}||\beta||^2$. Dessa forma, a hessiana associada a função f_{R} resulta numa matriz definida positiva que é dada por $\nabla^2 f_{\mathrm{R}}(\beta) = X^T X + \lambda I$, pois é a soma da matriz semidefinida positiva $X^T X$ com a matriz definida positiva λI, com $\lambda > 0$ e o estimador ridge é um minimizador. Do fato de que $\nabla^2 f_{\mathrm{R}}$ é definida positiva decorre também que f_{R} é convexa, o que garante que g admite um minimizador global.

Escrevendo o problema (2.4) na forma matricial, tem-se:

$$\underset{\beta}{\text{minimizar}} \quad \frac{1}{2}(y - X\beta)^T(y - X\beta) + \frac{\lambda}{2}\beta^T\beta.$$

Resolvendo $\nabla f_{\mathrm{R}}(\beta) = 0$, obtém-se

$$\hat{\beta}_{\mathrm{R}} = (X^T X + \lambda I)^{-1} X^T y. \tag{2.5}$$

Observa-se que, com a adição da penalidade quadrática $\lambda\beta^T\beta$ no problema, a solução (2.5) ainda é linear em y, da mesma forma que no estimador $\hat{\beta}$ de mínimos

quadrados. A diferença na solução do problema é a adição de um termo λ na diagonal principal de X^TX antes de invertê-la. Nota-se que, se $\alpha_1 > \ldots > \alpha_p$ representam os autovalores de X^TX, os autovalores de $(X^TX + \lambda I)^{-1}$ serão $(\alpha_j + \lambda)^{-1}$ para $j = 1, \ldots, p$. Se X^TX for singular, com menor autovalor α_k, o autovalor mínimo de $X^TX + \lambda I$ será $(\alpha_k + \lambda)$ e esta matriz não está tão próxima de ser singular, mesmo quando X^TX não possui posto cheio.

O problema (2.4) é escrito como um problema de otimização irrestrito. Como os parâmetros β não são restritos, eles podem se tornar grandes, o que resulta em alta variância (FRIEDMAN; HASTIE; TIBSHIRANI et al., 2001). Desta forma, para controlar a variância dos parâmetros, é igualmente possível escrevê-lo como um problema restrito da forma como segue:

$$\begin{aligned} \underset{\beta}{\text{minimizar}} \quad & ||y - X\beta||^2 \\ \text{s.a.} \quad & ||\beta||^2 \leq c \end{aligned} \tag{2.6}$$

para algum $c > 0$. Como a função objetivo é contínua e o conjunto viável é compacto, existe um minimizador global no conjunto viável do problema (2.6).

Os dois problemas são, naturalmente, equivalentes. Para provar a equivalência, é necessário mostrar que ambos têm a mesma solução. De fato, a seguir mostra-se que existe uma relação entre λ e c, e que o estimador $\hat{\beta}_{\mathrm{R}}$ em (2.5) para o problema irrestrito (2.4) também é solução de (2.6).

Retomando o problema de otimização irrestrito, pela condição necessária de otimalidade de primeira ordem, $\nabla f_{\mathrm{R}}(\hat{\beta}_{\mathrm{R}}) = 0$, ou seja,

$$\nabla f(\hat{\beta}_{\mathrm{R}}) + 2\lambda\hat{\beta}_{\mathrm{R}} = 0 \tag{2.7}$$

em que f é a função definida em (2.1).

Enquanto que, o problema restrito (2.6) pode ser resolvido pelo condições de otimalidade de Karush-Kuhn-Tucker (KKT). Construindo a Lagrangeana do problema restrito, tem-se:

$$(\beta, \nu) \in \mathbb{R}^p \times \mathbb{R}_+ \mapsto L(\beta, \nu) = ||Y - X\beta||^2 + \nu(||\beta||^2 - c).$$

Como $\hat{\beta}_{\mathrm{R}} \neq 0$, a condição de qualificação linear (LICQ) é satisfeita em $\hat{\beta}_{\mathrm{R}}$ e, usando (2.7), existe um $\nu^* \geq 0$ tal que as condições de KKT são satisfeitas:

$$\begin{cases} \nabla f(\hat{\beta}_{\mathrm{R}}) + 2\nu^*\hat{\beta}_{\mathrm{R}} = 0 \\ \nu^*(||\hat{\beta}_{\mathrm{R}}||^2 - c) = 0 \\ ||\hat{\beta}_{\mathrm{R}}||^2 \leq c \end{cases} .$$

De fato, tomando $\nu^* = \lambda$ e $c = ||\hat{\beta}_{\mathrm{R}}||_2^2$, as condições são satisfeitas. Assim, os problemas (2.4) e (2.6) têm a mesma solução $\hat{\beta}_{\mathrm{R}}$ (2.5).

2.2.1 Estimador de mínimos quadrados e regressão ridge

Dadas as discussões acerca da singularidade de X^TX e a equivalência entre a forma irrestrita e restrita do problema de regressão ridge, seria de interesse abordar também as diferenças existentes entre o estimador de mínimos quadrados e o estimador de ridge.

Considerando o caso em que X é ortogonal, ou seja, $X^TX = I$, pode-se mostrar que $\hat{\beta}_{\mathrm{R}}$ é proporcional à $\hat{\beta}$,

$$\begin{aligned}\hat{\beta}_{\mathrm{R}} &= (X^TX + \lambda I)^{-1}X^TY \\ &= (1+\lambda)^{-1}IX^TY \\ &= (1+\lambda)^{-1}(X^TX)^{-1}X^TY \\ \hat{\beta}_{\mathrm{R}} &= \frac{1}{1+\lambda}\hat{\beta}.\end{aligned}$$

Quando X não é ortogonal, uma forma de avaliar o efeito do parâmetro de regularização λ na diferença entre o estimador de mínimos quadrados e o estimador ridge é sob a perspectiva dos valores singulares. Por meio da decomposição em valores singulares (SVD) reduzida (A.1), pode-se escrever $X = UDV^T$, em que $U \in \mathbb{R}^{n\times p}$ e $V \in \mathbb{R}^{p\times p}$ são matrizes ortogonais, $D \in \mathbb{R}^{p\times p}$ é uma matriz diagonal com os valores singulares $d_1 \geq d_2 \geq \ldots \geq d_p$. Assim, o estimador de mínimos quadrados pode ser escrito como

$$\begin{aligned}\hat{\beta} &= (X^TX)^{-1}X^TY \\ &= (VDU^TUDV^T)^{-1}VDU^TY \\ &= (VD^2V^T)^{-1}VDU^TY \\ &= VD^{-2}V^TVDU^TY \\ &= VD^{-2}DU^TY.\end{aligned}$$

De forma similar, obtém-se

$$\begin{aligned}\hat{\beta}_{\mathrm{R}} &= (X^TX + \lambda I)^{-1}X^TY \\ &= (VDU^TUDV^T + \lambda I)^{-1}VDU^TY \\ &= (VD^2V^T + \lambda VV^T)^{-1}VDU^TY \\ &= V(D^2 + \lambda I)^{-1}V^TVDU^TY \\ &= V(D^2 + \lambda I)^{-1}DU^TY.\end{aligned}$$

A diferença entre os dois estimadores em relação à SVD, pode ser vista em

$$\begin{aligned}D^{-2}D = D^{-1} &= diag\left(\frac{1}{d_1}, \ldots, \frac{1}{d_p}\right), \\ (D^2 + \lambda I)^{-1}D &= diag\left(\frac{d_1}{d_1^2 + \lambda}, \ldots, \frac{d_p}{d_p^2 + \lambda}\right).\end{aligned} \tag{2.8}$$

Como $d_j^{-1} > d_j(d_j^2+\lambda)^{-1}$ com $\lambda > 0$, a regressão ridge encurta os valores singulares de X e as estimativas dos parâmetros de regressão.

Uma das motivações da utilização da regressão ridge originou-se com o tratamento de problemas mal-condicionados. Neste caso, o mal condicionamento é causado principalmente em problemas em que os valores singulares d_j são próximos de zero. Pela equação (2.8), percebe-se que valores singulares próximos de zero fazem os elementos da matriz D^{-1} crescerem e tomarem valores grandes, o que prejudica a estimativa de $\hat{\beta}$. O mesmo não ocorre na regressão ridge pois, quando os valores singulares de X se aproximam de zero, a adição de λ nos elementos $d_j(d_j^2 + \lambda)^{-1}$ evita o crescimento desproporcional.

Diz-se também que o estimador da regressão linear usual é não tendencioso, em outras palavras, $\mathbb{E}(\hat{\beta}) = \beta$. Por outro lado, a introdução do termo de regularização na regressão ridge introduz um pequeno viés nas estimativas dos parâmetros.

$$\mathbb{E}(\hat{\beta}_{\mathrm{R}}) = X^T X(X^T X + \lambda I)^{-1}\beta$$

para $\lambda \neq 0$. Esse viés se deve ao fato de que o termo de regularização tende a encolher os coeficientes para zero, mesmo que alguns deles possam explicar um pouco melhor os dados.

No entanto, esse viés leva a uma redução significativa na variância das estimativas dos parâmetros do modelo. Na regressão linear, quando o número de preditores é grande ou quando alguns preditores são altamente correlacionado, o modelo se torna sensível a pequenas alterações nos dados, levando a uma alta variância. A regressão ridge atenua esse problema controlando o crescimento dos coeficientes, tornando o modelo mais estável e menos sensível a variações nos dados.

2.3 OPERADOR PROXIMAL

Algoritmos proximais são métodos de otimização utilizados para resolver problemas de otimização convexa, especialmente aqueles que envolvem funções não diferenciáveis. A ideia central por trás dos algoritmos proximais é lidar eficientemente com a não diferenciabilidade das funções objetivo, por meio da introdução de um operador proximal. Algoritmos proximais são particularmente úteis quando os operadores proximais relevantes podem ser avaliados com eficiência (PARIKH; BOYD, 2014). Este é o caso de funções objetivo com a norma ℓ_1, em que o operador de soft thresholding é justamente a aplicação do operador proximal para funções objetivo que possuem esta norma.

Dessa forma, a presente seção tem como objetivo estabelecer o conceito de operador proximal e o operador de soft thresholding, uma vez que estes são fundamentais para as seções subsequentes deste trabalho.

Definição 2.1 (Operador Proximal)**.** Dada uma função $f : \mathbb{R}^n \to \mathbb{R}^n$, o operador proximal de f é dado por

$$\text{prox}_f(x) = \underset{u}{\text{argmin}}\left\{ f(u) + \frac{1}{2}\|u - x\|_2^2 \right\} \quad \forall x \in \mathbb{R}^n. \tag{2.9}$$

O operador proximal geralmente leva cada ponto $x \in \mathbb{R}^n$ para um conjunto de pontos em $\mathbb{R}^n$ que minimiza a expressão (2.9). Quando f é uma função convexa, a função minimizada em (2.9) é estritamente convexa, obtendo então um $\text{prox}_f(x)$ único para cada $x \in \mathbb{R}^n$.

Pela definição 2.1, pode-se pensar $\text{prox}_f(x)$ como um ponto que se compromete entre minimizar f e estar próximo de x. Por esse motivo, $\text{prox}_f(x)$ é conhecido também como ponto proximal de x em relação a f.

Proposição 2.2. Seja $f : \mathbb{R} \to \mathbb{R}$ definida por

$$f(x) = \begin{cases} \lambda x, & \text{se } x \geq 0 \\ \infty, & \text{caso contrário} \end{cases}.$$

Então $\text{prox}_f(x) = [x - \lambda]_+$. Este é o operador de soft-thresholding unilateral.

Demonstração. Observe que f é diferenciável para $x > 0$. $\text{prox}_f(x)$ é o minimizador da função

$$g(u) = \begin{cases} g_1(u), & \text{se } u \geq 0 \\ \infty, & \text{se } u < 0 \end{cases}$$

em que

$$g_1(u) = \lambda u + \frac{1}{2}(u - x)^2.$$

g é uma função convexa, e g é diferenciável para $u > 0$ com $g'(u) = g_1'(u)$ para $u > 0$. O único ponto não diferenciável é $u = 0$ no domínio de g, que é $u \geq 0$.

Se $g'(u) = 0$, então u é um minimizador de g, pois g é convexa. $g_1'(u) = 0$ resulta em $u = x - \lambda$.

Se $x > \lambda$, tem-se $u > 0$, então $g'(x - \lambda) = g_1'(x - \lambda) = 0$. Então $\text{prox}_f(x) = x - \lambda$ para $x > \lambda$. Se $x \leq \lambda$, tem-se $u < 0$, então $g'(u)$ nunca será zero. Como existe um mínimo para g, ele deve ocorrer em um ponto não diferenciável. O único ponto não diferenciável é $u = 0$. Então $\text{prox}_f(x) = 0$ para $x \leq \lambda$.

■

Exemplo 2.3. Considere a função quadrática $g(u) = 2u + \frac{1}{2}(u - 1)^2$, neste caso, tem-se $\lambda = 2$ e $x = 1$. Se o termo $2u$ fosse desconsiderado, o valor mínimo da função ocorreria em $u = 1$. No entanto, a presença do termo $2u$ desloca o mínimo para a esquerda em

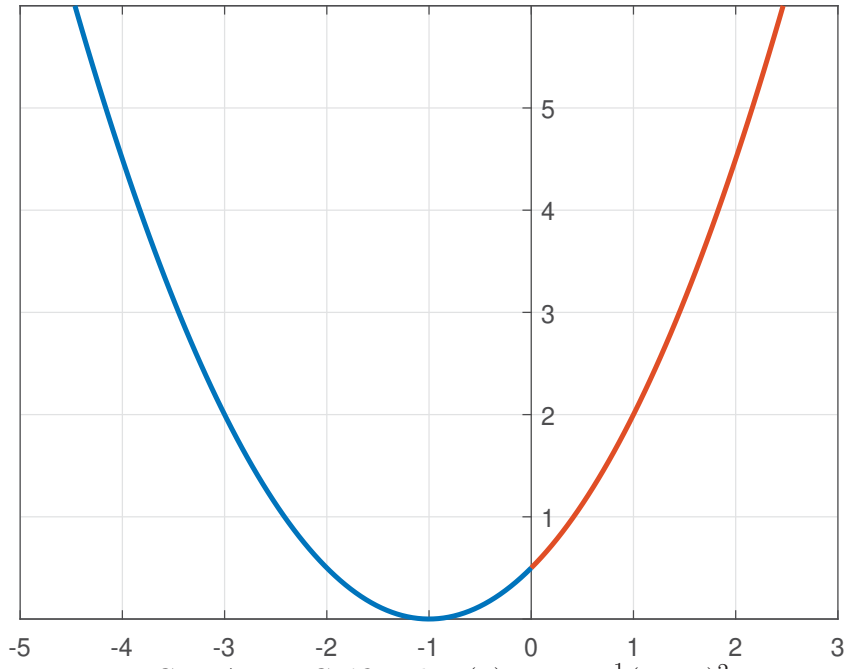


FIGURA 1 – Gráfico de $g(u) = 2u + \frac{1}{2}(u-1)^2$
Fonte: Elaborado pela autora.

2 unidades, o que resulta em $u = -1$. Se o domínio da função for limitado a $u \geq 0$, representado pela parte laranja do gráfico na Figura 1, o valor mínimo é obtido em $u = 0$.

Para uma função $g(u) = \lambda u + \frac{1}{2}(u-x)^2$ com domínio $u \geq 0$, se $x > \lambda$, o mínimo é obtido em $u = x - \lambda$. Caso $x < \lambda$, o mínimo é obtido em $u = 0$.

Proposição 2.4. Seja $f : \mathbb{R} \to \mathbb{R}$ definida por $f(x) = \lambda|x|$, então

$$\text{prox}_f(x) = \text{sinal}(x)\,[|x| - \lambda]_+.$$

Demonstração. Observe que f é diferenciável para $x \neq 0$. O $\text{prox}_f(x)$ é o minimizador da função

$$g(u) = \begin{cases} g_1(u) = \lambda u + \frac{1}{2}(u-x)^2 & u \geq 0 \\ g_2(u) = -\lambda u + \frac{1}{2}(u-x)^2 & u < 0 \end{cases}$$

Se o mínimo é obtido em $u > 0$, então $g_1'(u) = \lambda + u - x = 0$, e $u = x - \lambda$. Como assumimos que $u > 0$, então $x > \lambda$. Dessa forma, se $x > \lambda$ então $u = x - \lambda$ é o minimizador. De forma análoga, se $u < 0$, tem-se que $g_2'(u) = -\lambda + u - x = 0$ e $u = x + \lambda$. Se $u < 0$, tem-se $x < -\lambda$ e o minimizador é $u = x + \lambda$. Para $-\lambda < x < \lambda$ o minimizador está no único ponto de não diferenciabilidade de g, que é $u = 0$. Logo,

$$\text{prox}_f(x) = \begin{cases} x - \lambda, & x > \lambda \\ 0, & -\lambda < x < \lambda \\ x + \lambda, & x < -\lambda \end{cases}.$$

■

Este é conhecido também como operador de soft thresholding.

Definição 2.5. O operador de soft thresholding $S_\kappa : \mathbb{R} \to \mathbb{R}$ é definido por

$$S_\kappa(x) = \begin{cases} x - \kappa, & x > \kappa \\ 0, & |x| \leq \kappa \\ x + \kappa, & x < -\kappa \end{cases}.$$

Algumas definições equivalentes são

$$S_\kappa(x) = (x - \kappa)_+ - (-x - \kappa)_+.$$

$$S_\kappa(x) = \text{sinal}(x)\,[|x| - \kappa]_+.$$

Proposição 2.6. O minimizador de $f : \mathbb{R} \to \mathbb{R}$ definida por

$$f(x) = \lambda|x| + \frac{\rho}{2}(x - u)^2$$

é $S_{\lambda/\rho}(u)$.

Demonstração.

$$f(x) = \begin{cases} f_1(x) = \lambda x + \frac{\rho}{2}(x - u)^2, & x \geq 0 \\ f_2(x) = -\lambda x + \frac{\rho}{2}(x - u)^2, & x < 0 \end{cases}.$$

Se o mínimo é obtido em $x > 0$, então

$$0 = f'(x) = f_1'(x) = \lambda + \rho(x - u),$$

que resulta no ponto $x = u - \frac{\lambda}{\rho}$. Como $x > 0$ implica-se $u > \frac{\lambda}{\rho}$. Os outros casos são obtidos de forma similar aos apresentados na prova da Proposição 2.4. O minimizador de f é então,

$$S_{\lambda/\rho}(u) = \begin{cases} u - \frac{\lambda}{\rho}, & u > \frac{\lambda}{\rho} \\ 0, & |u| \leq \frac{\lambda}{\rho} \\ u + \frac{\lambda}{\rho}, & u < -\frac{\lambda}{\rho} \end{cases}.$$

■

Definição 2.7. O operador de soft thresholding por elemento $S_\kappa : \mathbb{R}^n \to \mathbb{R}^n$ é definido como

$$S_\kappa(x_i) = \begin{cases} x_i - \kappa, & x_i > \kappa \\ 0, & |x_i| \leq \kappa \\ x_i + \kappa, & x_i < -\kappa \end{cases},$$

para $i = 1 \ldots n$.

Proposição 2.8. O minimizador de $f : \mathbb{R}^n \to \mathbb{R}$ definida por

$$f(x) = \lambda\|x\|_1 + \frac{\rho}{2}\|x - u\|_2^2 \tag{2.10}$$

é $S_{\lambda/\rho}(u)$ aplicado elemento por elemento.

Demonstração. O resultado segue do fato de que f é separável como $f(x_i) = \sum f_i(x_i)$, $i = 1 \dots n$ com

$$f_i(x) = \lambda|x_i| + \frac{\rho}{2}(x_i - u)^2.$$

■

A Proposição 2.8 é um resultado proficiente para algoritmos de solução de problemas com funções compostos pela norma ℓ_1. Este resultado possibilita encontrar o minimizador de uma função como (2.10) dispensando um processo de minimização por meio da aplicação do operador de soft thresholding.

