PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Escuela de Posgrado

Enhancing safe screening rules using adaptive thresholding for regularized optimization problems

Tesis para obtener el grado académico de Maestro en Procesamiento de Señales e Imágenes Digitales que presenta:

Hector Francisco Chahuara Silva

Asesor:

Paul Antonio Rodríguez Valderrama

Lima, 2025

Informe de Similitud

Yo, Paul Antonio Rodríguez Valderrama, docente de la Escuela de Posgrado de la Pontificia Universidad Católica del Perú, asesor de la tesis titulada “Enhancing safe screening rules using adaptive thresholding for regularized optimization problems” del autor Hector Francisco Chahuara Silva, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud del 21 %. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 12/01/2026. No obstante, las referencias 1 y 7 del reporte corresponden a un mismo trabajo previamente publicado por el propio autor, el cual forma parte del desarrollo de la presente tesis. Las coincidencias del documento con dicho trabajo están asociadas a la definición formal y a la formulación matemática del método desarrollado en la tesis. El resto de las coincidencias señaladas en el reporte corresponden a la exposición de definiciones y expresiones matemáticas relacionadas con el tema de estudio. El índice de similitud, omitiendo las referencias previamente señaladas, es del 17 %, lo cual se encuentra dentro del límite establecido.
- He revisado con detalle dicho reporte y la tesis, y no se advierten indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha:

Lima, 12 de Enero de 2026.

<table>
<tr><td colspan="2">Apellidos y nombres del asesor:
Rodríguez Valderrama, Paul Antonio</td></tr>
<tr><td>DNI:
07754238</td><td rowspan="2">Firma</td></tr>
<tr><td>ORCID:
0000-0002-8501-0907</td></tr>
</table>

Resumen

La esparsidad es una propiedad ampliamente valorada en diversas aplicaciones de aprendizaje automático y procesamiento de señales pues permite una representación eficiente de datos y previene el sobreajuste en modelos de aprendizaje. A pesar de su amplia utilidad, el uso de la esparsidad para reducir las demandas computacionales sigue siendo un área activa de investigación. En optimización matemática, la esparsidad puede promoverse en las soluciones mediante técnicas de regularización o imponiendo restricciones explícitas. Recientemente se han propuesto técnicas como las reglas de cribado para explotar la esparsidad y disminuir los requerimientos computacionales de problemas de optimización a gran escala. Sin embargo, los métodos de cribado más avanzados actualmente disponibles proporcionan solo aproximaciones inexactas del soporte de la solución, descartando pocos elementos y, por lo tanto, ofreciendo ahorros computacionales limitados.

Esta tesis contribuye con una extensión de las reglas de cribado seguro basada en umbralización adaptativa orientada a problemas de optimización regularizada. La regla de umbralización propuesta se fundamenta en la observación de que la métrica utilizada para identificar características no contribuyentes puede considerarse que presenta una distribución aproximadamente unimodal para fines prácticos. El enfoque propuesto fue incorporado en el algoritmo gradiente proximal acelerado (APG) / algoritmo de contracción iterativa rápida (FISTA), un optimizador que recientemente ha atraído atención debido a su tasa de convergencia teórica $\mathcal{O}\left(k^{-2}\right)$. Para validar la técnica propuesta, se llevaron a cabo experimentos computacionales en múltiples contextos. Los resultados experimentales indican que el método propuesto proporciona mejoras de velocidad superiores tanto a las reglas de cribado seguro como a las de cribado fuerte en sus versiones estáticas (como etapa de preprocesamiento), tanto en conjuntos de datos sintéticos como reales, con una aceleración de hasta aproximadamente 33.9 veces. El método desarrollado en esta tesis fue presentado en un artículo de conferencia en la *2023 24th International Conference on Digital Signal Processing (DSP)*.

Palabras clave

Reglas de cribado, esparsidad, umbralización adaptativa

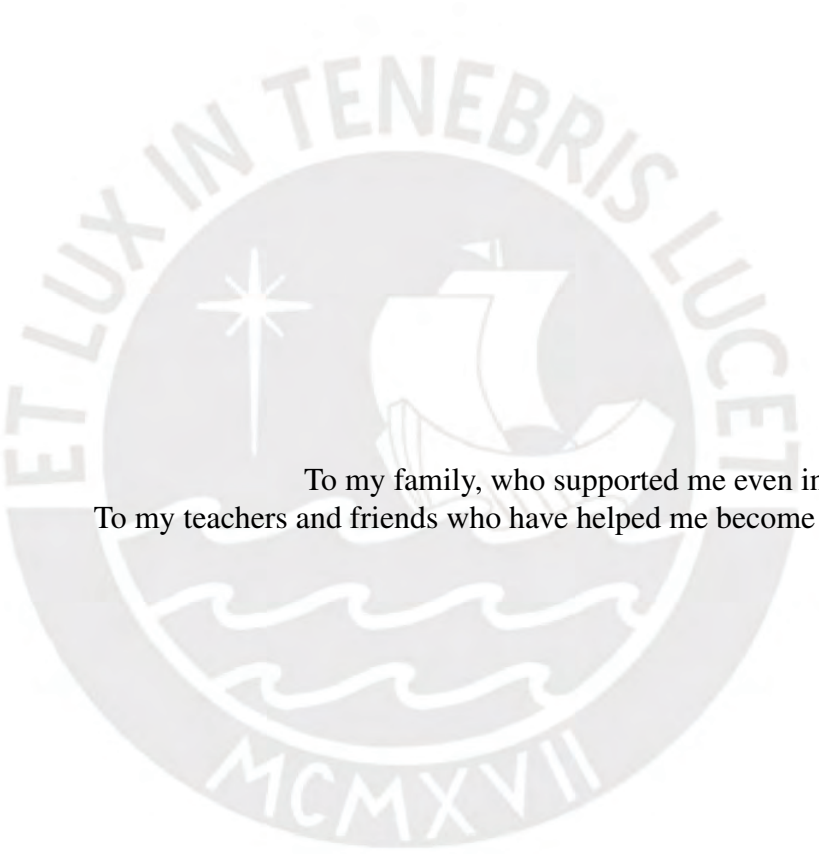
Abstract

Sparsity is a widely valued property in several machine learning and signal processing applications as it enables efficient data representation and prevents overfitting in learning models. Despite its broad utility, the use of sparsity to reduce computational demands remains an active research area. In mathematical optimization, sparsity can be promoted in solutions through regularization techniques or by imposing explicit constraints. Recently, techniques such as screening rules have been proposed to exploit sparsity to diminish the computational requirements of large and huge-scale optimization problems. Nevertheless, current state-of-the-art screening methods provide only rough approximations of the solution support, discarding few elements and thus yielding limited computational savings.

This thesis contributes with an extension of safe screening rules based on adaptive thresholding aimed at regularized optimization problems. The proposed thresholding rule is based on the observation that the metric used to identify non-contributing features can be considered to have an approximate unimodal distribution for practical purposes. The proposed approach was embedded into the accelerated proximal gradient (APG) / fast iterative shrinkage thresholding algorithm (FISTA), an optimizer that has recently attracted attention due to its theoretical convergence rate $\mathcal{O}\left(k^{-2}\right)$. To validate the proposed technique, computational experiments were conducted across multiple contexts. Experimental results indicate that the proposed method provides greater speedups than both safe and strong screening rules in their static version (each implemented as a preprocessing stage) across synthetic and real datasets with a speedup of up to 33.9, approximately. The method developed in this thesis was presented in a conference article at the 2023 24th International Conference on Digital Signal Processing (DSP).

Keywords

Screening rules, sparsity, adaptive thresholding



To my family, who supported me even in difficult times.
To my teachers and friends who have helped me become a better person.

Acknowledgments

I would like to thank my advisor, Dr. Paul Rodríguez, for his support and advice. Without his guidance as advisor, this work would not have been possible.

Thanks to my friends at PUCP, specially to those in the DSP and the LIM labs for helping me enjoying my time in the program and the time doing research. Special thanks to Gustavo Silva, Gabriel Ramírez, and Stefano Romero, your friendship and support made the challenges of the master's program much easier to navigate.

Special thanks to my friends abroad, Gustavo Chau and Jorge Quesada. The camaraderie moments we shared reminded me that life is more than just research, helping me stay balanced during difficult times. May your journeys take you to great heights!

Contents

List of Figures

List of Tables

Introduction

Sparse optimization has emerged as a key area of research in signal processing and machine learning. Its benefits include the ability to provide robust solutions in the presence of inexact data and the incorporation of prior knowledge in several problem domains [1, 2]. In machine learning specifically, sparsity also plays a crucial role in preventing overfitting and improving generalization. However, sparse optimization methods are often computationally intensive on their own, and as machine learning and signal processing applications continue to scale, they place even greater demands on processing time and memory. This has prompted the development of techniques aimed at reducing processing time, memory usage, and the overall problem size in optimization tasks. In particular, screening rules [3, 4, 5] have proven effective in sparse optimization by allowing the early identification of features that do not contribute to the solution. Nevertheless, most screening rules are limited by their reliance on surrogate quantities that relax the strict theoretical screening conditions, leading to the potential retention of non-contributing features when evaluating whether a feature can be discarded, thus limiting computational savings.