2.4 LASSO

O lasso (*Least Absolute Shrinkage and Selection Operator*) é um método de regressão penalizada assim como a regressão ridge, apresentando variações pequenas e significativas. Introduzida por Robert Tibshirani (1996), ele encontra os parâmetros do modelo de regressão por meio da solução do seguinte problema

$$\underset{\beta \in \mathbb{R}^p}{\text{minimizar}} \quad f_{\text{L}}(\beta) = \frac{1}{2}\|y - X\beta\|^2 + \lambda\|\beta\|_1. \tag{2.11}$$

Neste caso, a norma ℓ_1 é definida por $\|\beta\|_1 = \sum |\beta_j|$. Ao comparar os problemas (2.4) e (2.11), tem-se que as formulações são similares, a única diferença é que o termo $\|\beta\|^2$ na regressão ridge é substituído por $\|\beta\|_1$ no lasso.

Neste caso, a função de regularização é convexa mas não é diferenciável em $\beta = \mathbf{0}$, o que faz com que algoritmos dependentes da derivada da função não sejam aplicáveis para este problema sem modificações consideráveis.

Uma forma equivalente de escrever este problema, com $t > 0$, é

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \frac{1}{2}\|y - X\beta\|^2 \\ \text{s.a.} \quad & \|\beta\|_1 \leq t. \end{aligned} \tag{2.12}$$

A correspondência que pode ser feita entre os problemas equivalentes (2.11) e (2.12), ocorre quando a solução de (2.11) dada por $\hat{\beta}_L$ também é solução do problema (2.12) com $t = \sum_{j=1}^{p} \left|\hat{\beta}_j\right|$ (FRIEDMAN; HASTIE; HÖFLING et al., 2007).

De forma similar ao apresentado para regressão ridge na Seção 2.2, o intercepto β_0 não está presente na formulação dos problemas (2.11) e (2.12). Neste caso, X é padronizada de forma que cada coluna é centralizada ($\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0$) e possui variância unitária ($\frac{1}{n}\sum_{i=1}^{n} x_{ij}^2 = 1$), y também é centralizada ($\frac{1}{n}\sum_{i=1}^{n} y_i = 0$). Essas condições são convenientes pois permitem que o termo do intercepto seja omitido na otimização. Dada uma solução ótima $\hat{\beta}$ com os dados centralizados, a solução original com os dados não centralizados é o mesmo $\hat{\beta}$, e o intercepto pode ser obtidos por

$$\hat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_{ij}\hat{\beta}_j$$

em que $\bar{y}$ e $\bar{x}_{ij}$ são as médias dos dados originais. Então β_0 pode ser omitido da função objetivo.

O lasso também reduz a magnitude das estimativas dos coeficientes para zero, de forma similar à regressão ridge. Porém, com a utilização da penalidade ℓ_1 tem-se que algumas estimativas dos parâmetros são exatamente zero, dependendo se o parâmetro λ é suficientemente grande.

Quando a estimativa de um coeficiente é igual a zero, pode-se inferir que a variável preditora correspondente não é significativa para o modelo. Nesse sentido, o lasso é capaz de selecionar variáveis para o modelo. Além disso, o lasso produz modelos esparsos, ou seja, modelos que incluem apenas um subconjunto das variáveis originais. Neste contexto, o parâmetro λ de (2.12) controla a esparsidade do modelo. Quanto maior o valor de λ, mais restrita será a seleção e mais parâmetros β_j serão zerados.

Quando a dimensão p é alta, é razoável assumir que apenas um pequeno número de preditores entre $X_1, X_2 \cdots X_j$ (colunas de X) contribui para a resposta, o que equivale a assumir idealmente que o vetor de parâmetros β é esparso. Com esparsidade, a seleção de variáveis pode melhorar a precisão da estimativa, identificando efetivamente o subconjunto de preditores importantes, e também aprimorar a interpretabilidade do modelo com uma representação mais parcimoniosa (FAN; LV, 2008).

As formulações (2.6) e (2.12) estão representadas no contexto de um problema de duas dimensões na Figura 2. Se adotado um valor alto para t, a solução $\hat{\beta}$ satisfaz a restrição dos problemas (2.6) e (2.12) e estará dentro da região azul viável da Figura 2. Neste caso a solução da regressão ridge e lasso coincide com a solução do problema irrestrito $\hat{\beta}$, ou quando tem-se $\lambda = 0$. No caso da Figura 2, $\hat{\beta}$ não satisfaz a restrição do problema, e a solução do ridge e lasso estarão na intersecção entre as curvas de nível em vermelho, e a restrição em azul. Como a regressão ridge possui uma área em formato de círculo, a interseção entre as curvas de nível a área azul ocorre geralmente fora dos eixos coordenados, e os parâmetros de $\hat{\beta}$ são diferentes de zero. No lasso, percebe-se que a restrição forma uma área em formato de diamante, com cantos retos. Dessa forma, a interseção entre as curvas de nível e a restrição pode ocorrer nesses "cantos", que são

justamente onde estão os eixos coordenados, fazendo com que os coeficientes de $\hat{\beta}$ sejam exatamente zero.

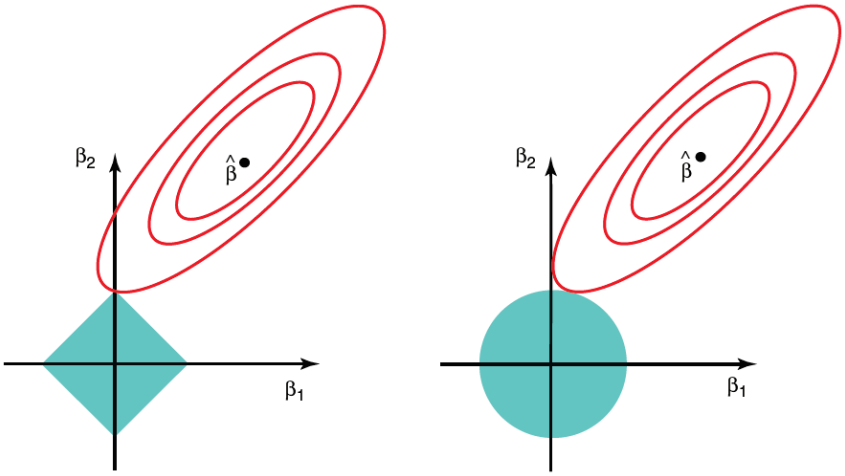


FIGURA 2 – Contornos das funções objetivo e região da restrição para o lasso (esquerda) e regressão ridge (direita). As áreas azuis são as regiões de restrição $|\beta_1| + |\beta_2| \leq t$ e $\beta_1^2 + \beta_2^2 \leq t$, respectivamente. Enquanto as elipses vermelhas são os contornos da função de soma dos quadrados dos resíduos. O ponto $\hat{\beta}$ representa a estimativa usual (irrestrita) de mínimos quadrados.
Fonte: James et al. (2014)

2.4.1 Algoritmos

O problema de otimização que é solucionado para encontrar os parâmetros do modelo do lasso poderia ser formulado como um problema cônico de segunda ordem e resolvido por algoritmos gerais como método de pontos interiores. Contudo, em cenários de grande escala, a velocidade de convergência pode ser lenta (TAO; BOLEY; ZHANG, 2016). Por conta de suas características, existem algoritmos que exploram a estrutura específica desse problema e foram desenvolvidos especialmente para a sua solução (2.11), ou possuem modificações para o lasso. Especificamente, tem-se um enfoque maior no ADMM, dado que este é relevante no próximo capítulo do presente trabalho.

Direção Alternada de Multiplicadores

O ADMM é um algoritmo para solução de problemas de otimização que tem como ideia partir o problema original em subproblemas menores mais fáceis de serem resolvidos. ADMM faz isso efetuando uma decomposição do problema, e então os subproblemas são resolvidos de forma coordenada com o intuito de resolver o problema original. Sua teoria pode ser vista como uma combinação dos benefícios da decomposição dual, e das propriedades de convergência de métodos de Lagrangeana Aumentada para problemas restritos. A descrição do método, uma extensa revisão de literatura, extensões e aplicações podem ser encontrada em Boyd et al. (2011), sendo esta a referência principal para a escrita da presente seção.

O algoritmo resolve problemas da forma

$$\begin{aligned} \text{minimizar} \quad & f(x) + g(z) \\ \text{sujeito a} \quad & Ax + Bz = c \end{aligned} \tag{2.13}$$

em que as variáveis são $x \in \mathbb{R}^n$ e $z \in \mathbb{R}^m$, e $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$ e $c \in \mathbb{R}^p$. A variável de interesse é dividida em duas partes, denominadas x e z, com a função objetivo sendo separável nesta divisão da variável.

A Lagrangeana aumentada clássica relacionada ao problema (2.13) é dada por

$$L_\rho(x, z, w) = f(x) + g(z) + w^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|^2.$$

As iterações do método de ADMM consistem em

$$x^{k+1} := \underset{x}{\operatorname{argmin}}\, L_p(x, z^k, w^k), \tag{2.14}$$

$$z^{k+1} := \underset{x}{\operatorname{argmin}}\, L_p(x^{k+1}, z, w^k), \tag{2.15}$$

$$w^{k+1} := w^k + \rho(Ax^{k+1} + Bz^{k+1} - c), \tag{2.16}$$

em que $\rho > 0$. O algoritmo consiste em uma etapa de minimização de x, uma etapa de minimização z e uma atualização da variável dual. De forma similar ao método de multiplicadores, a atualização da variável dual usa um tamanho de passo igual ao parâmetro de penalidade ρ da Lagrangeana aumentada.

Observa-se pelas iterações em (2.14) e (2.15) que as variáveis x e z são atualizadas de forma alternada, o que explica o termo direção alternada no método ADMM.

O fato da minimização de x e z ocorrer cada uma em uma etapa diferente do método é o que permite que a função objetivo original do problema seja decomposta em diferentes partes f e g se esta for separável.

Pelas características do método, o ADMM é adequado para classes de problemas que podem ser divididos em diferentes partes f e g, que então são resolvidas separadamente. Este é o caso de diversos métodos da literatura, especialmente problemas regularizados, que envolvem na função objetivo mais um termo de regularização.

Para o caso da regularização ℓ_1, tem-se que o termo é não diferenciável em zero. O ADMM adequa-se naturalmente a este problema, o que possibilita a separação da função em uma parte diferenciável e outra não diferenciável. Esta partição traz uma vantagem computacional, como explicitado a seguir.

Para este método, o problema do lasso é expresso em forma equivalente a (2.11) como

$$\begin{aligned} \text{minimizar} \quad & \frac{1}{2}\|y - X\beta\|^2 + \lambda\|\theta\|_1 \\ \text{sujeito a} \quad & \beta - \theta = 0, \end{aligned} \tag{2.17}$$

em que $\beta \in \mathbb{R}^p$ e $\theta \in \mathbb{R}^p$. A variável θ aparece na função objetivo somente no termo não diferenciável da norma 1. Restrições de igualdade são incluídas para garantir que os vetores de solução entregues pela otimização em cada bloco de dados concordem entre si na convergência.

A Lagrangeana aumentada do problema (2.17) é escrita como

$$L_\rho(\beta, \theta, \mu) = \frac{1}{2}\|y - X\beta\|^2 + \lambda\|\theta\|_1 + \mu^T(\beta - \theta) + \frac{\rho}{2}\|\beta - \theta\|^2. \tag{2.18}$$

Para atualização de β é necessário resolver o problema

$$\beta^{k+1} = \underset{\beta}{\operatorname{argmin}}\, L_\rho(\beta, \theta, \mu).$$

Fazendo o gradiente de L em função de β igual a zero, obtém-se

$$\begin{aligned}
\nabla_\beta L_\rho(\beta, \theta, \mu) &= -X^T(y - X\beta) + \mu + \rho(\beta - \theta) \\
0 &= -X^T(y - X\beta) + \mu + \rho(\beta - \theta) \\
0 &= -X^T y + X^T X\beta + \mu + \rho\beta - \rho\theta \\
X^T X\beta + \rho\beta &= X^T y + \rho\theta - \mu \\
(X^T X + \rho I)\beta &= X^T y + \rho\theta - \mu \\
\beta &= (X^T X + \rho I)^{-1}(X^T y + \rho\theta - \mu).
\end{aligned} \tag{2.19}$$

A atualização de θ é obtida ao minimizar (2.18) em relação a θ,

$$\theta^{k+1} = \underset{\theta}{\operatorname{argmin}} \quad L_\rho(\beta, \theta, \mu).$$

A solução deste problema está relacionada à proposição 2.8, em que o operador de soft thresholding pode ser aplicado elemento a elemento. Seja

$$h(\theta) = \frac{1}{2}\|y - X\beta\|^2 + \lambda\|\theta\|_1 + \mu^T(\beta - \theta) + \frac{\rho}{2}\|\beta - \theta\|^2.$$

Fazendo o gradiente de h em função de cada elemento θ_j tem-se:

$$\nabla h(\theta_j) = \begin{cases} \nabla h_1(\theta_j) = \lambda - \mu_j - \rho(\beta_j - \theta_j), & \theta_j \geq 0 \\ \nabla h_2(\theta_j) = -\lambda - \mu_j - \rho(\beta_j - \theta_j), & \theta_j < 0 \end{cases}$$

para $j = 1, \ldots, p$.

Se o mínimo for obtido em $\theta_j > 0$, tem-se que

$$\begin{aligned}
\nabla h_1(\theta_j) &= \lambda - \mu_j - \rho(\beta_j - \theta_j) \\
0 &= \lambda - \mu_j - \rho\beta_j + \rho\theta_j \\
\rho\theta_j &= \rho\beta_j + \mu_j - \lambda \\
\theta_j &= \beta_j + \frac{\mu_j}{\rho} - \frac{\lambda}{\rho}.
\end{aligned}$$

Como $\theta_j > 0$, então $\beta_j + \frac{\mu_j}{\rho} - \frac{\lambda}{\rho} > 0$, o que implica que $\beta_j + \frac{\mu_j}{\rho} > \frac{\lambda}{\rho}$. Dessa forma, se $\beta_j + \frac{\mu_j}{\rho} > \frac{\lambda}{\rho}$ for satisfeito, tem-se $\theta_j = \beta_j + \frac{\mu_j}{\rho} - \frac{\lambda}{\rho}$ como minimizador.

Se o mínimo for obtido em $\theta_j < 0$, ao fazer o gradiente de h_2 em função de θ_j igual a zero obtém-se $\theta_j = \beta_j + \frac{\mu_j}{\rho} + \frac{\lambda}{\rho}$. Logo, se $\theta_j < 0$, infere-se que $\beta_j + \frac{\mu_j}{\rho} < -\frac{\lambda}{\rho}$ e o mínimo é $\theta_j = \beta_j + \frac{\mu_j}{\rho} + \frac{\lambda}{\rho}$.

Para $-\frac{\lambda}{\rho} < \beta_j + \frac{\mu_j}{\rho} < \frac{\lambda}{\rho}$, o mínimo está no único ponto de não diferenciabilidade de $h(\theta_j)$, que é $\theta_j = 0$. Logo a atualização de θ_j é dada por

$$\theta_j^{k+1} = \begin{cases} \beta_j + \frac{\mu_j}{\rho} - \frac{\lambda}{\rho}, & \text{se } \beta_j + \frac{\mu_j}{\rho} > \frac{\lambda}{\rho} \\ 0, & \text{se } -\frac{\lambda}{\rho} < \beta_j + \frac{\mu_j}{\rho} < \frac{\lambda}{\rho} \\ \beta_j + \frac{\mu_j}{\rho} + \frac{\lambda}{\rho}, & \text{se } \beta_j + \frac{\mu_j}{\rho} < -\frac{\lambda}{\rho} \end{cases}$$

que é o operador de soft thresholding $\mathcal{S}_{\lambda/\rho}(\beta_j + \mu_j/\rho)$.

A atualização de μ segundo o método ADMM é dada por (2.16). No caso do lasso, as matrizes A e B são a identidade e o c é um vetor de zeros. Dessa forma, as atualizações do método ADMM para o método de regularização do lasso são

$$\begin{aligned} \beta^{k+1} &= (X^T X + \rho I)^{-1}(X^T y + \rho\theta^k - \mu^k) \\ \theta^{k+1} &= \mathcal{S}_{\lambda/\rho}(\beta^{k+1} + \mu^k/\rho) \\ \mu^{k+1} &= \mu^k + \rho(\beta^{k+1} - \theta^{k+1}). \end{aligned} \tag{2.20}$$

em que $\rho > 0$ é um parâmetro pequeno, $\mathcal{S}_\kappa$ é o operador de *soft thresholding* e $\mu \in \mathbb{R}^p$ é o vetor de multiplicadores de Lagrange associada à restrição.

Neste caso, a iteração mais difícil do método refere-se à atualização de β, dado por (2.20). A atualização de θ envolve somente a aplicação do operador de *soft thresholding*, e a atualização de μ envolve apenas operações entre vetores. Pela forma de (2.20) pode-se interpretar que o método de ADMM para o lasso é análogo a resolver uma sequência de problemas de minimização de regressão ridge.

Segundo Hastie, Tibshirani e Wainwright (2015), as vantagens da estrutura do ADMM incluem o fato que problemas convexos com restrições não diferenciáveis podem ser tratados pela separação das variáveis em β e θ, e a sua capacidade de quebrar um problema grande em partes menores. Para conjuntos de dados com grande número de observações, pode-se ainda dividir os dados em blocos e a otimização é feita sobre cada bloco.

Critério de Parada Conforme descrito em Boyd et al. (2011), durante as suas iterações, o ADMM possui dois resíduos referentes a iteração primal e dual, dadas

respectivamente por

$$\begin{aligned} r^{k+1} =& \beta^{k+1} - \theta^{k+1} \\ s^{k+1} =& -\rho(\theta^{k+1} - \theta^{k}). \end{aligned}$$

Ambos convergem para zero conforme o número de iterações aumenta, então um critério de parada razoável para o método é que estes resíduos sejam pequenos, de forma que $\|r^{k+1}\| \leq \epsilon_1$ e $\|s^{k+1}\| \leq \epsilon_2$. Em que ϵ_1 e ϵ_2 são tolerâncias definidas por

$$\begin{aligned} \epsilon_1 =& \sqrt{p}\epsilon^{abs} + \epsilon^{rel} \max\{\|\beta^{k+1}\|, \|\theta^{k+1}\|\} \\ \epsilon_2 =& \sqrt{p}\epsilon^{abs} + \epsilon^{rel}\|\mu^{k+1}\| \end{aligned}$$

em que $\epsilon^{abs} > 0$ é uma tolerância absoluta e $\epsilon^{rel} > 0$ é uma tolerância relativa. Estes valores foram adotados como $\epsilon^{rel} = 10^{-2}$ e $\epsilon^{abs} = 10^{-4}$.

2.5 ELASTIC NET

O método numérico denominado de elastic net proposto por Zou e Hastie (2005) foi criado para compensar uma propriedade do lasso, a saber, o fato de não lidar muito bem com variáveis altamente correlacionadas. Em uma situação onde existe um certo grupo de variáveis que possuem alta correlação entre si, o lasso tem a tendência de selecionar apenas uma variável do grupo sem um critério de qual delas é selecionada. Nesta situação de multicolinearidade, uma regularização como regressão ridge poderia auxiliar.

Como o seu nome sugere, a elastic net combina propriedades notáveis das duas técnicas de regularização apresentadas anteriormente: a regressão ridge e o lasso, como em uma rede elástica que captura ambas as regularizações. Sendo assim, a elastic net é capaz de fazer seleção de variáveis e encolhimento dos coeficientes, e pode selecionar grupos de variáveis correlacionadas.

A combinação das técnicas se dá pela utilização das normas ℓ_1 e ℓ_2 em conjunto, sendo realizada por meio da solução do problema (2.21) a seguir

$$\underset{\beta \in \mathbb{R}^p}{\text{minimizar}} \quad f_{\text{E}}(\beta) = \frac{1}{2}\|y - X\beta\|^2 + \lambda_1\|\beta\|_1 + \frac{\lambda_2}{2}\|\beta\|^2. \tag{2.21}$$

em que λ_1 e λ_2 são parâmetros de regularização não negativos.

Seja $\alpha = \lambda_2/(\lambda_1 + \lambda_2)$, uma forma equivalente de escrever este problema, é

$$\begin{aligned} \underset{\beta \in \mathbb{R}^p}{\text{minimizar}} \quad & \frac{1}{2}\|y - X\beta\|^2 \\ \text{sujeito a} \quad & (1-\alpha)\|\beta\|_1 + \alpha\|\beta\|^2 \leq t, \end{aligned}$$

para algum t. A função $(1-\alpha)\|\beta\|_1 + \alpha\|\beta\|^2$ é a penalização do elastic net, que é uma combinação convexa das penalidades da regressão ridge e lasso. Dessa forma, quando $\alpha = 1$, tem-se a regressão ridge e $\alpha = 0$, o lasso.

Segundo Zou e Hastie (2005), resolver o problema (2.21) é equivalente à um problema de minimização do lasso. Portanto, espera-se que os algoritmos que são eficientes na solução de problemas lasso sejam utilizados para a solução do problema (2.21).

Lema 2.9. Dado um conjunto de dados (X, y) e parâmetros (λ_1, λ_2), define-se um conjunto de dados artificial $(\widetilde{X}, \widetilde{y})$ como

$$\widetilde{X}_{(n+p)\times p} = (1+\lambda_2)^{-1/2}\begin{pmatrix} X \\ \sqrt{\lambda_2} I \end{pmatrix}, \qquad \widetilde{y}_{(n+p)} = \begin{pmatrix} y \\ 0 \end{pmatrix}.$$

Sejam $\gamma = \lambda_1/\sqrt{1+\lambda_2}$ e $\widetilde{\beta} = \sqrt{1+\lambda_2}\,\beta$. Então pode-se reescrever o problema (2.21) em uma forma equivalente com dados aumentados como:

$$\underset{\widetilde{\beta}\in\mathbb{R}^p}{\text{minimizar}} \quad \|\widetilde{y} - \widetilde{X}\widetilde{\beta}\|_2^2 + \gamma\|\widetilde{\beta}\|_1. \tag{2.22}$$

Seja $\widetilde{\beta}_E^*$ a solução do problema (2.22), então

$$\hat{\beta}_E = \frac{1}{\sqrt{1+\lambda_2}}\widetilde{\beta}_E^*.$$

O Lema 2.9 descreve que o problema da elastic net pode ser convertido em um problema do lasso equivalente quando considera-se um conjunto de dados aumentados. Dessa forma, mostra que o problema de otimização da elastic net pode ser solucionado por métodos que resolvem o lasso, e também é capaz de realizar a seleção automática de variável de maneira similar ao lasso.

2.6 VALIDAÇÃO CRUZADA

A *Cross Validation* (CV) é um conjunto de técnicas utilizada para medir a performance e a capacidade de generalização de um modelo. A ideia de métodos de CV é medir o erro de um modelo em dados que não foram utilizados ao ajustar o modelo, pois o erro é naturalmente menor para os dados utilizados no ajuste ou minimização do problema (JAMES et al., 2014). Os dados originais geralmente são divididos em duas partes: o conjunto de treinamento e o conjunto de teste. O conjunto de treinamento é utilizado apenas para estimar os parâmetros do modelo, enquanto que o conjunto de teste é utilizado apenas para avaliar a capacidade preditiva do modelo estimado.

Ao dividir o conjunto de dados em conjuntos de treinamento e teste, pode-se avaliar o quão bem o modelo generaliza para dados novos e nunca vistos antes. O conjunto de teste serve como um substituto para dados novos do mundo real e sua avaliação fornece uma estimativa do desempenho do modelo. Dado que avaliar o desempenho de um modelo usando os dados de treinamento pode levar à medidas de desempenho otimistas, o conjunto de teste fornece então uma medida imparcial do desempenho real do modelo.

Esta técnica pode ser utilizada também para evitar o *overfitting*, que ocorre quando o modelo tem um desempenho excelente nos dados de treinamento, mas não consegue generalizar para dados novos. Com a utilização de um conjunto de teste mutuamente exclusivo dos dados de treinamento, pode-se identificar se o modelo pode de fato ser utilizado para fazer previsões do fenômeno descrito no modelo.

Existem diferentes formas e ténicas de CV, como *Holdout Cross-Validation*, *Leave-One-Out Cross-Validation* (LOOCV), *K-Fold Cross-Validation* e outros. Neste trabalho, utiliza-se o *K-Fold* CV para as métricas de desempenho.

Ele recebe esse nome porque o conjunto de dados original foi dividido em K conjuntos com aproximadamente o mesmo número de elementos mutuamente exclusivos, de modo que uma dessas K partes (chamados de *folds*) foi usada como conjunto de teste e as partes $K-1$ restantes compõem o conjunto de treinamento. Assim, o modelo de regressão foi ajustado e uma medida de erro foi avaliada. Esse processo foi repetido K vezes até que cada um dos K *folds* fosse usado como conjunto de teste uma vez. Neste momento, tem-se K valores da métrica de erro escolhida. O desempenho do modelo final foi resumido através do valor médio das medidas de erro em cada *fold*. Quando o CV é realizado com $K = 1$, os resultados dependem fortemente da divisão aleatória do conjunto de treinamento e teste. Resolvendo o mesmo problema K vezes e tirando uma média, obtém-se um valor mais robusto que representa o desempenho do algoritmo.

No contexto deste trabalho, a aplicação e uso do *K-fold* CV podem ser resumidas de duas formas:

1. Avaliando o desempenho do modelo. A CV fornece uma estimativa de quão bem o modelo funciona em dados não vistos. Isso ajuda a entender a capacidade de generalização do modelo além dos dados de treinamento. A utilização deste método na avaliação da performance também se justifica em parte a prescindir simulações intensas com dados gerados aleatoriamente. Constata-se que os resultados apresentados no Capítulo 4 foram obtidos em um processo de *K-fold* CV.

2. Escolha de parâmetros. A CV é utilizada como uma das etapas do processo de escolha dos parâmetros. Diversos métodos da literatura possuem parâmetros e os métodos apresentados até então também possuem parâmetros de regularização como o λ que precisam ser estimados separadamente. A CV pode ser utilizada para avaliar os parâmetros que produzem o melhor desempenho do modelo de acordo com uma certa métrica. Nesse caso, uma lista de λ_i candidatos são escolhidos, utiliza-se cada um destes para ajustar um modelo. O desempenho de cada modelo é avaliado por um *K-fold* CV e o modelo que obteve menor valor na métrica de erro é escolhido, e o λ_i do melhor modelo é adotado.

Neste capítulo, foram explorados as técnicas de regressão ridge, lasso e elastic net, bem como o conceito do operador proximal e a importância da validação cruzada. Demonstrou-se como esses métodos são utilizados para lidar com problemas de multicolinearidade, redução de dimensionalidade e seleção de variáveis em modelos de regressão. O entendimento dessas metodologias é essencial para a próxima seção.

3 MÉTODO PROPOSTO

O presente capítulo tem como escopo a apresentação e análise da principal contribuição deste trabalho, o que consiste em uma proposta de algoritmo de ponto fixo para encontrar a solução de problemas de mínimos quadrados regularizados ou problemas de regressão ridge. Posteriormente, são abordados os detalhes do desenvolvimento do algoritmo, seguidos por uma demonstração formal de sua convergência.

Adicionalmente, são expostas possíveis aplicações do método para a solução de diferentes modelos que utilizam a regularização lasso e elastic net. Isto é feito por meio de outro algoritmo de otimização apresentado anteriormente, o ADMM. Conforme visto na Seção 2.4.1, as iterações do ADMM envolvem um passo de atualização da variável primária que se assemelha a resolver um problema de regressão ridge, o que torna o algoritmo proposto neste trabalho uma opção viável em potencial para solução deste problema, observando as modificações necessárias para as características específicas dele.

3.1 REGRESSÃO RIDGE

Considere o seguinte problema de mínimos quadrados,

$$\underset{\beta}{\text{minimizar}} \quad f(\beta) = \frac{1}{2}||y - X\beta||^2. \tag{3.1}$$

Quando X tem posto cheio, a solução de (3.1) é dada por (2.3)

$$\hat{\beta} = (X^T X)^{-1} X^T y. \tag{3.2}$$

Porém, quando $X^T X$ é mal condicionada, tem-se que $(X^T X)^{-1}$ é instável. Quando X não tem posto cheio, $X^T X$ também não é inversível. A fim de mitigar estes problemas, considera-se então a regressão ridge,

$$\underset{\beta}{\text{minimizar}} \quad f_R(\beta) = \frac{1}{2}||y - X\beta||^2 + \frac{\lambda}{2}||\beta||^2. \tag{3.3}$$

Pela condição necessária de otimalidade de primeira ordem, se β^* for um minimizador local de (3.3), então este satisfaz $\nabla f(\beta^*)_{\mathrm{R}} = 0$. Considerando também λ como um valor fixo e $\lambda > 0$, procura-se β tal que $\nabla f(\beta)_{\mathrm{R}} = 0$, ou seja:

$$\begin{aligned}
\nabla f(\beta)_{\mathrm{R}} &= -X^T(y - X\beta) + \lambda\beta \\
0 &= -X^T(y - X\beta) + \lambda\beta \\
\lambda\beta &= X^T(y - X\beta) \\
\beta &= \frac{1}{\lambda} X^T(y - X\beta).
\end{aligned}$$

Denotando-se

$$\phi(\beta) = \frac{1}{\lambda} X^T (y - X\beta), \tag{3.4}$$

tem-se que β representa um ponto fixo para ϕ e, naturalmente, produz um algoritmo iterativo. Dessa forma é possível introduzir um algoritmo baseado em ponto fixo, ou seja, definindo-se

$$z = y - X\beta$$

e escolhendo $x^0 = \beta^0$ arbitrário como ponto de partida, pode-se calcular o z^0 inicial e um processo iterativo, envolvendo apenas operações com matrizes, poderia ser definido como

$$z^0 = y - Xx^0,$$
$$x^{k+1} = \frac{1}{\lambda} X^T z^k,$$
$$z^{k+1} = y - Xx^{k+1}.$$

Este processo é descrito de forma estruturada no Algoritmo 1.

Algoritmo 1: Algoritmo de gradiente com ponto fixo

Dados: $X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, $x^0 \in \mathbb{R}^p$

Parâmetros: $\lambda > \|X^T X\|$;

Inicialize $k = 0$;

enquanto *critério de parada não for atendido* **faça**

- $z^k = y - Xx^k$;
- $x^{k+1} = \frac{1}{\lambda} X^T z^k$;
- $k = k + 1$

fim

Assim, define-se o processo iterativo no Algoritmo 1 como o método do gradiente com ponto fixo (GFP) para solução do problema (3.3). Cabe notar que este algoritmo possui um conceito simples e é de fácil implementação, além de empregar somente operações entre matrizes e vetores em sua execução.

A seguir, discute-se alguns resultados que descrevem como o algoritmo proposto converge para a solução e como o parâmetro λ pode ser determinado e escolhido.

Teorema 3.1. *Dado $x^0 \in \mathbb{R}^n$ e a sequência $\{x^k\}_{k \in \mathbb{N}}$ gerada pelo Algoritmo 1. Se $\lambda > ||X^T X||$, então $\{x^k\}$ converge.*

Demonstração. Retomando a função de iteração ϕ em (3.4), pode-se escrever a expressão em termos de uma sequência de pontos, denotando-se

$$x^{k+1} = \phi(x^k) = \frac{1}{\lambda} X^T (y - Xx^k)$$
$$x^k = \phi(x^{k-1}) = \frac{1}{\lambda} X^T (y - Xx^{k-1}).$$

Fazendo $x^{k+1} - x^k$, tem-se

$$\begin{aligned} x^{k+1} - x^k &= \frac{1}{\lambda}X^T y - \frac{1}{\lambda}X^T X x^k - \left[\frac{1}{\lambda}X^T y - \frac{1}{\lambda}X^T X x^{k-1}\right] \\ x^{k+1} - x^k &= \frac{1}{\lambda}X^T X(x^{k-1} - x^k) \\ ||x^{k+1} - x^k|| &= \frac{1}{\lambda}||X^T X(x^{k-1} - x^k)|| \\ ||x^{k+1} - x^k|| &\leq \frac{1}{\lambda}||X^T X||\,||x^{k-1} - x^k||. \end{aligned}$$

Considerando

$$\begin{aligned} k = 1: \qquad & ||x^2 - x^1|| \leq \frac{1}{\lambda}||X^T X||\,||x^0 - x^1|| \\ k = 2: \qquad & ||x^3 - x^2|| \leq \frac{1}{\lambda}||X^T X||\,||x^1 - x^2|| \leq \left(\frac{1}{\lambda}||X^T X||\right)^2 ||x^0 - x^1||. \end{aligned}$$

Dessa forma, tem-se que

$$||x^{k+1} - x^k|| \leq \left(\frac{1}{\lambda}||X^T X||\right)^k ||x^0 - x^1||$$

para todo $k = 1, 2, \ldots$, com $k \in \mathbb{N}$.

Como $x^0, x^1 \in \mathbb{R}^p$ são finitos e $\lambda > ||X^T X||$, quando $k \to \infty$ tem-se

$$\left(\frac{1}{\lambda}||X^T X||\right)^k \to 0$$

e consequentemente

$$||x^{k+1} - x^k|| \to 0.$$

Na próxima etapa desta demonstração, mostraremos que a sequência $\{x^k\}$ é uma sequência de Cauchy. Dado $q \in \mathbb{N}^*$,

$$x^{k+q+1} = \frac{1}{\lambda}X^T(y - Xx^{k+q}).$$

Fazendo $x^{k+q+1} - x^{k+1}$, tem-se

$$\begin{aligned} x^{k+q+1} - x^{k+1} &= \frac{1}{\lambda}X^T(y - Xx^{k+q}) - \frac{1}{\lambda}X^T(y - Xx^k) \\ &= -\frac{1}{\lambda}X^T X\left[x^{k+q} - x^k\right] \\ &= -\frac{1}{\lambda}X^T X\left[\frac{1}{\lambda}X^T(y - Xx^{k+q-1}) - \frac{1}{\lambda}X^T(y - Xx^{k-1})\right] \\ &= -\frac{1}{\lambda}X^T X\left[-\frac{1}{\lambda}X^T X(x^{k+q-1} - x^{k-1})\right] \\ &= \left(\frac{1}{\lambda}X^T X\right)^2\left(x^{k+q-1} - x^{k-1}\right). \end{aligned}$$

Prosseguindo o desenvolvimento, após q iterações, tem-se

$$x^{k+q+1} - x^{k+1} = (-1)^k \left(\frac{1}{\lambda} X^T X\right)^{k+1} (x^1 - x^0)$$
$$\|x^{k+q+1} - x^{k+1}\| \leq \left(\frac{1}{\lambda}\|X^T X\|^{k+1}\right) \|x^1 - x^0\| \to 0.$$

Logo, $\{x^k\}_{k\in\mathbb{N}}$ é uma sequência de Cauchy e portanto é convergente. ■

Uma condição do Teorema 3.1 é que o parâmetro de regularização $\lambda > \|X^T X\|$. Na implementação do método não foi necessário calcular $\|X^T X\|$, em vez disso utiliza-se a propriedade de submultiplicatividade das normas p induzidas, sendo

$$\|AB\|_p \leq \|A\|_p \|B\|_p, \tag{3.5}$$

em que $\|\cdot\|_p$ representa a norma p induzida.

Dessa forma, para computar λ toma-se

$$\lambda \geq \|X^T\|_p \|X\|_p = \|X\|_p^2.$$

Então, é possível tomar

$$\lambda \geq \|X\|_2^2,$$

em que $\|\cdot\|_2$ representa a norma 2 induzida ou norma espectral.

3.2 LASSO

A regularização com a norma ℓ_1 consiste no método lasso. Como desenvolvido no capítulo anterior, o ADMM aplicado à esta regularização possui em uma das atualizações da variável de suas iterações a solução de um problema de regressão ridge como dado em (2.20). Assim, o método proposto pode ser aplicado para a solução do subproblema do ADMM, e o intuito é utilizá-lo na solução de problemas de regularização de norma ℓ_1.

Partindo da expressão de atualização em (2.20), pode-se escrevê-la em um formato de algoritmo de ponto fixo, de forma semelhante ao realizado para GFP:

$$\begin{aligned}
\beta &= (X^T X + \rho I)^{-1}(X^T y + \rho\theta - \mu) \\
(X^T X + \rho I)\beta &= (X^T y + \rho\theta - \mu) \\
X^T X\beta + \rho\beta &= X^T y + \rho\theta - \mu \\
-X^T(y - X\beta) + \rho\beta &= \rho\theta - \mu \\
\beta &= \frac{1}{\rho} X^T(y - X\beta) + \theta - \frac{1}{\rho}\mu.
\end{aligned}$$

Então, define-se as iterações:

$$\beta^{k+1} = \frac{1}{\rho} X^T z^k + \theta^k - \frac{1}{\rho}\mu^k \tag{3.6}$$
$$z^{k+1} = y - X\beta^{k+1}. \tag{3.7}$$

O algoritmo GFP será utilizado para determinar β^{k+1} dado em (3.7). Embora a sequência seja gerada de maneira semelhante àquela produzida pelo Algoritmo GFP, uma demonstração formal poderia ser produzida para provar a convergência desse caso específico, não sendo apresentado no presente trabalho. Como veremos no próximo capítulo, nos experimentos numéricos realizados, não foram observados problemas de convergência.

No algoritmo GFP, é necessário que $\rho > \|X^T X\|$, que foi reduzido para $\rho > \|X\|^2$ como mencionado em (3.5). Segundo Boyd et al. (2011), para o ADMM ρ geralmente é um parâmetro relativamente pequeno. Neste caso, se o valor de ρ for pequeno, ele não satisfaz a condição do método GFP, ao mesmo tempo, se fizermos $\rho > \|X\|^2$ o ADMM leva mais iterações para convergência, de acordo com testes numéricos realizados inicialmente.

Uma alternativa proposta no presente trabalho diante deste problema é considerar um problema equivalente que é múltiplo do problema original. Considere uma constante $c^2 \in \mathbb{R}$ que multiplica a função objetivo do lasso em sua formulação para o ADMM como em (2.17).

$$\begin{aligned} h(\beta, \theta) &= \frac{1}{2} c^2 \|y - X\beta\|_2^2 + c^2 \lambda \|\theta\|_1 \\ &= \frac{1}{2} \|cy - cX\beta\|_2^2 + c^2 \lambda \|\theta\|_1. \end{aligned}$$

Chamando $W = cX$, deduz-se que

$$h(\beta, \theta) = \frac{1}{2} \|cy - W\beta\|_2^2 + c^2 \lambda \|\theta\|_1.$$

Almeja-se que $\rho > \|W^T W\|_2$ ou $\rho > \|W\|_2^2$. Dessa forma pode-se escrever,

$$\|W\|^2 = \|cX\|^2 = c^2 \|X\|^2 < \rho.$$

Logo,

$$c < \frac{\sqrt{\rho}}{\|X\|}.$$

Definindo $c = \upsilon \frac{\sqrt{\rho}}{\|X\|}$, em que $\upsilon < 1$ é um parâmetro escolhido, o valor de $\|W\|_2^2$ se torna pequeno de forma que é possível escolher um ρ relativamente pequeno para a utilização do ADMM.

Utilizando esta estratégia, as atualizações do método continuam similares ou idênticas ao problema original. As atualizações do ADMM para o problema equivalente gerado são

$$\begin{aligned} \beta^{k+1} &= (W^T W + \rho I)^{-1} (W^T cy + \rho\theta^k - \mu^k) \\ \theta^{k+1} &= \mathcal{S}_{c^2\lambda/\rho}(\beta^{k+1} + \mu^k/\rho) \\ \mu^{k+1} &= \mu^k + \rho(\beta^{k+1} - \theta^{k+1}). \end{aligned}$$

A diferença entre estas iterações e as iterações em (2.20) é que neste caso utiliza-se a matriz alterada W no lugar de X e a constante c multiplica y na atualização de β. O operador de *soft thresholing* é aplicado utilizando a constante $c^2\lambda/\rho$ em vez de λ/ρ.

Para a aplicação do método proposto no presente trabalho, as iterações do método também são modificadas ligeiramente. Neste caso específico, as iterações se tornam

$$\begin{aligned}\beta^{k+1} &= \frac{1}{\rho}W^T z^k + \theta^k - \frac{1}{\rho}\mu^k \\ z^{k+1} &= cy - W\beta^{k+1}.\end{aligned}$$

Dessa forma, é possível a utilização e aplicação do método GFP também para a solução de problemas do tipo lasso.

3.3 ELASTIC NET

Considerando o fato de que esta técnica de regularização é uma combinação de ambas as regularizações que são utilizadas na regressão ridge e o lasso, e também o fato da aplicabilidade do método proposto para problemas de regularização com a norma ℓ_1, como visto na seção anterior, naturalmente espera-se que uma modificação possa ser sugerida para a regularização elastic net.