The contribution of this thesis is a method for enhancing safe screening rules for regularized optimization problems through the introduction of adaptive thresholding. The proposed method stems from the observation that safe screening rules can be interpreted as thresholding over a measure that can reasonably be assumed to be unimodal. Furthermore, the method is versatile as it can be applied to any safe screening rule and thus incorporated into any optimizer. The proposed method was compounded with safe screening rules and embedded in the accelerated proximal gradient (APG) / fast iterative shrinkage thresholding algorithm (FISTA) optimizer as a preprocessing step and tested using several datasets. Experimental results indicate that the proposed technique improves the support estimation, thus yielding greater computational savings with little degradation in the solution. It is important to mention that the proposed method was presented in a conference article [6] at the 2023 IEEE 24th International Conference on Digital Signal Processing.

This document is organized as follows. Chapter 1 outlines the methodological framework of the thesis. Core concepts in convex analysis and the optimization algorithm used in this work are covered in Chapter 2. Chapter 3 reviews state-of-the-art techniques for reducing the size of optimization problems. Chapter 4 introduces the proposed method, providing an intuitive justification. Finally, Chapter 5 presents the computational experiments and their corresponding results, followed by concluding remarks.

Chapter 1

Methodological framework

This chapter provides an overview of the thesis by outlining its motivation, stating the central thesis problem, and detailing the objectives. It also discusses how the contributions of this work will be validated through both theoretical evaluation and experimental validation. Finally, this chapter presents key mathematical notation used throughout this document to ensure clarity and consistency.

1.1 Motivation

With the rise of machine learning in technology, mathematical optimization methods have become the backbone of many applications, ranging from scientific fields such as medicine, astronomy, and geology to practical computational tools such as text translation, speech recognition, and financial modeling. However, optimization algorithms often require intensive computations, involving a large number of operations, which makes it essential to develop mathematical techniques that remain efficient even as the volume of data increases, as seen in many modern applications. In several instances, data also carry inherent structural information or exhibit certain properties, such as redundancy or sparsity [1], that can be leveraged to simplify computations [7]. Sparsity, in particular, is highly desirable in several signal processing and machine learning applications because it results in robust solutions, aligns with prior knowledge, enhances stability, and helps prevent overfitting in models [2]. However, solving optimization problems in large or huge-scale settings poses significant challenges, making algorithms that require evaluating quantities over the entire dataset impractical, further complicating optimization in these contexts. Thereby, problem size reduction techniques, such as screening rules [3, 4, 5], are crucial to reduce the computational burden and making optimization feasible in large-scale applications.

1.2 Statement of the problem

In data-intensive applications, screening techniques [3, 4, 5] are used to reduce the dimensionality of data sets and thus improve computational efficiency. However, current state-of-the-art screening methods face significant limitations when applied to large-scale data. The fast development of both academic research and industry demands methods, several of them optimization-based, that must handle huge amounts of data. In this context, existing screening techniques often fail to eliminate enough features to achieve substantial computational savings, especially in fields where data grows exponentially. This bottleneck slows the development of new technologies and research, driving up financial expenses for processing systems and hindering progress in critical computationally demanding areas such as machine learning, artificial intelligence, and big data analytics [7]. Therefore, it is necessary to explore new methods and adapt existing screening techniques so they can handle the demands of high-dimensional data more effectively. By improving the feature elimination process, significant computational savings can be achieved, facilitating faster innovation and reducing costs for both industry and research sectors. This thesis hypothesizes that, by addressing the limitations of current screening rules, it is possible to refine screening techniques, thereby enhancing computational efficiency in sparse optimization and benefiting a wide range of data-intensive applications.

1.3 Objectives

1.3.1 Main objective

The main goal of this thesis project is to develop a computationally efficient method to enhance state-of-the-art screening techniques aiming to achieve greater computational savings by increasing the number of eliminated features.

1.3.2 Specific objectives

1. Implement sparse optimization problems that can serve as cases of study.
2. Apply state-of-the-art screening techniques to the established cases of study.
3. Propose a novel method to improve the screening support estimation (increasing the number of discarded features).
4. Test the proposed method for the defined cases of study and assess key performance metrics to quantify its effectiveness.

1.4 Validation of contributions

The proposed method will be subjected to both theoretical evaluation and experimental validation. The theoretical evaluation will focus on analyzing the method's computational cost and memory usage. On the other hand, the experimental validation will assess the performance of the method by applying it to a range of optimization problems corresponding to the selected cases of study (which will be fully defined in Chapter 5). These tests will verify the method's ability to yield computational savings and accurately identify non-contributing features across various tasks.

1.4.1 Theoretical evaluation

The theoretical time and space complexity of the proposed method will be thoroughly analyzed to assess its computational efficiency. This analysis is included in Section 4.3. Specifically, the asymptotic efficiency, i.e. how the resource consumption of an algorithm increases as the size of the input increases without bound, will be studied. The time complexity refers to the amount of time that an algorithm requires to run and is determined by counting the number of operations needed for the algorithm to complete its task [8]. This key metric is typically expressed as a function of the input size, providing insights into how the algorithm's performance scales with an increasing amount of data. On the other hand, the space complexity reflects the memory usage of the algorithm, measuring the amount of memory required as a function of the input size. Both time and space complexities are commonly expressed using asymptotic notation. Specifically, the big-O notation ($\mathcal{O}$) is used, since it describes the asymptotic upper bound or the worst-case scenario, within a constant factor, of a function.

1.4.2 Experimental validation

Experimental validation will be performed to assess the computational gains achieved by the proposed method, as well as to determine whether there is any quality loss in the resulting solutions. Chapter 5 thoroughly presents the experiments used to validate the proposed method in this work. A series of tests across different application contexts, ranging from signal processing to machine learning, will be employed to thoroughly characterize the performance of the proposed method.

The evaluation will focus on a few key metrics. First, the cardinality of the solution, i.e. the number of non-zero entries in the optimal solution, will be analyzed as it directly quantifies the immediate impact of applying screening rules. Since the primary goal of introducing the proposed method is to speed up the optimization process, the computational efficiency will be evaluated through the metrics detailed in the following. Total processing time will be recorded. This gives a direct measurement of the computational cost and

allows us to evaluate the effectiveness of variable discarding in reducing the runtime. To quantify the performance improvement when using screening techniques, the speedup, defined as the ratio of the processing time of a previously established baseline method to the processing time of the method under evaluation, will be analyzed.

In addition, quality metrics will be examined for different tasks. In cases where the optimization problem is related to classification or regression, the accuracy in both the training set and test set is evaluated [9]. This allows to measure how well the solution generalizes to unseen data. On the other hand, for signal and image reconstruction tasks, the peak signal-to-noise ratio (PSNR) [10] will be used as a measure of the quality of the reconstructed signal or image compared to a reference ground truth. PSNR is widely used in image processing and measures the ratio between the maximum possible power of a signal and the power of corrupting noise.

1.5 Mathematical notation

- **Variables:** Represented by lowercase letters (e.g., x, ϕ).
- **Vectors (1D tensors):** Denoted by bold lowercase letters (e.g., $\mathbf{x}$, $\boldsymbol{\phi}$).
- **Matrices (2D tensors):** Represented by bold uppercase letters (e.g., $\mathbf{X}$, $\boldsymbol{\Phi}$).
- **Superscripts:** Indicate:
  - The state or value of a variable at a specific iteration.
  - An element in a sequence.
  - Example: $\mathbf{x}^{(k)}$ represents:
    - The value of $\mathbf{x}$ at the k-th iteration.
    - The k-th element in the sequence $\left\{\mathbf{x}^{(k)}\right\}$.
- **Subscripts:** Used for array slicing:
  - Added to the letter to explicitly indicate the slicing type.
  - **For an n-dimensional array, the subscript is an n-element tuple:**
    - $a : b$ – selects elements between indices a and b (interval slicing).
    - **: (colon)** – keeps the entire dimension unchanged (no slicing applied).
    - a – selects only the single element at index a (single-element selection).
    - **Condition:** $b > a > 0$.
  - **Position in the tuple:**
    - Each element in the tuple corresponds to a specific dimension of the array.
    - The k-th position in the tuple determines how the slicing affects the k-th dimension.

Chapter 2

Theoretical background

2.1 Fundamentals on convex analysis

In this section, relevant concepts for the development of this thesis are reviewed. For a more detailed description of the concepts and theorems listed here, see [11].