Como mencionado anteriormente, o problema que é resolvido para realizar a estimação dos coeficientes de regressão do modelo elastic net pode ser reescrito de forma equivalente como um problema lasso. Então métodos que solucionam o lasso também podem ser utilizados para obter soluções para problemas com elastic net.

Em especial, o ADMM possui em uma de suas iterações a solução de um problema de regressão ridge. Para definir se este método pode ser utilizado para solução de problemas de regularização elastic net, desenvolve-se a seguir as iterações do ADMM para a estrutura do problema.

Primeiramente, reescreve-se o problema (2.21) na forma equivalente como

$$\underset{\beta\in\mathbb{R}^p,\,\theta\in\mathbb{R}^p}{\text{minimizar}} \quad \frac{1}{2}\|y - X\beta\|_2^2 + \frac{\lambda_2}{2}\|\beta\|^2 + \lambda_1\|\theta\|_1 \quad \text{s.a.} \quad \beta - \theta = 0. \tag{3.8}$$

A lagrangeana aumentada do problema (3.8) é definida como:

$$L(\beta, \theta, \mu) = \frac{1}{2}\|y - X\beta\|_2^2 + \frac{\lambda_2}{2}\|\beta\|^2 + \lambda_1\|\theta\|_1 + \mu^T(\beta - \theta) + \frac{\rho}{2}\|\beta - \theta\|^2. \tag{3.9}$$

Para atualização de β é necessário resolver o problema

$$\beta^{k+1} = \underset{\beta}{\text{argmin}} \quad L(\beta, \theta, \mu).$$

Fazendo o gradiente de L em função de β igual a zero, obtém-se

$$\begin{aligned}
\nabla_\beta L(\beta,\theta,\mu) &= -X^T(y - X\beta) + \lambda_2\beta + \mu + \rho(\beta - \theta) \\
0 &= -X^T(y - X\beta) + \lambda_2\beta + \mu + \rho(\beta - \theta) \\
0 &= -X^T y + X^T X\beta + \lambda_2\beta + \mu + \rho\beta - \rho\theta \\
X^T X\beta + \lambda_2\beta + \rho\beta &= X^T y + \rho\theta - \mu \\
(X^T X + (\rho + \lambda_2)I)\beta &= X^T y + \rho\theta - \mu \\
\beta &= (X^T X + (\rho + \lambda_2)I)^{-1}(X^T y + \rho\theta - \mu) \qquad (3.10)
\end{aligned}$$

A atualização de θ é obtida ao minimizar (3.9) em relação a θ,

$$\theta^{k+1} = \underset{\theta}{\operatorname{argmin}} \quad L(\beta,\theta,\mu).$$

Este problema é idêntico ao problema do ADMM aplicado ao lasso, observando-se apenas que agora utiliza-se λ_1 em vez de λ. Então os desenvolvimentos são os mesmos e a atualização será muito semelhante:

$$\theta^{k+1} = \mathcal{S}_{\lambda_1/\rho}(\beta^{k+1} + \mu^k/\rho).$$

Assim, as atualizações do ADMM para o problema elastic net são:

$$\begin{aligned}
\beta^{k+1} &= (X^T X + (\rho + \lambda_2)I)^{-1}(X^T y + \rho\theta - \mu) \\
\theta^{k+1} &= \mathcal{S}_{\lambda_1/\rho}(\beta^{k+1} + \mu^k/\rho) \\
\mu^{k+1} &= \mu^k + \rho(\beta^{k+1} - \theta^{k+1}).
\end{aligned}$$

Chamando $\gamma = \rho + \lambda_2$, utilizando o método proposto para resolver (3.10), deduz-se

$$\begin{aligned}
(X^T X + \gamma I)\beta &= (X^T y + \rho\theta - \mu) \\
X^T X\beta + \gamma\beta &= X^T y + \rho\theta - \mu \\
X^T X\beta - X^T y + \gamma\beta &= \rho\theta - \mu \\
X^T(X\beta - y) + \gamma\beta &= \rho\theta - \mu \\
\gamma\beta &= X^T(y - X\beta) + \rho\theta - \mu \\
\beta &= \frac{1}{\gamma}X^T(y - X\beta) + \frac{\rho}{\gamma}\theta - \frac{1}{\gamma}\mu.
\end{aligned}$$

Dessa forma, é possível definir as iterações do método proposto GFP equivalentes para o problema elastic net como

$$\begin{aligned}
\beta^{k+1} &= \frac{1}{\gamma}X^T z^k + \frac{\rho}{\gamma}\theta^k - \frac{1}{\gamma}\mu^k \\
z^{k+1} &= y - X\beta^{k+1}.
\end{aligned}$$

Seguindo as condições do método proposto, é necessário que

$$\rho + \lambda_2 > \|X^T X\| \iff \lambda_2 > \|X^T X\| - \rho.$$

Na implementação adota-se uma constante $\nu > 1$ e define-se $\lambda_2 = \nu(\|X\|^2 - \rho)$. λ_1 pode ser determinado de forma independente, neste caso utilizou-se o método de CV K-fold.

Dessa forma, o algoritmo de ponto fixo GFP proposto pode ser aplicado também na solução das atualizações de β do ADMM. Sendo assim, aplicado também a problemas de elastic net. Neste caso também ressaltamos que a sequência é gerada de maneira semelhante àquela produzida pelo algoritmo GFP, sendo uma aplicação do algoritmo. Uma demonstração formal poderia ser desenvolvida para garantir a convergência desse caso específico, não sendo apresentado no presente trabalho. Os experimentos numéricos realizados mostram a viabilidade do algoritmo para problemas de elastic net.

No presente capítulo foi apresentado um algoritmo de ponto fixo para solução de problemas do tipo regressão ridge e suas propriedades. Além disso, indiretamente ele pode ser aplicado na solução de problemas como lasso e elastic net, por meio de sua aplicação na solução de um dos subproblemas do ADMM.

4 EXPERIMENTOS NUMÉRICOS

Nesta seção são apresentados os resultados de testes numéricos do algoritmo proposto. Esses experimentos foram projetados com o intuito de compreender como nosso novo método se comporta em diferentes problemas, comparando-o com os métodos tradicionais e validando seu desempenho. Os testes foram executados em um processador Intel(R)® Core(TM) i7-8550U CPU, 1.80GHz, 8.0 GB de memória RAM e implementação em MATLAB® R2016a. O algoritmo foi implementado conforme descrito no Algoritmo 1, e também conforme as adaptações propostas para as técnicas de lasso e elastic net. O critério de parada é dado por $||x^k - x^{k+1}|| \leq \epsilon$, em que $\epsilon = 10^{-6}$ é a tolerância adotada para todos os métodos, ou até o número máximo de iterações de 5000 ser atingido.

4.1 MÉTODOS USADOS NA COMPARAÇÃO

Os métodos utilizados para comparação foram as funções mldivide (MD) do MATLAB®, gradientes conjugados (CG) implementado conforme apresentado por Ribeiro e Karas (2013), e método de mínimos resíduos (MS) (PAIGE; SAUNDERS, 1975). A função MD foi utilizado como benchmark de uma função implementada no próprio software MATLAB® com o intuito de verificar os resultados obtidos. Os métodos CG e MS foram selecionados por serem métodos clássicos em otimização. MS é um método iterativo Krylov para solução de sistemas lineares $Ax = b$ em casos mais gerais do que o CG, já que o MS resolve sistemas lineares para uma matriz A simétrica, enquanto CG considera A simétrica e positiva-definida.

Na segunda parte dos experimentos, envolvendo o problema da regularização com lasso e elastic net, foram utilizados os métodos de MD, CG e a abordagem cholesky-woodbury (CW). A função MD foi mantida como benchmark de uma função implementada no software, CG como um método clássico e a abordagem CW é diferente das demais como uma alternativa a métodos de otimização.

Em problemas de regressão ridge, a solução a ser computada depende do cálculo de $(X^T X + \lambda I)^{-1} X^T Y$. Em um problema em que $p > n$ este cálculo pode ser difícil por conta da alta dimensão de X. Na CW, a inversão da matrix $p \times p$ pode ser evitada de outra forma ao utilizar a identidade de Woodbury. Seja $A \in \mathbb{R}^{p \times p}$, $U \in \mathbb{R}^{p \times n}$, e $V \in \mathbb{R}^{n \times p}$. A forma simplificada da identidade de Woodbury é dada por

$$(A + UV)^{-1} = A^{-1} - A^{-1}U(I_{n \times n} + VA^{-1}U)^{-1}VA^{-1}. \tag{4.1}$$

A aplicação da identidade (4.1) para o estimador de regressão ridge é dada por

$$\begin{aligned}\hat{\beta}_{\mathrm{R}} &= (X^T X + \lambda I_{p\times p})^{-1} X^T Y \\ &= \left[\lambda^{-1} I_{p\times p} - \lambda^{-2} X^T (I_{n\times n} + \lambda^{-1} X X^T)^{-1} X\right] X^T Y \\ &= \lambda^{-1} X^T (I_{n\times n} + \lambda^{-1} X X^T)^{-1} Y.\end{aligned}$$

Dessa forma a inversão da matriz $p \times p$ em $X^T X + \lambda I$ é substituída pela $I_{n\times n} + \lambda^{-1} X X^T$, e a solução desse sistema é obtida por meio da decomposição de Cholesky. A adoção dessa abordagem foi considerada para expandir a variedade de métodos utilizados na comparação nos experimentos. Essa metodologia poderia ter sido igualmente aplicada na fase inicial dos testes, quando consideramos apenas a regressão ridge, não sendo usada apenas por questões de cronologia da pesquisa e condução da pesquisa.

4.2 DATASETS

O primeiro conjunto de problemas utilizados nos testes compreendem as matrizes densas. Estas foram criadas da mesma forma que em (BIERLAIRE; TOINT; TUYTTENS, 1991), sendo geradas baseadas na estrutura de uma decomposição em valores singulares

$$X = UDV^T,$$

em que o logaritmo dos elementos da diagonal de D são gerados uniformemente distribuídos no intervalo $[-\frac{1}{2}\ln\kappa(X), \frac{1}{2}\ln\kappa(X)]$, em que $\kappa(X)$ é o número de condição desejado para X, e as matrizes ortogonais U e V são a matriz Q obtida em uma decomposição QR de matrizes geradas aleatoriamente com elementos entre $[-100, 100]$. Os elementos do vetor da variável resposta y foram gerados por meio da expressão

$$y = X\beta + \epsilon,$$

em que $X \in \mathbb{R}^{n\times p}$ é a matriz do conjunto de dados do problema, β é um vetor gerado aleatoriamente com distribuição uniforme no intervalo $[-25, 25]$ e ϵ é um termo de erro com $\epsilon \sim$ Normal(0, 1).

Para os experimentos numéricos envolvendo o lasso e elastic net, a construção de $\beta \in \mathbb{R}^p$ foi feita considerando que os $p/2$ primeiros elementos são formados por uma sequência repetida dos valores $\{0.25, 0.5, 1\}$ e os $p/2$ elementos finais do vetor são preenchidos com zeros, a fim de simular um modelo esparso.

As matrizes foram geradas com diferentes dimensões e diferentes valores para o número de condição de X. As dimensões das matrizes foram determinadas por meio de diferentes razões $\frac{n}{p} \in \{0.01, 0.1, 0.2, 0.5, 2, 5, 10\}$ para números de linhas n entre 15 e 5000 e número de colunas p entre 10 e 10000, totalizando 30 dimensões diferentes. Os números de condição adotados para a geração das matrizes compreendem sete valores diferentes

$\kappa(X) \in \{10, 10^2, 10^3, 10^5, 10^7, 10^{11}, 10^{15}\}$, sendo adotados de forma a conter números de condição baixos e altos. Considerando quantidade de dimensões e números de condição, no total 210 matrizes foram geradas aleatoriamente.

Como mencionado anteriormente, a técnica de validação cruzada K-fold foi utilizada para avaliar a performance dos modelos no conjunto de teste. Cada um destes 210 problemas foi resolvido 5 vezes por cada método, sendo cada um destes resultados armazenados e sumarizados por uma média. Esta divisão é importante para diminuir a variância nos dados, pois resolver o problema apenas uma vez pode produzir métricas de avaliação com resultado bom ou ruim a depender do acaso.

O segundo conjunto de teste é composto por matrizes de problemas aplicados em diferentes áreas. As matrizes mal condicionadas foram selecionadas dentre as disponíveis no banco de matrizes singulares da San Jose State University (FOSTER, 2021; DAVIS; HU, 2011; HANSEN, 1994).

Ao todo, 129 matrizes reais com número de condição alto foram selecionadas de (FOSTER, 2021), filtradas por serem matrizes reais quadradas ou retangulares, a maior parte delas sendo esparsa. A variável resposta y também é proveniente das bases de dados. Outras informações sobre os problemas podem ser compreendidos na tabela disponível no Anexo I.

Por fim, para ilustrar os resultados, aplica-se o algoritmo proposto no conjunto de dados reais de alta dimensionalidade, especificamente dados genômicos sobre a produção da riboflavina (vitamina B_2) com *Bacillus subtilis*, disponíveis em Bühlmann, Kalisch e Meier (2014), para fins ilustrativos.

4.3 MÉTRICAS DE AVALIAÇÃO

Nesta seção descreve-se as métricas utilizadas para a avaliação e comparação dos métodos. A primeira delas é o tempo de CPU, medida em segundos, com o objetivo de avaliar se os métodos propostos são capazes de resolver os mesmos problemas com menor custo computacional. Esta medida é feita por meio da rotina *tic* do MATLAB®.

O erro quadrático médio (MSE) é utilizado para avaliar a qualidade da previsão do modelo, principalmente para os conjuntos de teste. Este é definido como

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y} - y)^2,$$

em que n é o número de observações sendo avaliadas, $\hat{y}$ é a predição fornecida pelo modelo e y é o valor real observado da variável.

Outra métrica avaliada é o valor da função objetivo (VFO), que neste caso é definida como

$$\text{FO} = \frac{1}{2}||y - X\beta||^2 + \frac{\lambda}{2}||\beta||^2. \tag{4.2}$$

Neste caso, a métrica (4.2) é referente ao problema de regressão ridge, sendo calculada para as técnicas do lasso e elastic net utilizando os valores de suas respectivas funções objetivo. O intuito é avaliar a qualidade do método por meio da qualidade da solução, uma vez que o estimador β ótimo fará com que a VFO tenha o menor valor possível. No contexto de aprendizado de máquina, esta métrica é também chamada de custo.

Por fim, utiliza-se os gráficos de perfil de desempenho como descrito em (DOLAN; MORÉ, 2002) como uma forma compacta de comparar diferentes algoritmos $a \in \mathbb{A}$ em um conjunto de dados $\mathbb{D}$. Seja n_d o número de conjuntos de dados que dão origem a n_d problemas e n_a o número de algoritmos sendo comparados, define-se $c_{d,a}$ como o custo computacional necessário para resolver $d \in \mathbb{D}$ usando a. Tal custo computacional poderia ser definido como o tempo de execução, número de iterações, memória utilizada, entre outras medidas dependendo da aplicação. No presente trabalho utiliza-se as métricas de tempo de CPU, MSE e VFO.

Assim, o desempenho do algoritmo a em um problema d é avaliado mediante um índice de desempenho definido como

$$r_{d,a} = \frac{c_{d,a}}{\min\{c_{d,a} \,|\, a \in \mathbb{A}\}}.$$

Observa-se que este índice é sempre maior ou igual a 1, em que $r_{d,a} = 1$ ocorre quando o custo computacional $c_{d,a}$ para o problema d é o menor dentre todos os algoritmos. O índice de desempenho representa o comportamento de um algoritmo na resolução de um problema, com o intuito de avaliar o desempenho de forma geral é definido

$$\omega_a(\tau) = \frac{1}{n_d}\, card\{d \in \mathbb{D} \,|\, r_{d,a} \leq \tau\},$$

em que $\omega_a(\tau)$ é a probabilidade do algoritmo a resolver um problema d em não mais do que τ vezes o melhor custo levado por qualquer algoritmo de a. Dessa forma, os algoritmos com grande probabilidade $\omega_a(\tau)$ possuem melhor desempenho segundo este método. Portanto, quanto mais próxima da reta $y = 1$ estiver uma curva de desempenho, mais eficiente é o algoritmo. Observa-se que $\omega_a(1)$ corresponde à probabilidade de um algoritmo a obter desempenho melhor do que todos os outros algoritmos. Nos gráficos de desempenho, τ é representado no eixo das abcissas variando entre $[1, r_M)$, em que $r_M = max(r_{d,a})$, enquanto o eixo das ordenadas representam os valores das probabilidades.

4.4 SELEÇÃO DO PARÂMETRO DE REGULARIZAÇÃO

Nesta seção, o valor de λ obtido pelo algoritmo proposto é comparado com diferentes valores de λ encontrados pelo método K-fold CV para o problema (3.3). O método K-fold foi executado para um conjunto de diferentes valores de $\lambda \in \{10^{-5}, 10^{-3}, 10^{-1},$

$0, 1, 10^3, 10^5, 10^7, 10^9, 10^{11}, 10^{15}\}$, e então dentre estes, o λ que obteve o menor MSE no CV foi adotado para cada problema e nomeado como λ_{CV}.

Para o novo algoritmo proposto neste trabalho adota-se que $\lambda = 2,5\|X\|_2^2$. Testes iniciais mostraram que quanto maior a constante adotada menor o tempo de convergência do algoritmo, ao mesmo tempo valores acima de $2,5$ não trouxeram maiores ganhos para o método. Os conjuntos de dados usados para avaliar os diferentes algoritmos foram os mesmos mencionados na Seção 4.2.

TABELA 1 – Número de problemas em que o valor de λ ou λ_{CV} foi maior que o outro, para cada conjunto de problemas.

	Conjunto 1	Conjunto 2
λ	210	71
λ_{CV}	0	58
$\lambda > \lambda_{\text{CV}}$ (%)	100.00	55.03

O primeiro resultado a ser destacado da Tabela 1 foi o fato de que o λ_{CV} escolhido por K-fold foi menor que o λ calculado pelo algoritmo proposto. Para o conjunto 1, tem-se $\lambda > \lambda_{\text{CV}}$ para todos os 210 problemas. Para o conjunto 2, observa-se $55,03\%$ dos problemas com λ sendo maior que λ_{CV}. Por conta disso, em alguns problemas percebe-se que λ_{CV} é menor que $\|X\|_2^2$ e neste caso, o algoritmo proposto neste trabalho não era aplicável para o problema.

Para a técnica de regularização lasso, tem-se a relação dada por $c = \upsilon\frac{\sqrt{\rho}}{\|X\|}$. Experimentos iniciais foram realizados de forma a adotar os parâmetros para cada abordagem. Adotou-se os parâmetros $\upsilon = 0.1$ e $\rho = 0.01$ para os dados gerados, enquanto $\rho = 0.01$ e $\upsilon = 0.1$ para a base de dados de matrizes singulares.

Para a abordagem do elastic net, tem-se dois parâmetros de regularização, correspondendo a regularização da norma ℓ_1 e da norma ℓ_2. Neste caso, tem-se que $\lambda_2 = \nu(\|X\|^2 - \rho)$. Para os dados gerados e para os dados extraídos da base de dados de matrizes singulares, tem-se $\nu = 2$ e $\rho = 10$, depois de definido λ_2, o parâmetro λ_1 é escolhido usando CV.

4.5 EXPERIMENTOS E DISCUSSÕES PARA REGRESSÃO RIDGE

Os resultados para GFP do primeiro conjunto de problemas (210 problemas densos) estão na Tabela 2, 3 e 4 considerando o tempo de execução de diferentes métricas de avaliação, MSE e VFO. O MSE foi medido durante a fase de teste dos problemas, ou seja, apenas as linhas da matriz X que não foram utilizadas na fase de treinamento foram consideradas para o cálculo do MSE.

A matriz de dados X foi padronizada para que cada coluna tenha média 0 e desvio padrão 1, e a variável resposta y foi centralizada. Assim, não há intercepto na estimação

do modelo.

Ao invés de mostrar os valores dos resultados de cada método e cada métrica de desempenho, os resultados são apresentados pelos rankings correspondentes. Para cada método foi dada uma classificação de 1 a 4 onde 1 indica o método que exigiu menor tempo médio de execução (usando k-fold) para resolver um problema e 4 o método mais lento. O mesmo processo foi feito para MSE e VFO, de modo que a posição 1 foi atribuída ao método com o menor valor de MSE ou VFO, e o método com o maior valor recebeu o posto 4. A comparação deve ser feita por linha (problemas agrupados por dimensão da matriz) e blocos de colunas (diferentes métricas de desempenho). A última linha da tabela indica a média da coluna correspondente acima.

Segundo os resultados dos testes, contatou-se que o número de condição de X não influenciou diretamente no ranking final quando considera-se uma dimensão $n \times p$ específica. Portanto, o número de condição de X foi omitido das tabelas e, em vez disso, os valores mostrados em cada linha das Tabelas 2, 3 e 4 são as médias dos ranks recebidos pelos métodos, agrupados pela dimensão da matriz. O tempo de execução de todos os testes foi medido em segundos. A magnitude dos tempos registrados variou de milissegundos a minutos a depender do problema.

TABELA 2 – Ranking médio de GFP e outros métodos para cada dimensão $n \times p$ para razão n/p de $0,01$ e $0,1$.

			Tempo				MSE				VFO			
n/p	n	p	GFP	ML	CG	MS	GFP	ML	CG	MS	GFP	ML	CG	MS
0.01	15	1500	1	4	3	2	1.43	3	3	2.57	2.86	1.29	1.29	2.86
0.01	25	2500	1	4	3	2	2.29	2.43	2.71	2.57	2.86	1.29	1.14	3.43
0.01	30	3000	1	4	3	2	2.14	2	2.57	3.29	2.71	1.29	1	2.86
0.01	50	5000	1	4	3	2	3	2.43	2.14	2.43	2.71	1.14	1	3.57
0.01	100	10000	1	4	3	2	2.86	2.43	2.86	1.86	2.57	1.29	1.29	3.57
0.1	10	100	1.14	4	1.86	3	2.71	2.43	2	2.86	2.29	1.57	1.29	3.43
0.1	50	500	2.14	4	1.57	2.29	2.43	2.43	2.57	2.57	2.29	1.14	1	3.86
0.1	100	1000	1	4	3	2	2.57	2.71	3	1.71	2.43	1.14	1.29	3.86
0.1	500	5000	1.14	4	3	1.86	3	2	1.86	3.14	2.29	1.14	1	3.86
0.1	150	1500	1.86	4	2.71	1.43	3.14	2	2.14	2.71	2	1.29	1.43	3.86
0.1	250	2500	1.57	4	2.86	1.57	1.57	3.29	2.57	2.57	2	1.43	1.29	3.86
0.1	300	3000	1.43	4	2.86	1.71	2.57	2.43	2.43	2.57	2	1.29	1.43	3.86
0.1	1000	10000	1	4	3	2	2.71	2	2.29	3	2.14	1.14	1.14	3.86
Média da coluna			**1.25**	4	2.76	1.99	2.49	**2.43**	2.47	2.6	2.4	1.26	**1.2**	3.59

Para algumas dimensões da matriz, o ranking médio não é um número inteiro, como para os problemas de dimensão 15×1500 para tempo de execução na Tabela 2, em que o ranking de GFP foi 2.86 para VFO, o que significa que as posições do *rank* eram diferentes para alguns números de condição e a média dessas posições produziram valores decimais.

Da Tabela 2, conclui-se que GFP teve classificação média próxima a 1 para tempo

TABELA 3 – Ranking médio de GFP e outros para cada dimensão $n \times p$ para razão n/p de $0,2$ e $0,5$.

			Tempo				MSE				VFO			
n/p	n	p	GFP	ML	CG	MS	GFP	ML	CG	MS	GFP	ML	CG	MS
0.2	1000	5000	1.57	4	2.86	1.57	2.57	2.29	2	3.14	2.43	1.57	1.14	4
0.5	50	100	2.29	4	1	2.71	2	3	2.29	2.71	2.29	1.29	1	4
0.5	250	500	3	4	1.14	1.86	3	2.57	3	1.43	1.86	1.43	1	4
0.5	500	1000	3.14	3.86	2	1	3.14	2.57	2.14	2.14	2	1.43	1.29	4
0.5	2500	5000	3	4	2	1	2.14	2.43	2.57	2.86	2.29	1.29	1.43	4
0.5	750	1500	3.14	3.86	2	1	2.29	2.29	2.29	3.14	2	1.29	1.29	4
0.5	1250	2500	3	4	2	1	2.43	2.43	2.14	3	1.86	1	1.14	4
0.5	1500	3000	3	4	2	1	2	2.71	2.57	2.71	2	1.43	1.29	4
Média da coluna			2.77	3.96	1.88	**1.39**	2.45	2.54	**2.38**	2.64	2.09	1.34	**1.2**	4

TABELA 4 – Ranking médio de GFP e outros para cada dimensão $n \times p$ para razão n/p de $2,5$ e 10.

			Tempo				MSE				VFO			
n/p	n	p	GFP	ML	CG	MS	GFP	ML	CG	MS	GFP	ML	CG	MS
2	100	50	2.29	4	1	2.71	2.71	2.57	2.43	2.29	2	1.57	1.43	3.71
2	500	250	3.14	3.86	1	2	2.71	2.57	2.43	2.29	1.57	1.43	1.43	4
2	1000	500	4	3	1.14	1.86	2.14	2.71	2.71	2.43	2.14	1.29	1.57	4
2	5000	2500	4	3	2	1	2.71	2.43	1.86	3	2	1.29	1	4
5	5000	1000	4	3	2	1	2.43	1.86	2.86	2.86	1.86	1.14	1.14	4
10	100	10	1.71	4	1.29	3	2.71	2.57	2.43	2.29	2	1.57	1.14	2.43
10	500	50	3	4	1	2	2.71	2.29	1.86	3.14	1.86	1.71	1.43	3.57
10	1000	100	3.14	3.86	1	2	1.86	2.71	2.14	3.29	2.14	1	1.57	3.86
10	5000	500	4	3	1.57	1.43	3.57	1.86	2.14	2.43	1.86	1.14	1.29	4
Média da coluna			3.25	3.52	**1.33**	1.89	2.62	2.4	**2.32**	2.67	1.94	1.35	**1.33**	3.73

de execução e foi o mais rápido. Os métodos CG e MS não se superaram e o ML foi o mais lento. Analisando os rankings médios para MSE, não houve método superior porque as médias dos rankings foram todas semelhantes. Observando VFO, percebe-se que ML e CD obtiveram um valor um pouco menor que os outros métodos e MS foi pior nessa métrica.

Esses resultados mostram que GFP tem rank médio de 1 para tempo de execução para matrizes de dimensão $15 \times 15000, 25 \times 2500, 30 \times 3000, 50 \times 5000, 100 \times 10000, 100 \times 1000$, que ou seja, a maioria das dimensões onde $p \gg n$.

A diferença de um problema de mesma dimensão para outro é o número de condição, o que indica que nesses casos o número de condição não afetou tanto os resultados computacionais em tempo de execução. Uma justificativa para esse resultado pode ser o fato de que na regressão de ridge um parâmetro λ é adicionado e, portanto, uma matriz com um número de condição alto também é regularizada por um valor alto de λ, de modo que não seja mal-condicionada.

Para avaliar a diferença entre os resultados obtidos, calcula-se o valor da diferença

relativa entre GFP e ML, CG ou MS, como em (4.3).

$$\text{Relat. Diff.}(c_{\text{método}}, c_{\text{GFP}}) = \frac{c_{\text{método}} - c_{GFP}}{c_{GFP}} \tag{4.3}$$

onde c pode ser uma métrica como tempo de execução, MSE ou VFO.

O tempo de execução, MSE e VFO do método GFP foram usadas como referência para calcular as diferenças relativas para todos os 210 problemas, então uma média foi feita agregando os problemas com o mesmo tamanho. Os resultados são mostrados na Tabela 5. O mesmo processo foi adotado para MSE e VFO, e os resultados também podem ser vistos na Tabela 5. Valores positivos para tempo de execução indicam que os métodos ML, CG ou MS foram mais lentos que GFP. Valores positivos para MSE e VFO significam que ML, CG ou MS tiveram valores mais altos nessas métricas.

TABELA 5 – Média da diferença relativa de GFP para os outros para cada dimensão $n \times p$ em percentual

			Tempo			MSE			VFO		
n/p	n	p	ML	CG	MS	ML	CG	MS	ML	CG	MS
0.01	15	1500	81.63	10.37	8.96	9.07E-09	9.07E-09	1.33E-08	-5.9E-14	-5.9E-14	-3.8E-14
0.01	25	2500	195.81	24.28	18.84	-5.7E-10	-5.7E-10	4.91E-10	-3.1E-14	-3.1E-14	1.5E-14
0.01	30	3000	275.19	33.20	25.87	-5.1E-10	-5.1E-10	5.85E-10	-4.7E-14	-4.7E-14	-1.5E-14
0.01	50	5000	150.20	12.41	8.60	-1.4E-08	-1.4E-08	-1.5E-08	-1.2E-13	-1.2E-13	-8.6E-14
0.01	100	10000	201.88	13.54	8.56	5.9E-10	5.9E-10	-1.6E-09	-4.4E-14	-4.4E-14	-3.7E-14
0.1	1000	10000	22.01	2.04	1.19	-8.5E-10	-8.4E-10	8.08E-10	-3.2E-15	-3.2E-15	1.16E-14
0.1	10	100	16.74	0.21	1.28	-6.2E-09	-6.2E-09	1.19E-09	-3.1E-15	-3.1E-15	4.91E-14
0.1	50	500	7.16	-0.07	0.01	9.11E-10	9.12E-10	3.71E-10	-4E-15	-4E-15	2.54E-14
0.1	100	1000	18.04	2.31	1.94	1.4E-09	1.4E-09	-5.1E-09	-2.5E-15	-2.5E-15	7.26E-15
0.1	150	1500	5.85	0.26	-0.05	-4.9E-10	-4.9E-10	3.24E-10	-3.5E-15	-3.6E-15	1.18E-14
0.1	250	2500	7.38	0.44	0.02	1.75E-09	1.75E-09	1.96E-09	-1.2E-15	-1.2E-15	4.6E-15
0.1	300	3000	7.92	0.47	0.04	-5E-10	-5E-10	9.18E-10	-6.4E-16	-5.9E-16	2.06E-14
0.1	500	5000	11.79	0.91	0.36	-6E-10	-6E-10	1.07E-09	-9.6E-16	-8.8E-16	1.11E-14
0.2	1000	5000	5.81	0.24	-0.05	1.63E-10	1.6E-10	1.34E-09	-5.9E-16	-5.8E-16	1.43E-14
0.5	50	100	7.69	-0.34	0.25	5.31E-10	5.3E-10	4.52E-09	-1.2E-15	-1.2E-15	1.37E-14
0.5	250	500	3.43	-0.29	-0.27	4.07E-10	4.07E-10	-2.1E-09	-2.8E-15	-2.8E-15	1.71E-14
0.5	500	1000	0.41	-0.68	-0.72	3.82E-10	3.82E-10	-1.1E-09	-2.3E-15	-2.2E-15	1.15E-14
0.5	750	1500	0.47	-0.62	-0.69	1.72E-10	1.71E-10	1.28E-09	-1.3E-15	-1.2E-15	1.17E-14
0.5	1250	2500	0.90	-0.47	-0.58	-1.9E-10	-1.9E-10	1.35E-09	-2.2E-15	-2.2E-15	1.33E-14
0.5	1500	3000	1.01	-0.45	-0.56	1.56E-12	1.94E-12	3.8E-10	-1.6E-15	-1.6E-15	1.12E-14
0.5	2500	5000	2.19	-0.16	-0.28	1.33E-10	1.35E-10	1.92E-10	-1.2E-15	-1.2E-15	1.43E-14
2	100	50	6.46	-0.39	0.31	1.47E-09	1.47E-09	-2E-10	-1.5E-15	-1.6E-15	2.72E-14
2	500	250	0.39	-0.78	-0.73	4.97E-10	4.97E-10	2.62E-09	-1.5E-15	-1.5E-15	1.05E-14
2	1000	500	-0.46	-0.87	-0.87	5.03E-10	5.03E-10	7.64E-10	-6.1E-16	-6.2E-16	1.53E-14
2	5000	2500	-0.34	-0.68	-0.70	-2.7E-10	-2.8E-10	3.56E-10	1.98E-17	0.98	1.59E-14
5	5000	1000	-0.76	-0.87	-0.87	5.84E-10	5.86E-10	9.61E-10	-2.9E-17	0.08	1.51E-14
10	100	10	15.88	2.75	3.89	-1.1E-08	-1.1E-08	-1E-08	-4.9E-15	-4.9E-15	1.6E-14
10	500	50	0.83	-0.85	-0.69	-1.7E-09	-1.7E-09	-1.3E-09	-2.9E-15	-2.9E-15	1.02E-14
10	1000	100	0.24	-0.83	-0.76	2.06E-09	2.06E-09	2.73E-09	-7.6E-16	-8.6E-16	1.38E-14
10	5000	500	-0.85	-0.92	-0.92	-1.2E-09	-1.2E-09	-1.3E-09	-1.4E-16	0.064903	1.46E-14

Conclui-se pela Tabela 5 que a solução encontrada por GFP está próxima das soluções de ML, CG e MS, mostrada pela baixa diferença relativa para MSE e VFO. Como o mesmo λ foi usado para GFP, CG, ML e MS, todos os métodos estão resolvendo o mesmo problema, então o fato de a diferença relativa ser pequena indica que todos os métodos

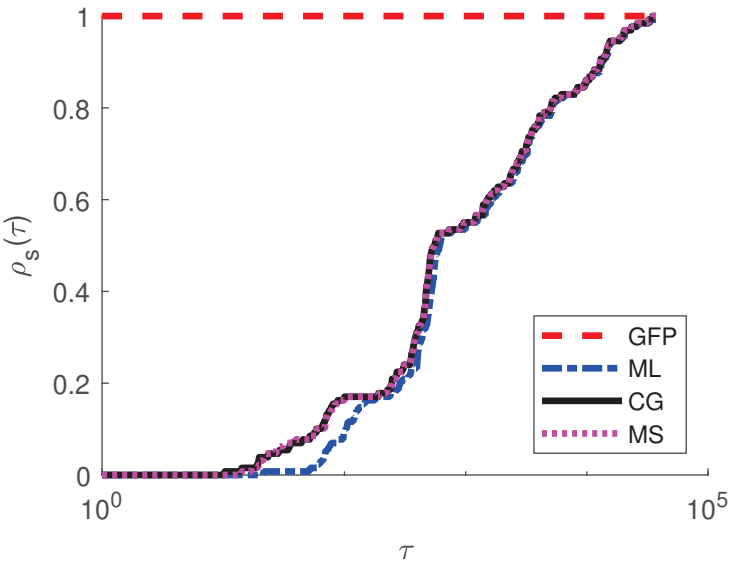


FIGURA 3 – Perfil de desempenho para tempo de execução de GFP, ML, CG e MS

estão obtendo resultados semelhantes em MSE, de modo que GFP funciona como esperado. Mesmo que GFP não tenha a posição mais baixa nas métricas MSE e VFO, constata-se que as diferenças relativas são bem pequenas e os resultados são quase os mesmos.

Quando considera-se o tempo de execução da CPU, concluímos que a diferença de tempo de execução entre GFP e os outros métodos é relevante principalmente nos casos em que a relação n/p é $0,01$ ou $0,1$, onde as diferenças relativas são todas positivas e muitas delas acima de 1. O que indica que GFP obtém resultados semelhantes de MSE e VFO com menos tempo computacional.

Para o conjunto 2, foram consideradas 129 matrizes mal condicionadas esparsas de diferentes dimensões, e não foi possível construir uma tabela compacta como a do conjunto 1. Portanto, gráficos de desempenho foram escolhidos para apresentar os resultados.

O perfil de desempenho relativo ao tempo de execução na Figura 3 mostra que $\rho_1(1) = 0,9689$, o que significa que GFP resolveu $96,89\%$ dos problemas com o menor tempo de execução, enquanto a rotina CG resolveu $3,10\%$ dos problemas mais rapidamente do que os outros métodos. Os métodos ML e MS não produziram o menor tempo de execução para nenhum dos problemas resolvidos.

A robustez dos métodos também pode ser observada pelos valores de $\rho(\tau_{max})$ que indica a porcentagem de problemas que os métodos realmente resolveram. Essa medição foi de $99,22\%$ para todos os métodos, o que significa que todos os métodos encontraram a solução para 127 dos 129 problemas testados. Assim, há evidências numéricas que sugerem que GFP é um algoritmo competitivo ao considerar o tempo de execução, nessa classe de problemas em que o número de condição da matriz é alto e a matriz é esparsa.

Na Figura 4, o MSE medido na fase de teste do modelo apontou que em $31,78\%$ dos problemas, GFP teve o menor MSE, enquanto os outros métodos ML, CG e MS

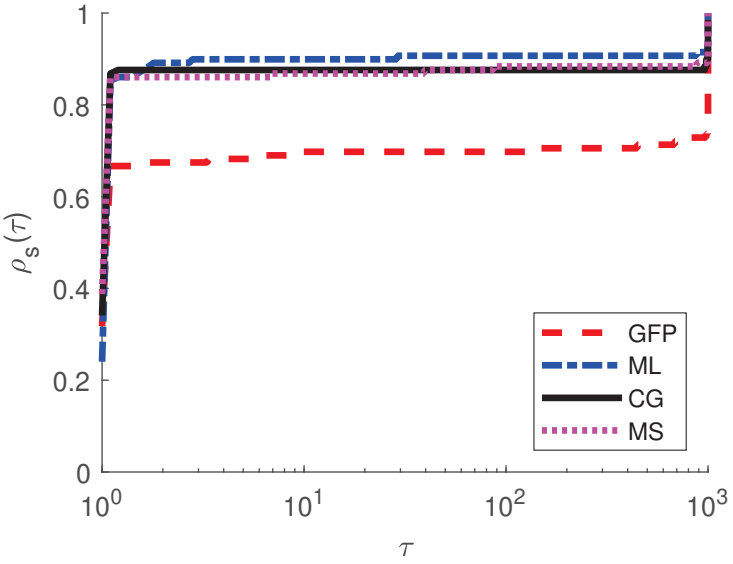


FIGURA 4 – Perfil de desempenho para MSE de GFP, ML, CG e MS

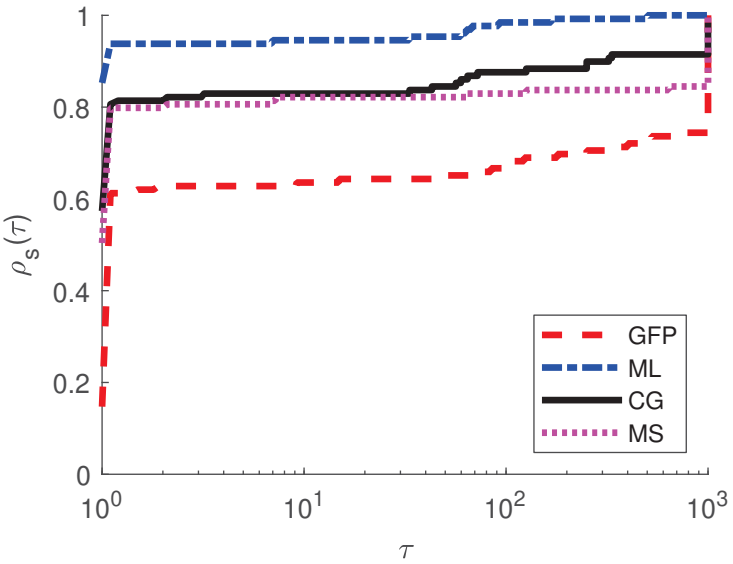


FIGURA 5 – Perfil de desempenho para VFO de GFP, ML, CG e MS

tiveram menor MSE em $24,03\%$, $34,10\%$ e $38,75\%$ de problemas, respectivamente.

A Figura 5 mostra os resultados para VFO. O algoritmo proposto registrou o menor valor em $14,72\%$ dos problemas, enquanto ML em $85,27\%$, CG em $57,36\%$ e MS em $50,38\%$ dos problemas resolvidos. Isso indica que os outros métodos, principalmente ML, tiveram desempenho melhor que GFP para VFO.