2.1.1 Basic concepts and properties

2.1.1.1 Convex sets and functions

A set $\mathcal{C}$ is convex if, for any $\theta \in [0,1]$ and points $x_1, x_2 \in \mathcal{C}$, the following holds

$$\theta \cdot x_1 + (1-\theta) \cdot x_2 \in \mathcal{C} \tag{2.1}$$

On the other hand, a function f is convex if, for any $\theta \in [0,1]$ and points $x_1, x_2 \in \text{dom}(f)$, the following holds

$$f(\theta \cdot x_1 + (1-\theta) \cdot x_2) \leq \theta \cdot f(x_1) + (1-\theta) \cdot f(x_2). \tag{2.2}$$

2.1.1.2 Indicator function

The indicator function characterizes the membership of an element in a set. Formally, given a set $\mathcal{C} \subset \mathbb{R}^n$, its indicator function, denoted by $\iota_{\mathcal{C}}$ is defined as

$$\iota_{\mathcal{C}}(\mathbf{u}) = \begin{cases} 0 & \text{if } \mathbf{u} \in \mathcal{C}, \\ +\infty & \text{if } \mathbf{u} \notin \mathcal{C}. \end{cases} \tag{2.3}$$

It is important to note that if $\mathcal{C}$ is a convex set, its indicator function $\iota_{\mathcal{C}}$ is also convex.

2.1.1.3 Subdifferential

The subdifferential is a generalization of the derivative for convex functions that allows for non-uniqueness. Formally, the subdifferential of a function f at $\mathbf{u}$, denoted by $\partial f(\mathbf{u})$, is the set defined by

$$\partial f(\mathbf{u}) = \{\mathbf{g} \in \mathbb{R}^n : \ f(\mathbf{u}) \geq f(\mathbf{v}) + \langle \mathbf{g}, \mathbf{u} - \mathbf{v} \rangle, \ \forall \mathbf{v} \in \mathbb{R}^n\}. \tag{2.4}$$

Geometrically, the subdifferential at a point $\mathbf{u}$ represents the set of all possible slopes of lines that lie below or touch the graph of f at $\mathbf{u}$. If f is differentiable at $\mathbf{u}$, then the subdifferential reduces to a singleton set containing the derivative of f at $\mathbf{u}$.

2.1.1.4 Smoothness

A function is smooth if it exhibits continuous and well-behaved behavior without abrupt changes or irregularities. Formally, a function f is L-smooth if its gradient ∇f is a L-Lipschitz function, i.e. $\forall \mathbf{u}, \mathbf{v} \in \text{dom}(f)$ it follows that

$$\|\nabla f(\mathbf{u}) - \nabla f(\mathbf{v})\| \leq L \|\mathbf{u} - \mathbf{v}\|, \tag{2.5}$$

where L is a parameter known as smoothness constant. This constant quantifies the rate at which the function's gradient can change and provides an upper bound on the function's curvature. A smaller Lipschitz constant L corresponds to a smoother function, where the gradient changes more gradually, while a larger L implies a more irregular behavior.

2.1.1.5 Proximal operator

Given a closed and proper function $f : \mathcal{C} \to \mathbb{R} \cup \{+\infty\}$, the proximal mapping of f for $\mathbf{u} \in \mathcal{C}$ [12], denoted as $\text{prox}_f(\mathbf{u})$, is defined as

$$\text{prox}_f(\mathbf{u}) = \underset{\mathbf{v}}{\text{argmin}} \ \ f(\mathbf{v}) + \frac{1}{2} \|\mathbf{v} - \mathbf{u}\|^2. \tag{2.6}$$

The proximal operator can be interpreted as the calculation of a point that is close to $\mathbf{u}$ with respect to the function f. It is also important mentioning that this operator preserves the convexity of the function f i.e. if f is convex, then $\text{prox}_{\lambda \cdot f}(\mathbf{u})$ for any $\lambda > 0$.

2.1.2 Karush-Kuhn-Tucker conditions

They are also referred as Kuhn-Tucker, optimality or KKT conditions [13]. Given the following general convex program

$$\min_{\mathbf{u}\in\mathbb{R}^n} \quad f(\mathbf{u}) \;\text{ subject to }\; c_i(\mathbf{u}) = 0, \;\; i \in \mathcal{I}, \;\; c_j(\mathbf{u}) \leq 0, \;\; j \in \mathcal{J}, \tag{2.7}$$

with solution continuously differentiable $\hat{\mathbf{u}}$. Then, there exist a Lagrange multiplier vector $\hat{\mathbf{v}}$, such that the following condition is satisfied

$$(\mathbf{0}, \mathbf{0}) \in \partial\mathcal{L}\left(\hat{\mathbf{u}}, \hat{\mathbf{v}}\right), \tag{2.8}$$

where $\mathcal{L}$ is the Lagrangian of the convex program. It is important to mention that the Karush-Kuhn-Tucker conditions can also be stated as

$$\nabla_{\mathbf{u}}\mathcal{L}\left(\hat{\mathbf{u}}, \hat{\mathbf{v}}\right) = 0 \tag{2.9a}$$
$$c_i(\mathbf{u}) = 0, \;\; i \in \mathcal{I} \tag{2.9b}$$
$$c_i(\mathbf{u}) \leq 0, \;\; i \in \mathcal{J} \tag{2.9c}$$
$$\mathbf{v}_i = 0, \;\; i \in \mathcal{I} \tag{2.9d}$$
$$\mathbf{v}_i \cdot c_i(\mathbf{u}) = 0, \;\; i \in \mathcal{I} \cup \mathcal{J}, \tag{2.9e}$$

as is indicated in [11, 14]. It is important to mention that:

- Equation (2.9a) is known as the stationarity condition,
- Equations (2.9b) and (2.9c) are known as the primal feasibility conditions,
- Equation (2.9d) is known as the dual feasibility condition,
- Equation (2.9e) is known as complementary slackness condition.

2.1.3 Duality

2.1.3.1 Fenchel-Legendre transform

The Fenchel-Legendre transform [13] is a mathematical operation that associates a convex function with its dual function. Formally, given a function $f : \mathbb{R}^n \to \mathbb{R}$, the convex conjugate, or Fenchel conjugate, of f is a function $f^* : \mathbb{R}^n \to \mathbb{R}$ defined as

$$f^*(\mathbf{v}) = \sup_{\mathbf{u}\in\mathbb{R}^n} \quad \langle \mathbf{u}, \mathbf{v} \rangle - f(\mathbf{u}). \tag{2.10}$$

The mapping $f \mapsto f^*$ is known as the Fenchel-Legendre transform. Some important properties and relationships associated with the Fenchel-Legendre transform include:

- If f is a convex function, then its Fenchel-Legendre transform f^* is also convex.
- If f is a proper, closed, and convex function, then the convex conjugate of its convex conjugate, denoted as $(f^*)^*$, recovers the original function f i.e. $(f^*)^* = f$.
- The Fenchel-Legendre transform provides a duality relationship between optimization problems. In particular, it relates primal and dual problems and facilitates the derivation of optimality conditions and duality gaps.

2.1.3.2 Primal-dual pair

A primal-dual pair [15] refers to a specific relationship between two closely related optimization problems: the primal problem and the dual problem. In the context of this work, the following primal-dual pair is considered

$$\hat{\mathbf{u}}^{(\lambda)} \in P_\lambda(\mathbf{u}) \coloneqq \underset{\mathbf{u}\in\mathbb{R}^n}{\operatorname{argmin}} \; f(\mathbf{X}\mathbf{u}) + \lambda \cdot g(\mathbf{u}) \tag{2.11a}$$

$$\hat{\mathbf{v}}^{(\lambda)} \in D_\lambda(\mathbf{v}) \coloneqq \underset{\mathbf{v}\in\mathbb{R}^n}{\operatorname{argmax}} \; -f^*(-\lambda \cdot \mathbf{v}) - \lambda \cdot g^*\left(\mathbf{X}^T\mathbf{v}\right), \tag{2.11b}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^m \to \mathbb{R}$ are convex and proper functions with convex conjugates f^* and g^*, respectively, $\mathbf{X} \in \mathbb{R}^{n\times m}$ is matrix (or operator) of the primal-dual pair, and $\lambda > 0$ is the regularization hyperparameter. It is important to mention that $P_\lambda(\mathbf{u})$ and $D_\lambda(\mathbf{v})$ are known as the primal and dual problems with solutions $\hat{\mathbf{u}}^{(\lambda)}$ and $\hat{\mathbf{v}}^{(\lambda)}$, respectively.