O fato de essas porcentagens não totalizarem 100% indica que os valores de MSE e VFO foram iguais para alguns métodos em alguns problemas, o que faz com que mais de um método tenha o menor MSE para um determinado problema. Isso reforça os resultados obtidos na Tabela 5, onde as diferenças relativas entre GFP e outros métodos foram pequenas para MSE e VFO. Com base nessas porcentagens, não foi possível destacar um método com melhor desempenho para MSE, e esse resultado está de acordo com o obtido para os problemas simulados conforme mostrado nas Tabelas 2, 3 e 4.

Quando compara-se os resultados obtidos para os problemas do conjunto 1 e do conjunto 2, as conclusões foram consistentes, fornecendo evidências de que o algoritmo GFP poderia ter um desempenho replicável para outros problemas não testados no presente trabalho, o que concluiria sua viabilidade.

4.6 EXPERIMENTOS E DISCUSSÕES PARA O LASSO

Nesta seção são apresentados os resultados dos experimentos numéricos realizados para a regressão com a regularização da norma ℓ_1 para os mesmos conjuntos de dados utilizados na seção anterior. Conforme discutido na Seção 3.3, quando o algoritmo do ADMM é utilizado para solucionar o problema de otimização do lasso, o subproblema de cada iteração assume a forma de um problema de regressão ridge. Portanto, o método proposto foi utilizado para resolver o subproblema do ADMM. Os demais métodos utilizados para a comparação também resolveram o mesmo subproblema do ADMM.

Conforme observado na seção anterior, os resultados numéricos do método proposto mostraram-se promissores quando $p > n$. Portanto, nos presentes experimentos numéricos, dá-se um enfoque maior para problemas com alta dimensão, tendo mais problemas com dimensões $p > n$ e proporções n/p diferentes. Os testes numéricos também indicaram que o número de condição de X não alterou os resultados obtidos, sendo assim, os resultados são novamente apresentados de forma agregada pela média por dimensão de X.

A Tabela 6 apresenta os resultados do ranking médio de GFP comparado aos demais métodos para as métricas de desempenho de tempo de execução, VFO e MSE. Como pode ser observado, GFP obteve ranking médio de 1.01 para o tempo de execução, obtendo o menor valor em quase todos os problemas resolvidos, com exceção de uma dimensão.

Para o VFO, GFP obteve ranking médio de 3.67. Nota-se pela Tabela 6, que o rank médio de GFP é menor conforme a proporção n/p diminui. Para o problema com dimensão 50×5000 obteve o a posição média de 2.13, e conforme aumenta n/p o *ranking* médio aumenta para o VFO. Enquanto que para a métrica de MSE, o método ML obteve o menor ranking médio de 1.24, com GFP ficando com a última posição na maior parte das dimensões testadas.

Na Tabela 7, tem-se a diferença relativa média de todas as métricas de desempenho em relação ao algoritmo GFP. Valores positivos da tabela indicam que os métodos ML, CG ou CW foram mais demorados em comparação com GFP para o tempo de execução. Valores positivos nas métricas MSE e VFO indicam que ML, CG ou CW apresentaram resultados mais elevados nessas métricas.

Para os problemas com proporção de $n/p = 0.01$ foram obtidos as maiores diferenças de tempo de execução, e essa diferença diminui conforme n/p aumenta. Para

TABELA 6 – Ranking médio de GFP e outros para cada dimensão $n \times p$ no lasso.

			Tempo				MSE				VFO			
n/p	n	p	A1	ML	CG	CW	A1	ML	CG	CW	A1	ML	CG	CW
0.01	25	2500	1.13	3.88	3.13	1.88	4	1.875	1.25	1.875	2.88	1.63	3.125	1.5
0.01	50	5000	1	4	3	2	4	1.5	2	1.50	2.13	2.25	2.375	2.25
0.01	75	7500	1	4	3	2	4	1.38	2.25	1.38	3.25	1.88	2	1.875
0.01	100	10000	1	4	3	2	4	1.50	2	1.50	2.5	2.25	2	2.25
0.1	250	2500	1	4	3	2	4	1.13	2.75	1.13	4	1.88	1.25	1.875
0.1	500	5000	1	4	3	2	4	1.29	2.43	1.29	4	1.43	2.14	1.43
0.1	750	7500	1	4	3	2	4	1	3	1	4	1.57	1.86	1.57
0.1	1000	10000	1	4	3	2	4	1	3	1	4	1.43	2.14	1.57
0.5	1250	2500	1	4	2.80	2.20	4	1.20	2.60	1.20	4	1.8	1.40	1.80
0.5	2500	5000	1	4	2	3	4	1.50	2	1.50	4	1.75	1.50	1.75
0.5	3750	7500	1	4	2	3	4	1	3	1.00	4	1.6	1.80	1.60
0.5	5000	10000	1	4	2	3	4	1.17	2.67	1.17	4	1.67	1.67	1.67
0.75	1875	2500	1	4	2.25	2.75	4	1.25	2.50	1.25	4	1.50	2.50	1.25
0.75	3750	5000	1	3	2	4	4	1	3	1	4	1.50	2	1.50
0.75	5625	7500	1	3	2	4	4	1	3	1	4	1.40	2.20	1.40
0.75	7500	10000	1	3	2	4	4	1	3	1	4	1.50	2.33	1.33
Média da Coluna			1.01	3.80	2.57	2.61	4.00	1.24	2.53	1.24	3.67	1.69	2.02	1.66

TABELA 7 – Média da diferença relativa de GFP para os outros para cada dimensão $n \times p$ no lasso em percentual

			Tempo			MSE			VFO		
n/p	n	p	ML	CG	CW	ML	CG	CW	ML	CG	CW
0.01	25	2500	362.63	177.12	44.64	-0.0118	-0.0118	-0.0118	317.79	317.79	317.79
0.01	50	5000	526.11	237.64	59.28	-0.0107	-0.0107	-0.0107	0.0015	0.0015	0.0015
0.01	75	7500	648.13	214.85	57.90	-0.0111	-0.0111	-0.0111	-0.0008	-0.0008	-0.0008
0.01	100	10000	827.31	235.25	68.89	-0.0110	-0.0110	-0.0110	0.0008	0.0008	0.0008
0.1	250	2500	26.80	14.62	4.52	-0.0100	-0.0100	-0.0100	-0.0062	-0.0062	-0.0062
0.1	500	5000	43.47	22.59	9.69	-0.0111	-0.0111	-0.0111	-0.0075	-0.0075	-0.0075
0.1	750	7500	55.33	22.69	11.55	-0.0103	-0.0103	-0.0103	-0.0066	-0.0066	-0.0066
0.1	1000	10000	68.49	25.26	14.27	-0.0104	-0.0104	-0.0104	-0.0067	-0.0067	-0.0067
0.5	1250	2500	4.81	3.47	2.95	-0.0098	-0.0098	-0.0098	-0.0082	-0.0082	-0.0082
0.5	2500	5000	8.21	5.10	5.68	-0.0089	-0.0089	-0.0089	-0.0068	-0.0068	-0.0068
0.5	3750	7500	12.24	7.13	8.77	-0.0095	-0.0095	-0.0095	-0.0075	-0.0075	-0.0075
0.5	5000	10000	15.61	9.01	11.55	-0.0099	-0.0099	-0.0099	-0.0081	-0.0081	-0.0081
0.75	1875	2500	27.86	14.79	4.68	-0.0925	-0.0925	-0.0925	-0.0854	-0.0854	-0.0854
0.75	3750	5000	6.21	4.26	6.84	-0.0096	-0.0096	-0.0096	-0.0078	-0.0078	-0.0078
0.75	5625	7500	9.66	6.29	10.33	-0.0098	-0.0098	-0.0098	-0.0082	-0.0082	-0.0082
0.75	7500	10000	12.65	8.38	14.04	-0.0100	-0.0100	-0.0100	-0.0084	-0.0084	-0.0084

as métricas de MSE e VFO, tem-se diferenças relativas negativas, o que indica que GFP obteve resultados maiores que os demais métodos. Este resultado concorda com a Tabela 6. Embora GFP tenha valores superiores para estas duas métricas, as diferenças relativas pequenas, o que indica resultados próximos entre GFP e demais métodos.

Em relação ao conjunto de dados mal condicionados, as Figuras 6, 7 e 8 mostram os resultados para as métricas de tempo de execução, MSE e VFO, respectivamente.

Observando a Figura 6, em relação ao tempo computacional, conclui-se que o algoritmo proposto solucionou mais rapidamente 93.35% dos 129 problemas do dataset. Os demais métodos tiveram valores de $\omega(\tau)$ pequenos.

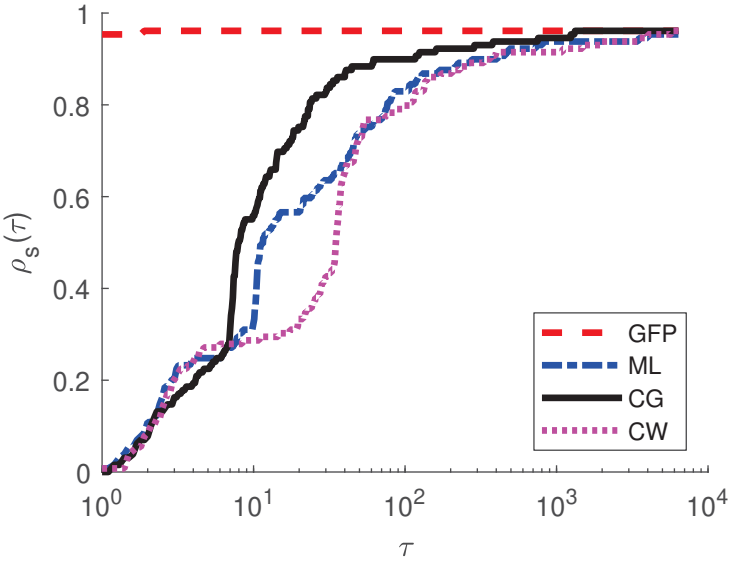


FIGURA 6 – Perfil de desempenho para tempo de execução de GFP, ML, CG e MS para o lasso

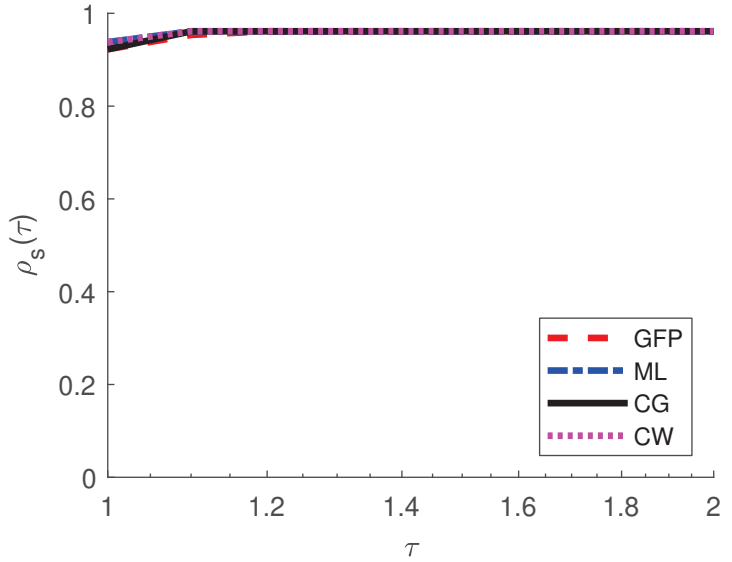


FIGURA 7 – Perfil de desempenho para MSE de GFP, ML, CG e MS para o lasso

Na Figura 7 os resultados para o MSE são expostos. Neste caso, observa-se que GFP solucionou 92.25% dos problemas com menor MSE Os demais métodos também tiveram percentuais similares como 0.9380, 0.9925, 0.9380 para ML, CG e CW respectivamente. O que indica que vários métodos tiveram o mesmo valor para o MSE para os problemas resolvidos. Todos os quatro métodos resolveram 96.2% dos 129 problemas do conjunto de dados.

Para a métrica de VFO, na Figura 8 tem-se que $\omega(1) = 0.9612$ o que significa que 96.12% dos 129 problemas foram solucionados com menor valor de VFO pelo algoritmo proposto neste trabalho. Os demais métodos obtiveram todos o valor de $\omega(1) = 0.9070$ o que significa que obtiveram o mesmo valor para VFO nos problemas solucionados.

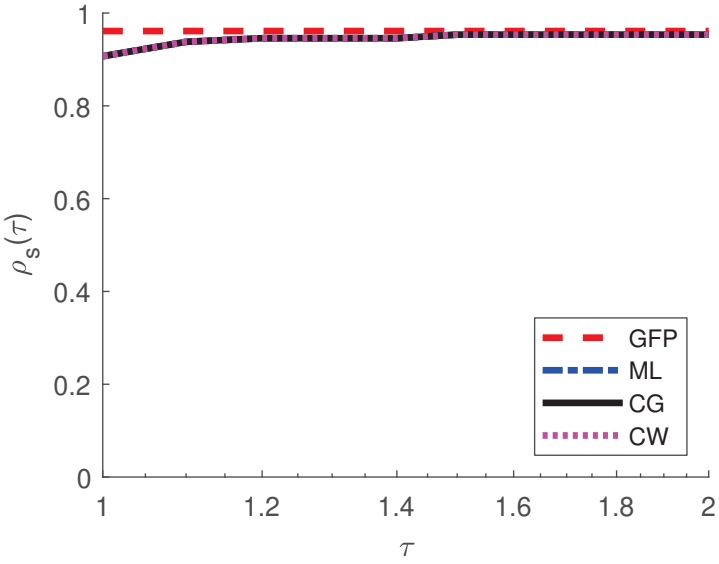


FIGURA 8 – Perfil de desempenho para VFO de GFP, ML, CG e MS para o lasso

4.7 EXPERIMENTOS E DISCUSSÕES PARA ELASTIC NET

Nesta seção, apresenta-se os resultados obtidos em experimentos numéricos ao minimizar a função objetivo do elastic net utilizando o algoritmo proposto e outros métodos.

A Tabela 8 apresenta os resultados comparativos dentre todos os métodos utilizados. Quando considera-se a métrica de tempo de execução, tem-se que GFP obteve ranking médio de 1.83, ficando com o menor tempo computacional em alguns problemas resolvidos, seguido pelo método CW que obteve ranking médio de 1.95. Na maior parte das dimensões, estes métodos ficaram em primeiro e segundo lugar no ranking, de forma alternada, enquanto que CG obteve ranking médio de 3.87 o que significa que foi o método com maior tempo de execução na maior parte dos problemas testados.

Sobre o VFO, conclui-se que CW obteve o menor ranking médio de 1.18, tendo o menor VFO em média, GFP obteve a 3ª posição em comparação aos demais métodos com um ranking médio de 3.26. Para a métrica MSE, o método CW obteve ranking médio de 1.78 sendo o método que obteve menor MSE, e GFP obteve ranking médio de 2.39, ficando na 3º posição do rank de MSE.

A Tabela 9 apresenta os resultados da diferença relativa média entre GFP e os demais métodos comparados para cada dimensão $n \times p$ para o elastic net. Pelos valores das diferenças relativas, entende-se que a diferença de tempo de execução para problemas com razão $n/p = 0.01$ é sempre maior do que 2 e obtendo os maiores valores de diferença relativa. Conforme a razão aumenta, as diferenças relativas vão diminuindo, o que mostra que todos os métodos tiveram um tempo de execução similar e uma maior diferença foi observada quando $n/p = 0.01$.

Para as métricas de VFO e MSE, os resultados foram similares e todas as diferenças

TABELA 8 – Ranking médio de GFP e outros para cada dimensão $n \times p$ no elastic net.

			Tempo				VFO				MSE			
n/p	n	p	GFP	ML	CG	CW	GFP	ML	CG	CW	GFP	ML	CG	CW
0.01	25	2500	1.13	3	4	1.88	3.63	3	1.25	1.25	2.13	3.38	1.88	1.88
0.01	50	5000	1	3	4	2	4	2.88	1	1.38	2.13	3.38	1.63	1.88
0.01	75	7500	1	3	4	2	3.88	3.13	1.25	1	2.13	3.63	1.63	1.75
0.01	100	10000	1	3	4	2	3.71	3.29	1.43	1.14	2.71	3.43	1.71	1.43
0.1	250	2500	2	3	4	1	3.38	3.13	1.5	1.38	2.13	3.38	2	1.75
0.1	500	5000	2	3	4	1	3.38	3.63	1.13	1.13	2.13	3.38	2	2
0.1	750	7500	1.75	3	4	1.25	3.25	3.75	1.25	1	2.5	3.5	1.75	1.63
0.1	1000	10000	1.13	3	4	1.88	3.25	3.75	1	1.13	2.88	3.38	1.63	1.38
0.5	1250	2500	3.5	2	3.5	1	2.75	4	1.25	1.13	2.88	3.13	1.63	2
0.5	2500	5000	2.63	2.25	4	1.13	3.13	3.88	1	1.38	2.13	3.38	2.13	2
0.5	3750	7500	1.63	1.88	4	2.5	3.13	3.88	1.5	1	1.75	3.5	2.13	2.38
0.5	5000	10000	1.25	2	4	2.75	3	3.75	1	1.25	2.5	3.25	1.71	1.88
0.75	1875	2500	4	1.25	3	1.75	2.75	4	1.25	1.5	2.5	3.25	2	1.63
0.75	3750	5000	2.57	1	4	2.43	3	4	1.14	1.14	2.29	3.57	2	1.86
0.75	5625	7500	1.71	1.29	3.86	3.14	3	4	1	1	2.71	3.43	1.71	1.57
0.75	7500	10000	1	2	3.57	3.43	3	4	1.14	1.14	2.71	3.43	2	1.57
Média da Coluna			1.83	2.35	3.87	1.95	3.26	3.63	1.19	1.18	2.39	3.4	1.84	1.78

relativas obtidas são pequenas em uma escala entre 10^{-7} a 10^{-14}. O que significa que apesar de GFP obter a 3ª posição no ranking de VFO e MSE, a diferença entre GFP e os demais métodos foi muito pequena.

TABELA 9 – Média da diferença relativa de GFP para os outros para cada dimensão $n \times p$ no elastic net em percentual

			Tempo			VFO			MSE		
n/p	n	p	ML	CG	CW	ML	CG	CW	ML	CG	CW
0.01	25	2500	63.47	100.59	4.9169	1E-07	1E-07	1E-07	6E-07	6E-07	6E-07
0.01	50	5000	29.242	71.231	2.4678	-3E-12	-3E-12	-3E-12	4E-07	4E-07	4E-07
0.01	75	7500	26.463	77.5	2.4024	-6E-13	-7E-13	-7E-13	2E-07	2E-07	2E-07
0.01	100	10000	26.087	95.76	2.9152	-2E-12	-2E-12	-2E-12	-5E-07	-5E-07	-5E-07
0.1	250	2500	2.0675	3.3871	-0.546	1E-11	1E-11	1E-11	2E-07	2E-07	2E-07
0.1	500	5000	2.0687	5.6778	-0.338	1E-12	-8E-13	-8E-13	-5E-09	-5E-09	-5E-09
0.1	750	7500	2.0044	6.4068	-0.067	2E-14	-1E-13	-1E-13	-9E-08	-9E-08	-9E-08
0.1	1000	10000	2.3773	9.1838	0.4012	5E-14	-9E-14	-9E-14	-1E-07	-1E-07	-1E-07
0.5	1250	2500	-0.258	0.0053	-0.556	3E-12	3E-12	3E-12	-5E-08	-5E-08	-5E-08
0.5	2500	5000	-0.058	0.611	-0.174	-2E-12	-2E-12	-2E-12	3E-08	3E-08	3E-08
0.5	3750	7500	0.0401	1.0113	-0.018	-7E-13	-1E-12	-1E-12	4E-08	4E-08	4E-08
0.5	5000	10000	0.4479	1.7902	0.6116	-2E-13	-2E-14	-5E-13	1E-08	1E-08	1E-08
0.75	1875	2500	-0.445	-0.274	-0.454	5E-12	4E-12	4E-12	3E-08	3E-08	3E-08
0.75	3750	5000	-0.285	0.1515	-0.017	1E-13	-3E-14	-3E-14	3E-08	3E-08	3E-08
0.75	5625	7500	-0.048	0.576	0.5008	2E-13	-2E-14	-2E-14	-7E-09	-7E-09	-7E-09
0.75	7500	10000	0.3597	1.2065	1.1901	2E-13	-1E-14	-1E-14	-3E-10	-3E-10	-3E-10

Os resultados indicam que o algoritmo proposto teve o menor ou segundo menor tempo de execução na maioria dos problemas. Embora tenha apresentado VFO e MSE ligeiramente maiores que os outros métodos, a diferença foi pequena, na ordem de 10^{-7}. Considerando que os valores de VFO e MSE são bem similares, o método proposto possui uma vantagem no tempo de execução dos problemas. Sendo uma metodologia a ser

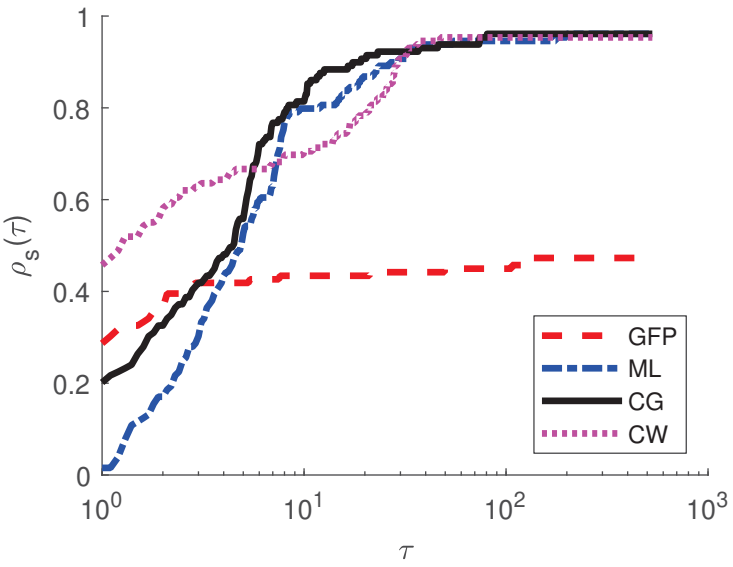


FIGURA 9 – Perfil de desempenho para tempo de execução de GFP, ML, CG e MS para o elastic net

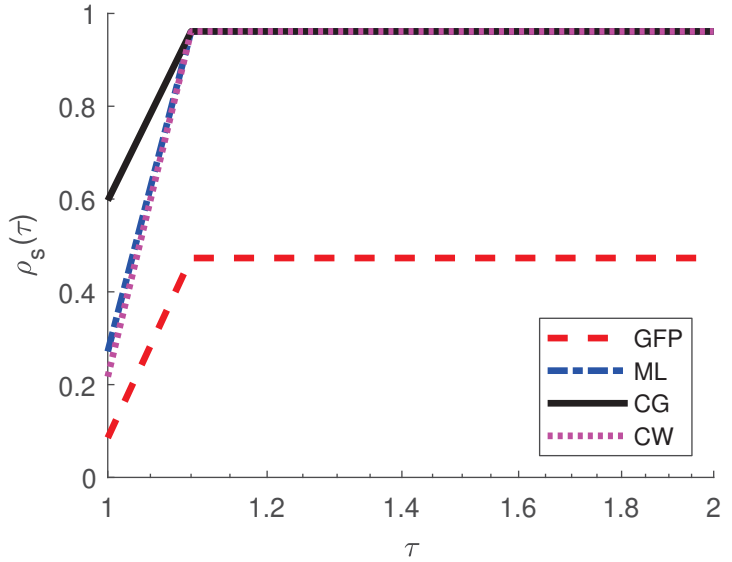


FIGURA 10 – Perfil de desempenho para MSE de GFP, ML, CG e MS para o elastic net

considerada para solução de problemas com estas características.

No que se refere ao segundo conjunto de dados utilizados nos experimentos numéricos, as figuras mostram o gráfico de perfil de desempenho dos 4 métodos comparados. Na Figura 9, tem-se os resultados para a métrica de tempo de execução. Os valores para $\omega(\tau)$ foram de 0.2868 para GFP, 0.0155 para ML, 0.2016 para CG e 0.4574 para CW. Percebe-se que a ponta da curva de GFP é mais baixa do que as dos demais métodos, o que indica que não foi possível encontrar a solução para uma parcela dos problemas.

Na figura 10 os resultados para o MSE são expostos. Neste caso, observa-se que GFP solucionou 8.35% dos problemas com menor MSE, e conseguiu resolver 47.29% dos 129 problemas. Para a métrica de VFO, os resultados na Figura 11 são semelhantes aos obtidos para o MSE. GFP solucionou 18.20% dos problemas com menor VFO, enquanto

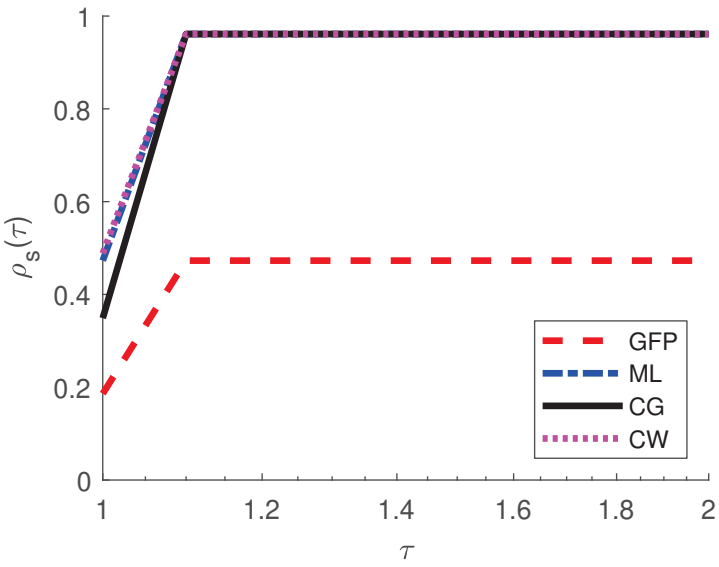


FIGURA 11 – Perfil de desempenho para VFO de GFP, ML, CG e MS para o elastic net

ML, CG e CW obtiveram valores como 47.29%, 34.88% e 48.84%, respectivamente.

Observa-se contrastes nos resultados dos experimentos ao lidar com matrizes mal condicionadas e quase singulares em comparação com os datasets de alta dimensionalidade mencionados anteriormente. Nos datasets gerados, o elastic net apresentou resultados promissores, contudo, para estes datasets com matrizes quase singulares, os três outros algoritmos exibiram um desempenho superior. Adicionalmente, na seção anterior dedicada ao lasso, destaca-se que o método GFP foi eficaz na resolução da maioria dos problemas identificados.

Assim, a eficácia do algoritmo proposto está intrinsecamente condicionada à natureza do problema a ser abordado. Tal observação não surpreende, dada a diversidade de domínios de conhecimento nos quais metodologias de regularização podem ser aplicadas.

4.8 APLICAÇÃO EM DADOS GENÔMICOS DE PRODUÇÃO DE RIBOFLAVINA

Para ilustrar os resultados, foi aplicado o algoritmo proposto no conjunto de dados de riboflavina de Bühlmann, Kalisch e Meier (2014). Este conjunto de dados considera a produção de riboflavina (vitamina B_2) com *Bacillus subtilis*. O objetivo de analisar conjuntos de dados genômicos como este é identificar genes que tenham uma relação positiva com a produção de vitaminas, informação que está presente na solução $\hat{\beta}$. Este conjunto de dados consiste em 71 amostras e 4088 variáveis de características. As variáveis explicativas são o logaritmo do nível de expressão de 4088 genes. A variável de resposta é o logaritmo da taxa de produção de riboflavina.

O mesmo procedimento aplicado aos experimentos na seção 4.5 foi empregado aqui. O λ foi definido como $\lambda = 2,5\|X\|_2^2$, enquanto o λ_{CV} foi encontrado por meio de uma pesquisa usando K-fold CV, com $K = 5$. Diferentes valores para o parâmetro são

candidatos e aquele com o menor MSE foi escolhido como λ_{CV}.

Para o ajuste dos modelos, os dados foram padronizados de forma que cada coluna de X tivesse média 0 e desvio padrão 1, e y estivesse centralizado. Por causa disso, não é necessário estimar um termo de interceptação.

O CV K-fold também foi utilizado para que haja uma divisão entre os conjuntos de treinamento e teste nos experimentos. Como $K = 5$ foi usado, cada um dos resultados apresentados na Tabela 10 são médias de 5 execuções. O tempo de execução foi medido durante a estimação do modelo (treinamento), o MSE foi medido na fase de teste e o VFO foi medido durante o treinamento.

TABELA 10 – Comparação de métodos no conjunto de dados de riboflavina, com as métricas tempo de execução da CPU, MSE e VFO

		Tempo (s)			
		GFP	ML	CG	MS
λ	50215	0.0193	1.7772	0.1452	0.1277
λ_{CV}	10	-	1.8495	0.4866	0.3277
		MSE			
		GFP	ML	CG	MS
λ	50215	0.8084	0.8084	0.8084	0.8084
λ_{CV}	10	-	0.2570	0.2570	0.2570
		VFO			
		GFP	ML	CG	MS
λ	50215	1478.34	1478.34	1478.34	1478.34
λ_{CV}	10	-	1456.18	1456.18	1456.18

Conforme observado na Tabela 10, como λ_{CV} é menor que $\|X\|_2^2$, GFP não convergiu para este parâmetro, e não registrou as métricas. Como esperado, considerando que λ_{CV} foi determinado para minimizar o MSE, o MSE foi menor para o modelo estimado com λ_{CV} do que para o modelo estimado com λ. Observe que os VFO para os modelos estimados por λ_{CV} foram ligeiramente inferiores aos estimados por λ.

Quando considera-se apenas os resultados do parâmetro λ e compara-se GFP com os outros métodos, o MSE e o VFO ficaram muito próximos. A diferença entre o MSE de GFP e os demais métodos ocorre na sexta casa decimal, para o VFO está na sétima casa decimal. Assim, GFP consegue obter resultados próximos aos demais métodos para MSE e VFO utilizando menor tempo computacional. GFP é aproximadamente 6 vezes mais rápido que o segundo método mais rápido (MS).

A razão n/p para este problema é 0.0173, que está próxima da razão 0.01 na Tabela 2. Nesse caso GFP é mais rápido que os outros métodos e o MSE e VFO não são

muito diferentes. Este teste com dados reais mostra que os resultados obtidos nos testes com dados simulados são consistentes, pois é possível chegar a conclusões semelhantes em casos de dados reais.

4.8.1 Lasso e Elastic Net

O algoritmo proposto no presente trabalho pode também ser aplicado para os métodos de regularização lasso e elastic net. Neste caso, os coeficientes dos modelos são encontrados por meio da solução dos problemas de otimização utilizando-se o ADMM. E então, o subproblema da variável primal do método é resolvida pelo algoritmo proposto.

A Tabela 11 expõe os resultados obtidos para a o problema com regularização ℓ_1. O problema lasso escalonado por c^2 como proposto no presente trabalho é apresentado nas linhas com o valor de λ, enquanto o problema original, que não foi multiplicado por c^2 aparece nas linhas da tabela em que tem-se λ_{orig}. Para λ_{orig} os resultados coincidiram com o esperado, em sua forma original o GFP não era aplicável ao problema. O mesmo ocorreu com CG que não encontrou uma solução para este problema.

Quando consideramos a solução encontrada para o λ estimado, todos os métodos encontraram uma solução. Em relação ao tempo de execução, observa-se que GFP foi o método com menor tempo de execução, seguido de MS, CG e ML. Quanto às métricas de MSE e VFO, tem-se que GFP obteve os menores valores, enquanto os demais métodos tiveram resultados similares.

TABELA 11 – Comparação de métodos no conjunto de dados de riboflavina, com as métricas tempo de execução da CPU, MSE e VFO para lasso

		Tempo (s)			
		GFP	ML	CG	MS
λ	1E-05	0.0016	0.5556	0.1579	0.0690
λ_{orig}	1	-	103.25	-	0.1146
		MSE			
		GFP	ML	CG	MS
λ	1E-05	1.35E-09	9.71E-09	9.71E-09	9.71E-09
λ_{orig}	1	-	51.2730	-	51.2729
		VFO			
		GFP	ML	CG	MS
λ	1E-05	2.28344E-06	2.96094E-06	2.96094E-06	2.96094E-06
λ_{orig}	1	-	1.4556	-	1.4556

A Tabela 12 apresenta os resultados dos experimentos para a técnica elastic net. De forma análoga aos experimentos anteriores, os parâmetros λ_1 e λ_2 foram estimados de

TABELA 12 – Comparação de métodos no conjunto de dados de riboflavina, com as métricas tempo de execução da CPU, MSE e VFO para elastic net

			Tempo (s)			
	λ_1	λ_2	GFP	ML	CG	CW
λ estimado	1E-3	3.49E+7	0.0302	0.8122	493.0218	0.0970
λ_{CV}	1E-5	1E+9	0.0060	0.6633	-	0.0782
			MSE			
	λ_1	λ_2	GFP	ML	CG	CW
λ estimado	1E-3	3.49E+7	6.5607	6.5593	51.2575	6.5593
λ_{CV}	1E-5	1E+9	0.8446	0.8446	-	0.8446
			VFO			
	λ_1	λ_2	GFP	ML	CG	CW
λ estimado	1E-3	3.49E+7	1242.8725	1242.8725	4.53E+6	1242.8725
λ_{CV}	1E-5	1E+9	1818.0483	1818.0483	-	1818.0483

duas formas diferentes. Primeiro utilizando a estimação sugerida pelo método proposto em que λ_2 é estimado como descrito na seção 4.4 e λ_1 é encontrado por CV, e que aparece na primeira linha de cada métrica. A segunda abordagem é a utilização de CV para escolher ambos parâmetros, em que seus resultados aparecem nas linhas de λ_{CV} na tabela. Cabe notar que ao escolher parâmetros de regularização diferentes, se resolve problemas diferentes, então estas duas abordagens para escolha dos parâmetros ocasionam a solução de dois problemas distintos.

Para esta aplicação, tem-se que o λ_2 estimado por CV é maior do que o λ_2 calculado, e o algoritmo proposto neste trabalho resolve os dois problemas. O método CG não encontrou solução para nenhum dos problemas. Como os λ_{CV} foram encontrados por CV ao minimizar o MSE, espera-se que as métricas de MSE e VFO sejam inferiores para este problema, o que é confirmado pela Tabela 12.

Considerando o problema com λ_{CV}, os resultados na Tabela 12 mostram que o tempo de execução foi o menor para GFP, seguido de CW e ML. Os valores de MSE e VFO são similares, com diferenças que ocorrem na oitava casa decimal, para todos os três métodos de solução.

Considerando λ_2 estimado e λ_1 encontrado por CV. Os resultados encontrados corroboram com as conclusões anteriores. O tempo de execução do método proposto é menor que os demais e para as métricas de MSE e VFO os valores obtidos são similares.

Dessa forma, os resultados obtidos indicam que o método proposto consegue solucionar os problemas mais rapidamente que os demais métodos, porém mantendo o mesmo valor para as métricas de MSE e VFO. Esta aplicação indica também que existem

problemas em que as condições para a aplicação do algoritmo proposto são cumpridas e este pode ser empregado na solução desses problemas.

5 CONSIDERAÇÕES FINAIS

No presente trabalho, foi proposto um novo algoritmo para resolver problemas de regressão ridge. Este algoritmo é baseado em manipulações algébricas da condição de otimalidade necessária de primeira ordem, com o intuito de encontrar uma solução fechada para o problema. Esta recai em uma expressão que é tratada como um problema de ponto fixo. O processo iterativo é então conduzido por operações básicas entre matrizes e vetores. No capítulo 3 apresentou-se o algoritmo proposto, além da análise de convergência. Em seguida, o algoritmo foi avaliado por meio de experimentos numéricos comparativos usando tempo de execução, VFO e MSE como métricas de desempenho.

O algoritmo foi implementado em MATLAB ® e aplicado à problemas gerados aleatoriamente, de forma que possuíssem alta dimensão e fossem mal condicionados, a fim de avaliar a viabilidade da proposta como um método de otimização. Adicionalmente, foram considerados para os experimentos, datasets de problemas singulares o que adicionaram uma variedade maior na natureza de problemas testados. Os resultados dos experimentos de comparação evidenciam o potencial do algoritmo na resolução dos problemas testados. Outro aspecto a ser notado é a facilidade de implementação e entendimento do algoritmo.

O λ foi calculado e escolhido pelo algoritmo proposto neste trabalho, dispensando assim uma etapa adicional para computar um parâmetro de regularização e evitando a busca pelo parâmetro.

Para dados gerados, os resultados indicam que o algoritmo proposto forneceu resultados muito semelhantes a outros métodos bem estabelecidos, como CG, MS e ML, além de ser de fácil entendimento e fácil implementação.

Em relação ao tempo de execução, GFP se destaca entre os métodos concorrentes quando $p \gg n$. Com relação ao valor de MSE e OFV, todos os métodos comparados apresentaram valores diferentes, não havendo um único método que prevalecesse em todos os problemas. O algoritmo proposto produziu valores semelhantes a outros métodos. Este fato indica que a solução β foi muito próxima da solução dos métodos estabelecidos.

Para as matrizes obtidas em Foster (2021), tem-se que os resultados do problema de regressão ridge e lasso concordam com as conclusões obtidas para as matrizes geradas. Ou seja, possui menor tempo computacional que os demais e as métricas de VFO e MSE são semelhantes ou melhores. Ao mesmo tempo, para o problemas solucionados para a técnica elastic net, os demais métodos comparados obtiveram métricas com valores inferiores.

Dessa forma, o presente trabalho se concentrou nas técnicas de regularização de regressão ridge, lasso e elastic net. Um novo algoritmo para a resolução do problema

de otimização, específico da regressão ridge, foi proposto. Baseado nos resultados dos experimentos numéricos, tem-se uma abordagem promissora para aprimorar a eficiência no tempo de execução de problemas com matrizes com alta dimensionalidade.

SEQUÊNCIA DA PESQUISA

Seguem passos a serem tomados para prosseguimento da pesquisa.

- Os resultados obtidos na aplicação do algoritmo proposto para solução do problema de otimização do lasso para os um dos dois conjuntos de dados adotados não atenderam as expectativas da pesquisa. A sequência da pesquisa visa empreender análises mais aprofundadas com o intuito de aprimorar a eficácia da abordagem proposta para estes tipos de problemas.
- Implementação da geração de datasets para experimentos numéricos a fim de abranger um maior número de parâmetros. Dessa forma, é possível avaliar de forma mais aprofundada quais condições de problema o algoritmo proposto obtém resultados superiores. Os dados gerados neste trabalho possuem a variação e controle apenas do número de condição da matriz X. Porém existem outros parâmetros que poderiam ser empregados de forma a expandir a natureza dos datasets. Alguns exemplos de parâmetros incluem: correlação da matriz X, nível de ruído nos dados, diferentes graus de esparsidade do vetor β.

REFERÊNCIAS

BIERLAIRE, M.; TOINT, Ph L.; TUYTTENS, D. On iterative algorithms for linear least squares problems with bound constraints. **Linear Algebra its Appl.**, North-Holland, v. 143, p. 111–143, C jan. 1991. ISSN 0024-3795. DOI: 10.1016/0024-3795(91)90009-L.

BOYD, Stephen P. et al. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. **Foundations and Trends in Machine Learning**, v. 3, n. 1, p. 1–122, 2011. DOI: 10.1561/2200000016.

BÜHLMANN, Peter; KALISCH, Markus; MEIER, Lukas. High-Dimensional Statistics with a View Toward Applications in Biology. **Annual Review of Statistics and Its Application**, v. 1, n. 1, p. 255–278, 2014. DOI: 10.1146/annurev-statistics-022513-115545.

CASAGRANDE, M. H. **Comparação de métodos de estimação para problemas com colinearidade e/ou alta dimensionalidade (p>n)**. 2016. Dissertação (Mestre em Estatística) – Instituto de Ciências Matemáticas e de Computação, USP, São Carlos.

DAVIS, Timothy A.; HU, Yifan. The University of Florida Sparse Matrix Collection. **ACM Trans. Math. Softw.**, Association for Computing Machinery, New York, NY, USA, v. 38, n. 1, dez. 2011. ISSN 0098-3500. DOI: 10.1145/2049662.2049663.