2.1.3.3 Duality gap

The duality gap [14] is defined as the difference between the optimal values of the primal and dual problems. Mathematically, given a primal-dual feasible pair of vectors $(\mathbf{u},\mathbf{v}) \in \operatorname{dom}(P_\lambda) \times \operatorname{dom}(D_\lambda)$, the duality gap is defined as

$$\operatorname{gap}_\lambda(\mathbf{u},\mathbf{v}) = P_\lambda(\mathbf{u}) - D_\lambda(\mathbf{v}). \tag{2.12}$$

It is important to mention that the optimal solution to the primal problem cannot be lower than the value obtained by solving the dual problem i.e. $P_\lambda\left(\hat{\mathbf{u}}^{(\lambda)}\right) \geq D_\lambda\left(\hat{\mathbf{v}}^{(\lambda)}\right)$, meaning that the duality gap can only take non-negative values. The duality of the primal-dual pair can be characterized by using the value of the duality gap. Strong duality, a property associated with convex optimization problems, is the situation associated with

the duality gap being zero i.e. $P_\lambda\left(\hat{\mathbf{u}}^{(\lambda)}\right) = D_\lambda\left(\hat{\mathbf{v}}^{(\lambda)}\right)$ and implies that the optimal solution to the primal problem can be found by solving the dual problem, and vice versa. On the other hand, weak duality is a property associated with both non-convex and convex optimization problems and presents itself when the duality gap is greater or equal than zero i.e. $P_\lambda\left(\hat{\mathbf{u}}^{(\lambda)}\right) \geq D_\lambda\left(\hat{\mathbf{v}}^{(\lambda)}\right)$. It is important to mention that due to the weak duality property, the following property holds

$$P_\lambda(\mathbf{u}) - P_\lambda\left(\hat{\mathbf{u}}^{(\lambda)}\right) \leq \text{gap}_\lambda(\mathbf{u}, \mathbf{v}). \tag{2.13}$$

2.1.3.4 Fenchel duality theorem

The Fenchel duality theorem [13], also known as the Fenchel-Moreau duality theorem, is a fundamental result in convex analysis and optimization theory. It establishes a strong duality relationship between a convex function and its convex conjugate function, providing insights into the optimization problem associated with the original function. Formally and in the context of this work, for a primal-dual pair given by (2.11a) and (2.11b), strong duality holds if and only if the following holds

$$\mathbf{X}^T\hat{\mathbf{v}}^{(\lambda)} \in \partial g\left(\hat{\mathbf{u}}^{(\lambda)}\right) \longleftrightarrow \hat{\mathbf{u}}^{(\lambda)} \in \partial g^*\left(\mathbf{X}^T\hat{\mathbf{v}}^{(\lambda)}\right) \tag{2.14a}$$

$$-\lambda\cdot\hat{\mathbf{v}}^{(\lambda)} \in \partial f\left(\mathbf{X}\hat{\mathbf{u}}^{(\lambda)}\right) \longleftrightarrow \mathbf{X}\hat{\mathbf{u}}^{(\lambda)} \in \partial f^*\left(-\lambda\cdot\hat{\mathbf{v}}^{(\lambda)}\right). \tag{2.14b}$$

It is important to mention that (2.14a) and (2.14b) are also referred to as optimality or KKT conditions in recent literature [5].

2.2 Optimization algorithms

In the context of large and huge scale optimization, the challenges of solving optimization problems range from memory limitation, non-linearity, non-smoothness to even non-convex problems. Among several optimizers, gradient descent and its variants are very well-known popular methods applied in those fields in the machine learning and signal processing fields. The class of gradient methods provide cheap computations, convergence guarantees, easy-to-implement parallelization, and efficiency in high dimensions. In the following, relevant details related to the accelerated proximal gradient (APG) method, a popular variant of gradient descent that is the optimizer used in this work, are presented.

2.2.1 Gradient methods

Gradient methods constitute a class of optimization techniques that rely on the information conveyed by the gradient (first-order derivative) to perform the optimization of a function. They are aimed at optimization problems of the form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x}), \tag{2.15}$$

where f is convex and L-smooth. The usage of the gradient provides a nice tradeoff between computational cost and convergence properties that allows for these methods to be applied in different applications.

2.2.1.1 Gradient descent

Gradient descent [16] is a well-known optimization solver and is widely applied to several problems in machine learning and computer vision. It consists of following updates in a direction opposite to ∇f as

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha^{(k)} \cdot \nabla f\left(\mathbf{x}^{(k)}\right). \tag{2.16}$$

It is important to mention that due to its low convergence rate, which is asymptotically $\mathcal{O}\left(k^{-1}\right)$, the application of gradient descent is limited only to a few problems.

2.2.1.2 Nesterov's accelerated gradient

The Nesterov's accelerated gradient [17] is an extension of the standard gradient descent algorithm with the goal of improving convergence speed, especially in situations where the optimization landscape has a high degree of curvature. This algorithm is outlined as two steps and the use of a

$$\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)} - \alpha^{(k)} \cdot \nabla f\left(\mathbf{y}^{(k)}\right) \tag{2.17a}$$

$$t^{(k+1)} = \frac{1+\sqrt{1+4\cdot {t^{(k)}}^2}}{2} \tag{2.17b}$$

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} + \frac{t^{(k)}-1}{t^{(k+1)}} \cdot \left(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right), \tag{2.17c}$$

where $\gamma^{(k)} = \frac{t^{(k)}-1}{t^{(k+1)}}$ is the inertial sequence of the algorithm. This algorithm wields an improved asymptotically convergence rate of $\mathcal{O}\left(k^{-2}\right)$.

2.2.2 Proximal methods

Proximal methods constitute a class of optimization techniques that is considered as an extension of gradient methods commonly used to solve convex optimization problems involving non-smooth functions. In particular, these methods are used to solve optimization problems of the form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; F(\mathbf{x}) \coloneqq f(\mathbf{x}) + g(\mathbf{x}), \tag{2.18}$$

where f is a convex and L-smooth function and g is a non-smooth function. These methods are widely applied in several fields, including machine learning and signal processing.

2.2.2.1 Proximal gradient

Gradient descent was extended, by introducing the proximal operator, to the proximal gradient (PG) algorithm in order to solve problems with non-smooth regularization [18], and it consists of the following update

$$\mathbf{x}^{(k+1)} = \operatorname{prox}_{\alpha^{(k)}\cdot g}\left(\mathbf{x}^{(k)} - \alpha^{(k)}\cdot\nabla f\left(\mathbf{x}^{(k)}\right)\right). \tag{2.19}$$

It is important to mention that when $g(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$, the PG method is known as iterative shrinkage thresholding algorithm (ISTA). It was independently verified that early variants of PG achieve a convergence rate of $\mathcal{O}\left(k^{-1}\right)$ [19, 20, 21].

2.2.2.2 Accelerated proximal gradient

Conversely to the PG, the Nesterov's accelerated gradient algorithm can be extended using the proximal operator. The extension for problems with non-smooth regularization is known as accelerated proximal gradient (APG), also known as the fast iterative shrinkage thresholding algorithm (FISTA) [22] allows, to solve optimization problems of the form (2.18) by using the following updates

$$\mathbf{x}^{(k+1)} = \operatorname{prox}_{\alpha^{(k)}\cdot g}\left(\mathbf{y}^{(k)} - \alpha^{(k)}\cdot\nabla f\left(\mathbf{y}^{(k)}\right)\right) \tag{2.20a}$$

$$t^{(k+1)} = \frac{1+\sqrt{1+4\cdot {t^{(k)}}^2}}{2} \tag{2.20b}$$

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} + \frac{t^{(k)}-1}{t^{(k+1)}}\cdot\left(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right), \tag{2.20c}$$

where $f, g : \mathbb{R}^n \to \mathbb{R}$ are convex functions, ∇f is L-Lipschitz continuous, and $g(.)$ could be non-smooth. APG uses information of the two previous iterates of the solutions, achieving a convergence rate of $\mathcal{O}\left(k^{-2}\right)$.

2.2.3 Adaptive step size techniques

Several strategies have been proposed to accelerate gradient-based and proximal methods. For instance, methods for selecting the step size $\alpha^{(k)}$ have been a key focus since they allow significantly reducing the number of iterations required for convergence, although often at the cost of increased per-iteration computation. One such strategy is the line search method for step size selection [14], however, although this method is effective, it incurs in a high computational cost, as an additional optimization problem

$$\alpha_{\text{LS}}^{(k)} = \underset{\alpha \in \mathbb{R}^+}{\operatorname{argmin}} \; f\left(\mathbf{x}^{(k)} - \alpha \cdot \nabla f\left(\mathbf{x}^{(k)}\right)\right) \tag{2.21}$$

is required to be solved per iteration. Another widely applied choice to compute the step size is the Cauchy step size selection technique [16], tailored for quadratic problems, it proposes to use the following expression

$$\alpha_{\text{Ca}}^{(k)} = \frac{\left\|\mathbf{g}^{(k)}\right\|_2^2}{\left\langle \mathbf{g}^{(k)}, \mathbf{Q}\mathbf{g}^{(k)} \right\rangle}, \tag{2.22}$$

where $\mathbf{g}^{(k)} = \nabla f\left(\mathbf{x}^{(k)}\right)$, and $\mathbf{Q}$ represents the Hessian of the function f at $\mathbf{x}^{(k)}$. It is also worth mentioning that a multiplicative variant of the Cauchy step size [23] has been proposed and is expressed as

$$\alpha_{\text{Ca-mul}}^{(k)} = c \cdot \frac{\left\|\mathbf{g}^{(k)}\right\|_2^2}{\left\langle \mathbf{g}^{(k)}, \mathbf{Q}\mathbf{g}^{(k)} \right\rangle}, \tag{2.23}$$

where $c \in [0, 2]$. Furthermore, in order to exploit the sparsity in the solutions of certain optimization problems, a support-aware variant of the Cauchy step size technique [24], initially proposed in the context of iterative hard thresholding, is given by

$$\alpha_{\text{Ca-supp}}^{(k)} = \frac{\left\|\mathbf{s}^{(k)} \odot \mathbf{g}^{(k)}\right\|_2^2}{\left\langle \mathbf{s}^{(k)} \odot \mathbf{g}^{(k)}, \mathbf{Q}\left(\mathbf{s}^{(k)} \odot \mathbf{g}^{(k)}\right) \right\rangle}, \tag{2.24}$$

where $\odot$ represents Hadamard product i.e. pointwise multiplication, and the support (estimated at the k-th iteration) is given by $\mathbf{s}^{(k)} = \mathbf{1}\left(\mathbf{x}^{(k)}\right)$ with $\mathbf{1}(.)$ representing a pointwise

indicator function. On the other hand, the Barzilai-Borwein technique [25] proposes the two following step size sequences

$$\alpha_{\text{BB-v1}}^{(k)} = \frac{\left\langle \mathbf{z}^{(k-1)}, \mathbf{y}^{(k-1)} \right\rangle}{\left\| \mathbf{z}^{(k-1)} \right\|_2^2} \qquad (2.25a) \qquad \alpha_{\text{BB-v2}}^{(k)} = \frac{\left\| \mathbf{y}^{(k-1)} \right\|_2^2}{\left\langle \mathbf{z}^{(k-1)}, \mathbf{y}^{(k-1)} \right\rangle}, \qquad (2.25b)$$

where $\mathbf{y}^{(k-1)} = \mathbf{x}^{(k-1)} - \mathbf{x}^{(k)}$ and $\mathbf{z}^{(k-1)} = \mathbf{g}^{(k-1)} - \mathbf{g}^{(k)}$, and more recently, a variant based on these was proposed in [26] as

$$\alpha_{\text{BB-v3}}^{(k)} = \sqrt{\alpha_{\text{BB-v1}}^{(k)} \cdot \alpha_{\text{BB-v2}}^{(k)}} = \frac{\left\| \mathbf{y}^{(k-1)} \right\|_2}{\left\| \mathbf{z}^{(k-1)} \right\|_2}. \qquad (2.26)$$

It is worth to mention that in [27], the previous original proposals were combined into the Cauchy-Barzilai-Borwein step size selection, which reads as

$$\alpha_{\text{CBB}}^{(k)} = \begin{cases} \alpha_{\text{Ca-mod}}^{(k)} & \text{if } \mod(k,2) = 0, \\ \alpha_{\text{CBB}}^{(k-1)} & \text{otherwise.} \end{cases} \qquad (2.27)$$

Chapter 3

Problem size reduction

High-dimensional regularized optimization problems present significant computational challenges due to their intensive resource requirements. As the demand for efficient optimizers grows, particularly with the adoption of sparsity-inducing regularization, several techniques have been developed to improve performance. Among these methods, feature screening stands out: a collection of optimizer-agnostic rules designed to reduce problem complexity by identifying and eliminating non-active entries in the solution of sparse optimization problems. This not only enhances computational efficiency but also adapts to a variety of iterative solvers. This chapter outlines the key problem size reduction strategies and provides a synthesis of the current state of the art in screening methodologies.

3.1 Facial reduction

Facial reduction is a method designed to aid in the size reduction of optimization problems. It was originally conceived to be applied to linear programming [28, 29], and extended to conic programming [30], and further adapted to semidefinite programming [31]. All these problems can be written as generic conic programming problems i.e. in the following form

$$\min_{\mathbf{u}\in\mathcal{E}} \ \langle \mathbf{c}, \mathbf{u} \rangle \ \text{ subject to } \ \mathbf{A}\mathbf{u} = \mathbf{b}, \ \ \mathbf{u} \succeq_{\mathcal{K}} \mathbf{0}_{\text{card}(\mathcal{E})}, \tag{3.1}$$

where $\mathcal{K}$ is a convex cone defined in $\mathcal{E}$, and $\mathbf{a} \succeq_{\mathcal{K}} \mathbf{b}$ is a relational operator that implies $\mathbf{a} - \mathbf{b} \in \mathcal{K}$. Furthermore, depending on the choice of $\mathcal{K}$, the optimization problem can categorized as linear or semidefinite programming when $\mathcal{K}$ is chosen to be $\mathbb{R}^n_+$ (non-negative orthant), or $\mathbb{S}^n_+$ (semidefinite cone), respectively.

Facial reduction simplifies the feasible region of an optimization problem by working with its auxiliary systems (constraints) to eliminate redundancies. This technique is grounded in the Theorem of the Alternative [32], which provides the mathematical basis

for facial reduction computations in problems that do not satisfy the Slater condition, that is, problems that are not strictly feasible. The formal characterization of facial reduction is based on the concept of a face. A face is a subset of the feasible set of the convex cone $\mathcal{K}$ that is itself a convex cone. The minimal face of $\mathcal{K}$ containing a set $\mathcal{M} \subseteq \mathcal{K}$ is defined as the intersection of all faces of $\mathcal{K}$ that contain $\mathcal{M}$, and is denoted by face$(\mathcal{M},\mathcal{K})$. Let $\mathcal{F}_p$ be the feasible region of the primal problem. The minimal face of $\mathcal{K}$ that contains $\mathcal{F}_p$ is denoted by face$(\mathcal{F}_p,\mathcal{K})$. Then, facial reduction allows replacing $\mathcal{K}$ with face$(\mathcal{F}_p,\mathcal{K})$, and $\mathcal{E}$ with span$(\mathcal{K})$ in problem (3.1).

In linear programming, this procedure is often required only once. However, for conic and semidefinite programming, repeated applications may be necessary. The result is a lower-dimensional primal problem that automatically satisfies Slater's condition, because $\mathcal{F}_p$ intersects the relative interior of face$(\mathcal{F}_p,\mathcal{K})$. Finally, it is important to mention that the facial reduction procedure is conceptual. This is due to the fact that its implementation involves either demonstrating that the auxiliary systems are inconsistent or solving them to machine precision at each iteration.

Recent advancements in facial reduction have yielded significant progress. For instance, the relaxation of auxiliary problems to achieve solvable systems—since solving the auxiliary system is not always simpler than the original problem—was explored in [33] for semidefinite programming and its impact on facial reduction. Additionally, [34] demonstrated how combining facial reduction with other techniques aligns well with an alternating direction method of multipliers (ADMM) solver [35] to address semidefinite programming with double non-negativity constraints. Moreover, the facial reduction technique has proven valuable in applications such as matrix completion [36], robust principal component analysis [37], and challenging combinatorial problems [38].

3.2 Correlation screening

Correlation screening is a class of dimensionality reduction techniques for sparse problems that is based on a property coined as sure screening. Sure screening denotes the property where, in the context of solving a sparse problem, all essential variables reliably survive the elimination process, with probability tending to one. An important screening technique of this class that was formulated for linear models i.e.

$$\mathbf{y} = \mathbf{X}\boldsymbol{\omega} + \boldsymbol{\varepsilon}, \tag{3.2}$$

where $\mathbf{y} \in \mathbb{R}^m$ is the response of the model, $\mathbf{X} \in \mathbb{R}^{m\times n}$ is a random design matrix, $\boldsymbol{\omega} \in \mathbb{R}^n$ is the parameter of the model, and $\boldsymbol{\varepsilon} \in \mathbb{R}^m$ represents i.i.d. random errors, is sure independence screening (SIS) [39] which uses correlation learning which filters out the features that are weakly correlated with the response. Then, the SIS test to evaluate the k-th of the design matrix reads as

$$\left|\mathbf{X}_{k,:}^{T}\mathbf{y}\right| > t \rightarrow \hat{\boldsymbol{\omega}}_k = 0, \tag{3.3}$$

where $\hat{\boldsymbol{\omega}}$ is the obtained estimation, and t is a selected critical threshold (such that the number of selected variables is smaller than a prescribed proportion of the features). Several improvements to the standard SIS test have emerged over time. For instance, [39] introduces iterative variants of SIS to enhance performance. To address challenges posed by heavy-tailed distributions and highly skewed responses, a robust version of SIS was proposed in [40]. The SIS method has also been adapted to the compressed sensing framework, as demonstrated in [41]. Additionally, [42] proposes incorporating a conditioning set (prior knowledge) to reduce false positives and false negatives when dealing with highly correlated covariates.