DOLAN, E. D.; MORÉ, J. J. Benchmarking optimization software with performance profiles. **Mathematical Programming**, v. 91, n. 2, p. 201–213, 2002.

FAN, Jianqing; LV, Jinchi. Sure Independence Screening for Ultrahigh Dimensional Feature Space. **Journal of the Royal Statistical Society Series B: Statistical Methodology**, v. 70, n. 5, p. 849–911, out. 2008. ISSN 1369-7412. DOI: 10.1111/j.1467-9868.2008.00674.x.

FOSTER, L. **San Jose State University Singular Matrix Database**. 2021. Disponível em: <http://www.math.sjsu.edu/singular/matrices/index.html>. Acesso em: 1 out. 2021.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. et al. **The elements of statistical learning**. [S.l.]: Springer series in statistics New York, 2001. v. 1.

FRIEDMAN, Jerome; HASTIE, Trevor; HÖFLING, Holger et al. Pathwise coordinate optimization. **The Annals of Applied Statistics**, v. 1, n. 2, p. 302–332, dez. 2007. ISSN 1932-6157. DOI: 10.1214/07-AOAS131.

GOLUB, G. H.; VAN LOAN, C. F. **Matrix Computations**. Third. [S.l.]: The Johns Hopkins University Press, 1996.

HANSEN, Per Christian. Regularization tools – a matlab package for analysis and solution of discrete ill-posed problems. **Numer. Algorithms**, 1994. DOI: 10.1007/BF02149761.

HARVILLE, D.A. **Matrix Algebra From a Statistician's Perspective**. [S.l.]: Springer New York, 2008.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition**. [S.l.]: Springer New York, 2009. (Springer Series in Statistics). ISBN 9780387848587.

HASTIE, T.; TIBSHIRANI, R.; WAINWRIGHT, M. **Statistical Learning with Sparsity: The Lasso and Generalizations**. [S.l.]: CRC Press, 2015. (Chapman & Hall/CRC Monographs on Statistics & Applied Probability). ISBN 9781498712170.

HOERL, A. E.; KENNARD, R. W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. **Technometrics**, v. 12, p. 55–67, 1970.

JAMES, G. et al. **An Introduction to Statistical Learning: with Applications in R**. [S.l.]: Springer New York, 2014. (Springer Texts in Statistics). ISBN 9781461471370.

JOLLIFFE, Ian. **Principal component analysis**. New York: Springer Verlag, 2002. DOI: https://doi.org/10.1007/b98835.

KUMAR, Shailesh. **Soft Thresholding**. 2015. Disponível em: <https://sparse-plex.readthedocs.io/en/latest/book/opt/soft_thresholding.html>. Acesso em: 28 jul. 2022.

LIU, Guohui; XU, Hong-kun. PRIMAL-DUAL FIXED POINT METHODS FOR REGULARIZED LEAST-SQUARES PROBLEMS. **Fixed Point Theory**, v. 24, p. 661–674, 2023. ISSN 1583-5022. DOI: https://doi.org/10.24193/fpt-ro.2023.2.13.

PAIGE, C. C.; SAUNDERS, M. A. Solution of Sparse Indefinite Systems of Linear Equations. **SIAM J. Numer. Anal.**, v. 12, n. 4, p. 617–629, 1975. DOI: 10.1137/0712047.

PARIKH, N.; BOYD, S. Proximal algorithms. **Foundations and Trends in optimization**, Now Publishers Inc. Hanover, MA, USA, v. 1, n. 3, p. 127–239, 2014.

RIBEIRO, A. A.; KARAS, E. W. **Otimização Contínua: Aspectos teóricos e computacionais**. [S.l.]: Cengage Learning, 2013.

RIBEIRO, Ademir Alves; RICHTÁRIK, Peter. The complexity of primal-dual fixed point methods for ridge regression. **Linear Algebra and its Applications**, v. 556, p. 342–372, 2018. ISSN 0024-3795. DOI: https://doi.org/10.1016/j.laa.2018.07.017.

SALEH, A.K.M.E.; ARASHI, M.; KIBRIA, B.M.G. **Theory of Ridge Regression Estimation with Applications**. [S.l.]: Wiley, 2019. (Wiley Series in Probability and Statistics).

SILVA, T. C. **Algoritmos primais-duais de ponto fixo aplicados ao problema ridge regression**. 2016. Tese (Doutorado em Métodos Numéricos em Engenharia) – Setores de Tecnologia e de Ciências Exatas, Universidade Federal do Paraná, UFPR, Curitiba.

SILVA, Tatiane C.; RIBEIRO, Ademir A.; PERIÇARO, Gislaine A. A new accelerated algorithm for ill-conditioned ridge regression problems. **Computational and Applied Mathematics**, Wiley Online Library, v. 37, 2018. DOI: 10.1007/s40314-017-0430-4.

TAO, Shaozhe; BOLEY, Daniel; ZHANG, Shuzhong. Local Linear Convergence of ISTA and FISTA on the LASSO Problem. **SIAM Journal on Optimization**, v. 26, n. 1, p. 313–336, 2016. DOI: 10.1137/151004549.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 58, n. 1, p. 267–288, 1996.

TIBSHIRANI, Robert. Regression shrinkage and selection via the lasso. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 58, n. 1, p. 267–288, 1996.

ZOU, H.; HASTIE, T. Regularization and Variable Selection via the Elastic Net. **Journal of the Royal Statistical Society: Series B (Statistical Methodology)**, n. 2, p. 301–320, 2005.

APÊNDICES

APÊNDICE A – CONCEITOS FUNDAMENTAIS

Na presente seção, são apresentados conceitos e resultados fundamentais que desempenham o papel de fundamentação teórica da presente tese, mas que não puderam ser incorporados ao texto principal por conta da organização da tese. Este capítulo é opcional caso o leitor já esteja familiarizado com conceitos apresentados, motivo de estarem em apêndice.

As referências para os conceitos apresentados podem ser consultados com maior detalhe em Golub e Van Loan (1996) e Harville (2008).

A.1 CONCEITOS DE ÁLGEBRA

Teorema A.1 (*Decomposição em Valores Singulares* (SVD))**.** *Seja $A \in \mathbb{R}^{n\times p}$ uma matriz real, então existem matrizes ortogonais*

$$U = [u_1, \ldots, u_n] \in \mathbb{R}^{n\times n} \quad e \quad V = [v_1, \ldots, v_p] \in \mathbb{R}^{p\times p}$$

tais que

$$U^T AV = diag(d_1, \ldots, d_m) \in \mathbb{R}^{n\times p}, \quad m = min(n, p)$$

em que $d_1 \geq d_2 \geq \ldots \geq d_m \geq 0$.

Os elementos não nulos d_i da matriz diagonal são as raízes quadradas dos autovalores não nulos, não necessariamente distintos, de A^TA ou AA^T. Como o título do teorema sugere, estes escalares d_i são chamados de valores singulares de A. Os vetores u_i e v_i, que denotam as colunas de U e V, são o i-ésimo vetor singular à esquerda e o i-ésimo vetor singular à direita de A, respectivamente. Denotando $D = diag(d_1, \ldots, d_m)$, pode-se descrever a decomposição como:

$$A = UDV^T.$$

A SVD de uma matriz fornece informações sobre a estrutura da mesma e portanto, possui diferentes aplicações. Seja a SVD de A dada de acordo com o Teorema A.1, e define-se r de forma que

$$d_1 \geq \ldots \geq d_r > d_{r+1} = \ldots = d_m = 0,$$

então o $\text{rank}(A) = r$ e pode-se escrever a expansão da SVD

$$a_{ij} = \sum_{k=1}^{r} d_k u_{ki} v_{kj}.$$

Corolário A.2. O número de valores singulares não nulos de uma matriz A é igual ao posto de A.

Dessa forma, pode-se utilizar a SVD para determinar o posto de uma dada matriz. Dependendo da dimensão de A, pode-se obter uma matriz diagonal D com diversas linhas ou colunas nulas,

- se $n = p$,
$$\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_m \end{pmatrix}$$
- se $n < p$
$$\begin{pmatrix} d_1 & & & \dots \\ & \ddots & & \dots \\ & & d_m & \dots \end{pmatrix}$$
- se $n > p$
$$\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_m \\ \vdots & \vdots & \vdots \end{pmatrix}.$$

Em virtude disso, é usual escrever uma versão reduzida da SVD, que tem o intuito de obter informações baseadas apenas nos vetores definidos pelo posto da matriz A.

Se $A = UDV^T \in \mathbb{R}^{n\times p}$ é a SVD de A e $n \geq p$, então

$$A = U_l D_l V^T, \tag{A.1}$$

em que

$$U_l = [u_i, \dots, u_p] \in \mathbb{R}^{n\times p} \quad \text{e} \quad D_l = diag(d_1, \dots, d_p) \in \mathbb{R}^{p\times p}.$$

Dessa forma, em sua versão reduzida, não é necessário realizar a computação de $(n - p)$ colunas para obter U_l, em comparação com U, o que torna a SVD reduzida mais eficiente que a SVD do Teorema A.1.

É igualmente possível escrever uma decomposição reduzida para o caso em que $n \leq p$, ao alterar a dimensão de outra matriz da decomposição. Neste caso, $A = UD_kV_k^T$, em que $U \in \mathbb{R}^{n\times n}$, $D_k \in \mathbb{R}^{n\times n}$ e $V_k \in \mathbb{R}^{p\times n}$.

ANEXOS

ANEXO A – DATASETS

A Tabela 13 apresenta algumas características das matrizes utilizadas nos testes, retiradas de Foster (2021). O cabeçalho desta tabela pode ser compreendido por:

n : observações, número de linhas da matriz X,

p : atributos, número de colunas da matriz X,

$\kappa(X)$: número de condição de X,

ω : número de elementos não nulo em X,

ω_{row} : número de elementos não nulo em X por linha,

η : razão $100\frac{\omega}{np}$, que representa a densidade da matriz e o percentual de elementos não nulo.

TABELA 13 – Características dos problemas.

Problema	**n**	**p**	$\kappa(X)$	ω	ω_{row}	η
Muite/Chebyshev2	2053	2053	6,71E+16	18447	8,99	0,44
NYPA/Maragal_1	32	14		234	7,31	52,23
NYPA/Maragal_2	555	350		4357	7,85	2,24
NYPA/Maragal_3	1690	860		18391	10,88	1,27
NYPA/Maragal_4	1964	1034		26719	13,60	1,32
NYPA/Maragal_5	4654	3320		93091	20,00	0,60
Grund/b_dyn	1089	1089	6,05E+14	4144	3,81	0,35
Regtools/baart_100	100	100	6,86E+17	10000	100	100
Regtools/baart_200	200	200	9,48E+18	40000	200	100
Regtools/baart_500	500	500	6,78E+19	250000	500	100
Schenk_IBMNA/c-30	5321	5321	2,10E+13	65693	12,35	0,23
Meszaros/de063155	852	1848		4913	5,77	0,31
Meszaros/de063157	936	1908		5119	5,47	0,29
Marini/eurqsa	7245	7245	6,55E+04	46142	6,37	0,09
Regtools/foxgood_100	100	100	5,98E+18	10000	100	100
Regtools/foxgood_1000	1000	1000	2,78E+21	1000000	1000	100
Regtools/foxgood_200	200	200	6,90E+18	40000	200	100
Regtools/foxgood_500	500	500	1,87E+20	250000	500	100
Sandia/fpga_dcop_01	1220	1220	2,24E+34	5892	4,83	0,40
Sandia/fpga_dcop_02	1220	1220	1,58E+17	5892	4,83	0,40

Continuação na próxima página

TABELA 13 – Continuação da página anterior

Problema	**n**	**p**	$\kappa(X)$	τ	τ_{row}	ρ
Sandia/fpga_dcop_04	1220	1220	1,06E+14	5884	4,82	0,40
Sandia/fpga_dcop_05	1220	1220	2,74E+13	5852	4,80	0,39
Sandia/fpga_dcop_06	1220	1220	1,80E+13	5860	4,80	0,39
Sandia/fpga_dcop_07	1220	1220	6,25E+14	5855	4,80	0,39
Sandia/fpga_dcop_08	1220	1220	4,14E+13	5888	4,83	0,40
Sandia/fpga_dcop_10	1220	1220	1,42E+13	5884	4,82	0,40
Hollinger/g7jac010	2880	2880	5,98E+18	18229	6,33	0,22
Hollinger/g7jac010sc	2880	2880	9,74E+13	18229	6,33	0,22
Regtools/gravity_100	100	100	3,92E+18	10000	100	100
Regtools/gravity_1000	1000	1000	4,68E+20	1000000	1000	100
Regtools/gravity_200	200	200	1,13E+19	40000	200	100
Regtools/gravity_500	500	500	8,38E+19	250000	500	100
Regtools/heat_100	100	100	4,26E+36	5050	50,50	50,50
Regtools/heat_1000	1000	1000	1,32E+232	500500	500,50	50,05
Regtools/heat_200	200	200	4,08E+58	20100	100,50	50,25
Regtools/heat_500	500	500	5,39E+124	125250	250,50	50,10
Regtools/i_laplace_100	100	100	1,32E+32	7590	75,90	75,90
Regtools/i_laplace_1000	1000	1000	6,55E+04	301876	301,88	30,19
Regtools/i_laplace_200	200	200	1,19E+33	24055	120,28	60,14
Regtools/i_laplace_500	500	500	6,55E+04	104149	208,30	41,66
Meszaros/iprob	3001	3001	1,20E+23	9000	3,00	0,10
Mallya/lhr04	4101	4101	1,80E+24	81057	19,77	0,48
LPnetlib/lp_25fv47	821	1876		10705	13,04	0,70
LPnetlib/lp_bnl1	643	1586		5532	8,60	0,54
LPnetlib/lp_bore3d	233	334		1448	6,21	1,86
LPnetlib/lp_brandy	220	303		2202	10,01	3,30
LPnetlib/lp_cre_a	3516	7248		18168	5,17	0,07
LPnetlib/lp_cre_c	3068	6411		15977	5,21	0,08
LPnetlib/lp_cycle	1903	3371		21234	11,16	0,33
LPnetlib/lp_degen2	444	757		4201	9,46	1,25
LPnetlib/lp_dfl001	6071	12230		35632	5,87	0,05
LPnetlib/lp_greenbea	2392	5598		31070	12,99	0,23
LPnetlib/lp_greenbeb	2392	5598		31070	12,99	0,23
LPnetlib/lp_modszk1	687	1620		3168	4,61	0,28
LPnetlib/lp_pds_02	2953	7716		16571	5,61	0,07
LPnetlib/lp_scorpion	388	466		1534	3,95	0,85

Continuação na próxima página

TABELA 13 – Continuação da página anterior

Problema	**n**	**p**	$\kappa(X)$	τ	τ_{row}	ρ
LPnetlib/lp_shell	536	1777		3558	6,64	0,37
LPnetlib/lp_ship04l	402	2166		6380	15,87	0,73
LPnetlib/lp_ship04s	402	1506		4400	10,95	0,73
LPnetlib/lp_ship08l	778	4363		12882	16,56	0,38
LPnetlib/lp_ship08s	778	2467		7194	9,25	0,37
LPnetlib/lp_ship12l	1151	5533		16276	14,14	0,26
LPnetlib/lp_ship12s	1151	2869		8284	7,20	0,25
LPnetlib/lp_standgub	361	1383		3338	9,25	0,67
LPnetlib/lp_tuff	333	628		4561	13,70	2,18
LPnetlib/lp_wood1p	244	2595		70216	287,77	11,09
LPnetlib/lpi_cplex2	224	378		1215	5,42	1,43
LPnetlib/lpi_gosh	3792	13455		99953	26,36	0,20
LPnetlib/lpi_gran	2658	2525		20111	7,57	0,30
LPnetlib/lpi_greenbea	2393	5596		31074	12,99	0,23
LPnetlib/lpi_mondou2	312	604		1208	3,87	0,64
Meszaros/model2	379	1321		7607	20,07	1,52
Meszaros/model5	1888	11802		89925	47,63	0,40
Meszaros/model6	2096	5289		27628	13,18	0,25
Meszaros/model9	2879	10939		55956	19,44	0,18
Meszaros/nemspmm1	2372	8903		55867	23,55	0,26
Sandia/oscil_dcop_06	430	430	1,90E+20	1544	3,59	0,84
Sandia/oscil_dcop_07	430	430	3,56E+16	1544	3,59	0,84
Sandia/oscil_dcop_08	430	430	2,49E+14	1544	3,59	0,84
Sandia/oscil_dcop_17	430	430	2,22E+14	1544	3,59	0,84
Sandia/oscil_dcop_18	430	430	1,91E+16	1544	3,59	0,84
Sandia/oscil_dcop_19	430	430	3,20E+15	1544	3,59	0,84
Sandia/oscil_dcop_20	430	430	6,57E+15	1544	3,59	0,84
Sandia/oscil_dcop_21	430	430	2,93E+14	1544	3,59	0,84
Sandia/oscil_dcop_22	430	430	6,24E+15	1544	3,59	0,84
Sandia/oscil_dcop_23	430	430	2,28E+19	1544	3,59	0,84
Sandia/oscil_dcop_24	430	430	7,05E+19	1544	3,59	0,84
Sandia/oscil_dcop_25	430	430	2,33E+14	1544	3,59	0,84
Sandia/oscil_dcop_26	430	430	4,81E+14	1544	3,59	0,84
Sandia/oscil_dcop_27	430	430	1,00E+15	1544	3,59	0,84
Sandia/oscil_dcop_28	430	430	3,78E+26	1544	3,59	0,84
Sandia/oscil_dcop_29	430	430	2,17E+26	1544	3,59	0,84

Continuação na próxima página

TABELA 13 – Continuação da página anterior

Problema	**n**	**p**	$\kappa(X)$	τ	τ_{row}	ρ
Sandia/oscil_dcop_30	430	430	7,36E+26	1544	3,59	0,84
Sandia/oscil_dcop_31	430	430	9,05E+19	1544	3,59	0,84
Sandia/oscil_dcop_32	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_33	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_34	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_35	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_36	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_37	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_38	430	430	2,88E+17	1544	3,59	0,84
Sandia/oscil_dcop_39	430	430	5,30E+18	1544	3,59	0,84
Sandia/oscil_dcop_40	430	430	2,34E+20	1544	3,59	0,84
Sandia/oscil_dcop_41	430	430	2,31E+20	1544	3,59	0,84
Sandia/oscil_dcop_42	430	430	4,43E+20	1544	3,59	0,84
Sandia/oscil_dcop_43	430	430	4,12E+20	1544	3,59	0,84
Sandia/oscil_dcop_44	430	430	2,35E+20	1544	3,59	0,84
Sandia/oscil_dcop_45	430	430	3,20E+19	1544	3,59	0,84
Regtools/parallax_100	26	100	5,60E+14	2600	100	100
Regtools/parallax_1000	26	1000	4,59E+14	26000	1000	100
Regtools/parallax_200	26	200	4,89E+14	5200	200	100
Regtools/parallax_500	26	500	4,65E+14	13000	500	100
Simon/raefsky6	3402	3402	2,67E+16	130371	38,32	1,13
Regtools/shaw_100	100	100	2,90E+19	10000	100	100
Regtools/shaw_1000	1000	1000	1,01E+21	1000000	1000	100
Regtools/shaw_200	200	200	2,11E+19	40000	200	100
Regtools/shaw_500	500	500	9,28E+19	250000	500	100
HB/sherman3	5005	5005	6,90E+16	20033	4	0,08
Regtools/tomo_100	100	100	9,30E+18	1342	13,42	13,42
Regtools/tomo_2500	2500	2500	6,55E+04	166782	66,71	2,67
Regtools/tomo_900	900	900	6,55E+04	35598	39,55	4,39
Regtools/ursell_100	100	100	9,36E+13	10000	100	100
Regtools/ursell_1000	1000	1000	1,82E+13	1000000	1000	100
Regtools/ursell_200	200	200	3,34E+13	40000	200	100
Regtools/ursell_500	500	500	3,64E+13	250000	500	100
Regtools/wing_100	100	100	1,50E+20	10000	100	100
Regtools/wing_1000	1000	1000	8,68E+22	1000000	1000	100
Regtools/wing_200	200	200	7,06E+19	40000	200	100

Continuação na próxima página

TABELA 13 – Continuação da página anterior

Problema	**n**	**p**	$\kappa(X)$	τ	τ_{row}	ρ
Regtools/wing_500	500	500	2,43E+22	250000	500	100