3.3 Screening rules

Screening rules [3] are a class of methods designed to exploit sparsity to reduce the size of sparse optimization problems by leveraging the structure of the associated primal-dual pairs. These techniques have proven effective for eliminating features by estimating and tracking the support of optimization problem solutions, or for sample elimination by identifying redundancies among training examples. The focus in what follows is specifically on feature screening techniques, commonly referred to in the literature as screening rules.

In general, modern screening techniques arise as a consequence of the generalized Kuhn-Tucker theorem i.e. Karush-Kuhn-Tucker (KKT) conditions [13]. For an optimization problem with a sparsity-inducing regularization and matrix system $\mathbf{X} \in \mathbb{R}^{m\times n}$, its associated primal-dual pair can be expressed as

$$\hat{\boldsymbol{\Omega}}^{(\lambda)} \in \underset{\boldsymbol{\Omega}\in\mathbb{R}^{n\times p}}{\operatorname{argmin}} \; P_\lambda(\boldsymbol{\Omega}) \coloneqq f(\mathbf{X}\boldsymbol{\Omega}) + \lambda \cdot g(\boldsymbol{\Omega}) \tag{3.4a}$$

$$\hat{\boldsymbol{\Theta}}^{(\lambda)} \in \underset{\boldsymbol{\Theta}\in\mathbb{R}^{m\times p}}{\operatorname{argmax}} \; D_\lambda(\boldsymbol{\Theta}) \coloneqq -f^*(-\lambda\cdot\boldsymbol{\Theta}) - \lambda \cdot g^*\left(\mathbf{X}^T\boldsymbol{\Theta}\right), \tag{3.4b}$$

where f and g are convex functions (f is L-smooth and g is non-smooth and separable) with Fenchel conjugates f^* and g^*, respectively, and $\hat{\boldsymbol{\Omega}}^{(\lambda)}$ and $\hat{\boldsymbol{\Theta}}^{(\lambda)}$ are the primal and dual solutions. It is important to mention the convenience of proceeding with this screening framework while taking into account the separability assumption i.e. the structure-promoting regularization term g is considered to be separable (this is the case for a broad class of applications in signal processing and machine learning), to perform an analysis for each feature or group of features in an independent fashion. Then, the considered optimality conditions with the separability consideration can be expressed as

$$\mathbf{X}_{:,k}^T \hat{\mathbf{\Theta}}^{(\lambda)} \in \partial g_k \left(\hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} \right) \qquad (3.5a) \qquad -\lambda \cdot \hat{\mathbf{\Theta}}^{(\lambda)} = \nabla f \left(\mathbf{X} \hat{\mathbf{\Omega}}^{(\lambda)} \right). \qquad (3.5b)$$

Conversely, the basic screening rule to discard the k-th feature arises from (3.5a) [5, Theorem 3], and takes the following form

$$\left\| \mathbf{X}_{:,k}^T \hat{\mathbf{\Theta}}^{(\lambda)} \right\| < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} = \mathbf{0}_p. \tag{3.6}$$

It is important to mention that the screening rule (3.6) is not applicable since $\hat{\mathbf{\Theta}}^{(\lambda)}$ is not available until the end of the optimization procedure.

3.3.1 Safe screening rules

Screening rules are considered safe if they ensure the identification and elimination of only those features or variables that do not contribute to the optimal solution [5]. These rules arise from the basic rule (3.6) by defining a region $\mathcal{R}$ (known as safe region) that contains the dual solution $\hat{\mathbf{\Theta}}^{(\lambda)}$. This leads to the following evaluation rule for the k-th feature

$$\max_{\mathbf{\Theta} \in \mathcal{R}} \ \left\| \mathbf{X}_{:,k}^T \mathbf{\Theta} \right\| < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} = \mathbf{0}_p. \tag{3.7}$$

It is important to note that by incorporating the safe region constraint, the rule (3.7) does not depend on $\hat{\mathbf{\Theta}}^{(\lambda)}$, enabling its practical implementation. This is a key feature that makes the rule computationally feasible. However, to make the rule applicable in practice, a specific point in the dual solution space, denoted as $\mathbf{P} \in \mathcal{R}$ (referred to as the dual point), must be selected. This choice of a dual point simplifies the evaluation of the screening rule while ensuring its correctness within the defined safe region leading to

$$\left\| \mathbf{X}_{k,:}^T \mathbf{P} \right\| + \left\| \mathbf{X}_{k,:} \right\| \cdot \max_{\mathbf{\Theta} \in \mathcal{R}} \ \| \mathbf{\Theta} - \mathbf{P} \| < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} = \mathbf{0}_p. \tag{3.8}$$

The rule expressed in (3.8) is still not applicable as is, but it is possible to find an upper bound dependent on the dual region and other elements

$$\max_{\mathbf{\Theta} \in \mathcal{R}} \ \| \mathbf{\Theta} - \mathbf{P} \| \leq d_{P_\lambda, D_\lambda, \mathcal{R}} \left(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda \right),$$

resulting in the following generic rule for the computational application of safe screening in evaluating the k-th feature of the primal problem

$$\left\| \mathbf{X}_{k,:}^T \mathbf{P} \right\| + \left\| \mathbf{X}_{k,:} \right\| \cdot d_{P_\lambda, D_\lambda, \mathcal{R}} \left(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda \right) < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} = \mathbf{0}_p, \tag{3.9}$$

where $d_{P_\lambda,D_\lambda,\mathcal{R}}(\mathbf{X},\mathbf{Z},\boldsymbol{\Omega},\boldsymbol{\Theta},\lambda)$ is a function dependent on the structure of the primal-dual pair (P_λ,D_λ) and the geometry of the dual region $\mathcal{R}$. On the other hand, the dual point $\mathbf{P}$ can be obtained using the dual solution $\boldsymbol{\Theta}$, and its calculation depends on the geometry of $\mathcal{R}$. In turn, the primal-dual link, which provides the following easy-to-compute way to calculate the dual solution, can be written as

$$\boldsymbol{\Theta}_{:,k} = -\frac{\nabla f_k\left(\mathbf{X}_{:,k}\boldsymbol{\Omega}\right)}{\max\left(\lambda,\lambda_{\max}\right)}, \tag{3.10}$$

where $\lambda_{\max}$ is the smallest value that λ could take that corresponds to an all-zero primal solution. Safe screening rules have been proposed to accelerate the solution of prevalent problems in signal processing and machine learning, such as LASSO [43, 44, 45, 46], sparse logistic regression [47, 48, 49], and sparse support vector machines [50, 51, 52], among others. Studied choices for defining the safe region include ball (often referred to as a sphere in the literature) [3, 53], ellipsoid [54], dome [55, 56], and polytope [57]. Despite the variety of these choices, the safe sphere remains the most widely adopted due to its minimal computational overhead in optimization algorithms. Among the many safe screening techniques discussed in the literature, the most prominent are the gap safe screening rules, which use the duality gap to determine the function $d_{P_\lambda,D_\lambda,\mathcal{R}}(\mathbf{X},\mathbf{Z},\boldsymbol{\Omega},\boldsymbol{\Theta},\lambda)$, enabling efficient identification and elimination of irrelevant features.

3.3.2 Strong screening rules

Introduced in [58], the relaxation of the initial safe screening rules led to the proposal of a new set of rules, now known as strong screening rules. These rules, while allowing for a higher number of excluded features compared to the original ones, come with the trade-off of potentially discarding relevant (i.e. active) features. This relaxation involves relaxing the non-expansiveness condition on the gradient of the data fitting term, unless stronger assumptions about the dictionary or system matrix are made.

The initial strong screening rule was aimed at ℓ_1-norm regularized statistical models, i.e. problems of the form

$$\min_{\boldsymbol{\omega}\in\mathbb{R}^n} \quad \frac{1}{2}\left\|\mathbf{X}\boldsymbol{\omega}-\mathbf{y}\right\|_2^2+\lambda\left\|\boldsymbol{\omega}\right\|_1, \tag{3.11}$$

where λ is the regularization hyperparameter, $\mathbf{X}$ is the matrix of the system, $\mathbf{y}$ is the observed data. Considering that the minimum hyperparameter for which the solution of the problem is all-zero is given by $\lambda_{\max}=\left\|\mathbf{X}^T\mathbf{y}\right\|_\infty$, then the basic strong screening rule to evaluate the k-th feature is expressed as

$$\left|\mathbf{X}_{k,:}^T\mathbf{y}\right| < 2\cdot\lambda-\lambda_{\max} \rightarrow \hat{\boldsymbol{\omega}}_k = 0, \tag{3.12}$$

where $\hat{\boldsymbol{\omega}}$ is the solution of the problem. This rule is obtained by replacing $\frac{\left\|\mathbf{X}_{k,:}\right\|_2 \|\mathbf{y}\|_2}{\lambda_{\max}}$ with one in the safe screening rule proposed for LASSO in [3]. A simple intuition of this comes from analyzing standardized dictionaries or features i.e. $\left\|\mathbf{X}_{k,:}\right\|_2 = 1$, so the following Cauchy-Schwarz inequality holds

$$\frac{\|\mathbf{y}\|_2}{\lambda_{\max}} \geq 1. \tag{3.13}$$

Strong screening rules were extended to tackle other classes of optimization problems. A generalization of strong rules was proposed in [58] for a broader class of sparse regularized problems, so it can be applied to optimization problems of the form given by (3.4a). The extended strong rule to evaluate of the k-th feature reads as

$$\left\|\mathbf{X}_{k,:}^T \nabla f\left(\mathbf{0}_{m\times p}\right)\right\| < 2\cdot\lambda - \lambda_{\max} \rightarrow \hat{\boldsymbol{\Omega}}_{k,:} = \mathbf{0}_p, \tag{3.14}$$

where $\hat{\boldsymbol{\Omega}}$ is the solution to the problem. Finally, since strong screening rules have the risk of incorrectly discarding relevant features, they must be supplemented with a check on the discarded features to perform corrections. The verification is done by checking the KKT conditions for all entries of the solution. If there are no violations, the solution is correct. Otherwise, the discarded entries that violate the KKT conditions are included in the active set of entries and the optimization problem is solved with the updated set of active entries.

3.3.3 Non-static screening rules

The safe and strong screening rules discussed above can be classified as static screening rules, as they are typically applied prior to solving the optimization problem, that is, as a preprocessing step. Their effectiveness can be enhanced by adapting them into methods that operate during the optimization process itself, allowing for a more refined estimation of the active support and, consequently, the elimination of additional features. These adaptations, known as non-static screening rules, adjust the estimated active set dynamically as the optimization proceeds, leading to a more effective feature selection. Below, the most noteworthy adaptations of static screening rules into non-static variants are presented.

3.3.3.1 Sequential screening rules

Warm start [14] is an optimization technique that allows to accelerate the convergence and improve the efficiency of solving a particular optimization problem by leveraging the information gained from solving similar or related problems. Formally, the standard warm start sequence of problems, i.e. the problems that are solved sequentially, are defined as

$$\min_{\boldsymbol{\Omega}\in\mathbb{R}^{n\times p}} \quad f\left(\mathbf{X}\boldsymbol{\Omega}\right)+\lambda^{(i)}\cdot g\left(\boldsymbol{\Omega}\right), \tag{3.15}$$

where $i=0,\ldots,n$, and all problems in the sequence have hyperparameters that are elements of a strictly decreasing sequence i.e. $\lambda^{(i-1)}>\lambda^{(i)}$. Here, the solution of the $(i-1)$-th problem is used to initialize the i-th problem, thus improving convergence.

In the context of warm start, the static screening techniques can be adapted to eliminate non-contributing features to refine the support of the solution for each problem to solve. This is done by using the solution of the previous problem in the warm start sequence, i.e. the solution of the $(i-1)$-th problem is used to compute the elements to perform screening in the i-th problem instead of using an all-zero solution (which is typically used in static screening), thus achieving a more refined support. This idea applied to safe screening rules was explored in [3, 59], leading to sequential safe screening rules (also known as recursive safe screening rules). In particular, the most popular sequential safe screening rules are based on the gap safe sphere, and its application to the i-th problem in the sequence of optimization problems reads as

$$\left\|\mathbf{X}_{k,:}^{T}\hat{\boldsymbol{\Theta}}^{(i-1)}\right\|+r^{(i)}\cdot\left\|\mathbf{X}_{k,:}\right\|<1\rightarrow\hat{\boldsymbol{\Omega}}_{k,:}^{(i)}=\mathbf{0}_{p}, \tag{3.16}$$

where the center $\hat{\boldsymbol{\Theta}}^{(i-1)}$ and the radius $r^{(i)}$ are calculated as

$$\hat{\boldsymbol{\Theta}}^{(i-1)}=-\frac{\nabla f\left(\mathbf{X}\hat{\boldsymbol{\Omega}}^{(i-1)}\right)}{\max\left(\lambda^{(i-1)},g^{\circ}\left(\mathbf{X}^{T}\nabla f\left(\mathbf{X}\hat{\boldsymbol{\Omega}}^{(i-1)}\right)\right)\right)} \tag{3.17a}$$

$$r^{(i)}=\sqrt{\frac{2}{L\cdot{\lambda^{(i)}}^{2}}\cdot\mathrm{gap}_{\lambda^{(i)}}\left(\hat{\boldsymbol{\Omega}}^{(i-1)},\hat{\boldsymbol{\Theta}}^{(i-1)}\right)}, \tag{3.17b}$$

where $\hat{\boldsymbol{\Omega}}^{(i)}$ and $\hat{\boldsymbol{\Theta}}^{(i)}$ are the primal and dual solutions that correspond to the i-th problem in the sequence, respectively, and g° is the dual norm of g. Conversely, the application of this idea to strong screening, coined as sequential strong screening rules, was proposed in [58]. For a warm start sequence of problems given by (3.15), the sequential strong screening to evaluate the k-th feature in the i-th problem in the warm start sequence reads as

$$\left\|\mathbf{X}_{k,:}^{T}\nabla f\left(\mathbf{X}\hat{\boldsymbol{\Omega}}^{(i-1)}\right)\right\|<2\cdot\lambda^{(i)}-\lambda^{(i-1)}\rightarrow\hat{\boldsymbol{\Omega}}_{k,:}^{(i)}=\mathbf{0}_{p}, \tag{3.18}$$

where $\hat{\boldsymbol{\Omega}}^{(i-1)}$ is the solution of the $(i-1)$-th problem in the warm start sequence i.e. when the regularization hyperparameter is $\lambda^{(i-1)}$.

3.3.3.2 Dynamic screening rules

An important note on safe screening rules in its static form is that their formulation uses the initial values of primal and dual solutions, $\mathbf{\Omega}$ and $\mathbf{\Theta}$, which are not obtained but selected. Standard practice involves choosing $\mathbf{\Omega}$ as all-zero and $\mathbf{\Theta}$ as its correspondent obtained by the primal-dual link. By applying the underlying principle of sequential screening, the performance of the screening procedure can be enhanced by strategically selecting these values resulting in a more accurate support estimation. This idea was explored in [60, 61, 62] for updating the parameters associated with screening by using the primal and dual solutions at a given iteration in the optimization procedure, thus performing screening along with the iterations of the optimization algorithm leading to dynamic screening. The resulting technique is known as dynamic since it allows to refine the number of discarded features as the algorithm progresses towards the optimal solution.

The dynamic screening idea was elaborated into a general framework for gap safe screening, in particular for a safe sphere region, in [5]. Furthermore, a theoretical guarantee was provided there for that particular case, since demonstrating that the sequence of safe spheres produced by dynamic screening converges as the iterations of the optimization solver go on. In particular, the rule of dynamic screening using a gap safe sphere to evaluate the k-th feature at the i-th iteration of the optimization solver, read as

$$\left\|\mathbf{X}_{k,:}^{T}\mathbf{\Theta}^{(i-1)}\right\| + r^{(i)} \cdot \left\|\mathbf{X}_{k,:}\right\| < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:} = \mathbf{0}_{p}, \tag{3.19}$$

where the center $\mathbf{\Theta}^{(i-1)}$ and the radius $r^{(i)}$ are calculated as

$$\mathbf{\Theta}^{(i-1)} = -\frac{\nabla f\left(\mathbf{X}\mathbf{\Omega}^{(i-1)}\right)}{\max\left(\lambda, g^{\circ}\left(\mathbf{X}^{T}\nabla f\left(\mathbf{X}\mathbf{\Omega}^{(i-1)}\right)\right)\right)} \tag{3.20a}$$

$$r^{(i)} = \sqrt{\frac{2}{L\cdot\lambda^{2}}\cdot \mathrm{gap}_{\lambda}\left(\mathbf{\Omega}^{(i-1)}, \mathbf{\Theta}^{(i-1)}\right)}, \tag{3.20b}$$

where $\mathbf{\Omega}^{(i)}$ and $\mathbf{\Theta}^{(i)}$ are the values of the primal and dual solutions at the i-th iteration of the optimization solver, respectively, and g° is the dual norm of g. Other notable works on dynamic screening include [63] that introduces dynamic sasvi, a dynamic extension of the sasvi screening technique. Finally, it is important to mention that it is often recommended to perform dynamic screening only every few iterations due to the expensive computational cost of the duality gap calculation [5].

3.3.4 Screening rules for non-traditional regularization schemes

Screening rules have been extended to handle sparse regularizers that do not conform to the separability property or the convexity condition typical of classical screening approaches. For instance, [64] addressed screening rules for non-convex sparsity-promoting

regularizers by employing an iterative majorization-minimization (MM) approach, embedding a screening rule in its inner solver, with a condition to propagate screened variables across MM iterations. In [65], screening rules were adapted to problems involving both ℓ_2-norm regularization and either ℓ_0-pseudonorm regularization or ℓ_0-pseudonorm-based constraints by applying convex perspective relaxations to the mixed programming formulations of these problems. Similarly, [66] applied screening rules to ℓ_0-pseudonorm regularized least squares, particularly in the context of bound-and-branch solvers. Screening rules for constrained problems were adapted in [4], using the indicator function of the constrained set as regularization and employing the Wolfe gap function instead of the conventional duality gap. The challenge of non-separability posed by overlapping group norms was tackled by screening rules introduced in [67], which proposed evaluating each group using the dual polytope projection screening approach, while accounting for overlapping groups inclusive of the one under evaluation. Finally, strong screening rules [68] and safe screening rules [69] were proposed for Sorted *L*-One Penalized Estimation (SLOPE), a class of optimization problems that use the convex but non-smooth sorted ℓ_1-norm as a sparse regularizer, which imposes heavier penalties on larger entries of the solution.

3.3.5 Beyond the limits of classical screening rules

Screening rules have also benefited from both theoretical advancements and practical findings, enabling their application in broader setups or enhancing their efficiency within traditional schemes. For example, it was shown in [70] that screening rules can function effectively in non-convex optimization contexts, as long as the loss function is convex over certain subsets of its domain (locally strongly convex), removing the requirement for global convexity in the optimization problem. Moreover, in the context of LASSO, screening rules were extended into Adascreen [71], incorporating multiple half-space constraints on the dual optimal solution in addition to the traditional sphere constraint (safe sphere screening rule), forming an ensemble of screening rules. A hybrid approach, combining safe and strong rules, was introduced as hybrid safe-strong rules (HSSR) in [72], integrating safe screening into a strong screening procedure to reduce the need for post-convergence KKT checks on features eliminated by safe rules. Similarly, the combination of safe screening with a relaxation procedure for tracking zero and non-zero entries in non-negative elastic net regularized least squares was explored in [73]. Lastly, region-free safe screening rules were introduced in [74] for ℓ_1-norm regularized convex problems, comparing the values of relaxed primal and dual functions at feasible points.

3.4 Squeezing rules

Spreading the information of a signal uniformly over its representation coefficients is also a desirable property in certain applications such as the design of robust analog-to-digital

conversion schemes [75, 76] or to reduce the peak-to-average power ratio (PAPR) in multicarrier transmissions [77, 78]. This property leads to democratic or antisparse representations in the sense that the representations coefficients are dense with several saturated entries [79]. The associated optimization problem to find such representations is known as antisparse coding and is modeled as a ℓ_∞-norm constrained problem, though it is more common to use the regularized form

$$\min_{\boldsymbol{\omega}\in\mathbb{R}^n} \quad \frac{1}{2}\|\mathbf{X}\boldsymbol{\omega}-\mathbf{y}\|_2^2+\lambda\|\boldsymbol{\omega}\|_\infty, \tag{3.21}$$

where $\mathbf{X}\in\mathbb{R}^{m\times n}$ is the coding matrix, $\mathbf{y}\in\mathbb{R}^n$ represents the signal measurements, and the antisparse representation of the signal denoted by $\hat{\boldsymbol{\omega}}$ is the solution to the problem.

It is important to note that the problem given by (3.21) does not produce a sparse solution, nevertheless, the uniformly spreading property of its solution can also be exploited to perform feature elimination in a similar fashion to screening rules. Squeezing is a technique proposed in [80, 81] that allows to discard features for the antisparse coding class of problems by identifying the entries that were guaranteed to saturate i.e. to take the allowed maximum value. Furthermore, these squeezing rules were found to relate to the safe screening rules (thus dubbing them as safe squeezing rules) due to them complying with the safe property (since the both of them only discard non-contributing features) and the similarity in their structure.

Formally, the squeezing rules aim at estimating the set of indices of saturated entries in the solution of the antisparse coding problem as $\mathfrak{u}^{(\lambda)}=\mathfrak{u}_+^{(\lambda)}\cup\mathfrak{u}_-^{(\lambda)}$, where

$$\mathfrak{u}_+^{(\lambda)}=\left\{k:\hat{\boldsymbol{\omega}}_k^{(\lambda)}=\left\|\hat{\boldsymbol{\omega}}^{(\lambda)}\right\|_\infty\right\} \tag{3.22a}$$

$$\mathfrak{u}_-^{(\lambda)}=\left\{k:\hat{\boldsymbol{\omega}}_k^{(\lambda)}=-\left\|\hat{\boldsymbol{\omega}}^{(\lambda)}\right\|_\infty\right\}, \tag{3.22b}$$

with $\mathfrak{u}_+^{(\lambda)}$ and $\mathfrak{u}_-^{(\lambda)}$ representing the sets of indices of positive and negative saturated entries in the solution, respectively. Clearly, it is not possible to know $\mathfrak{u}^{(\lambda)}$ without solving the antisparse coding problem, so, in practice, the aforementioned sets are estimated as $\mathfrak{u}$, $\mathfrak{u}_+$, and $\mathfrak{u}_-$, where $\mathfrak{u}\subseteq\mathfrak{u}^{(\lambda)}$, $\mathfrak{u}_+\subseteq\mathfrak{u}_+^{(\lambda)}$, $\mathfrak{u}_-\subseteq\mathfrak{u}_-^{(\lambda)}$, and $\mathfrak{u}=\mathfrak{u}_+\cup\mathfrak{u}_-$. In order to exploit the knowledge of the estimated set $\mathfrak{u}$, the problem given by (3.21) must be rewritten as

$$\min_{(\mathbf{w},w)\in\mathbb{R}^{\operatorname{card}(\mathcal{C})}\times\mathbb{R}} \quad \frac{1}{2}\left\|\begin{bmatrix}\mathbf{X}_{\mathcal{C}} & \bar{\mathbf{x}}\end{bmatrix}\begin{bmatrix}\mathbf{w}\\ w\end{bmatrix}-\mathbf{y}\right\|_2^2+\lambda\cdot w \ \text{ subject to } \ -w\leq\mathbf{w},\ \ w\leq\mathbf{w}, \tag{3.23}$$

where w represents the absolute value of the saturated entries, $\mathcal{C}$ is the set of indices that correspond to the entries in the solution that are guaranteed not to saturate, the vector $\bar{\mathbf{x}}$ is calculated as the difference between the columns of $\mathbf{X}$ that correspond to the positively saturated entries and those that correspond to the negatively saturated entries, i.e.

$$\bar{\mathbf{x}} = \sum_{k \in \mathfrak{l}_+} \mathbf{X}_{k,:} - \sum_{k \in \mathfrak{l}_-} \mathbf{X}_{k,:}. \tag{3.24}$$

It is important to note that the dimensionality of the solution of the reduced problem, given by (3.23), has been effectively reduced since $\operatorname{card}(\mathcal{C}) = n - \operatorname{card}(\mathfrak{l})$.

The dual of the reduced antisparse coding problem is given by

$$\max_{\boldsymbol{\theta} \in \mathbb{R}^m} \ \frac{1}{2}\|\mathbf{y}\|_2^2 - \frac{1}{2}\|\boldsymbol{\theta} - \mathbf{y}\|_2^2 \ \text{ subject to } \ \left\|\mathbf{X}_{\mathcal{C}}^T \boldsymbol{\theta}\right\|_1 + \bar{\mathbf{x}}^T \boldsymbol{\theta} \leq \lambda, \tag{3.25}$$

and the basic squeezing rule for evaluating the k-th feature is expressed as

$$\left|\mathbf{X}_{k,:}^T \hat{\boldsymbol{\theta}}^{(\lambda)}\right| > 0 \rightarrow k \in \mathfrak{l}^{(\lambda)}_{\operatorname{sign}\left(\mathbf{X}_{k,:}^T \hat{\boldsymbol{\theta}}^{(\lambda)}\right)}, \tag{3.26}$$

where $\hat{\boldsymbol{\theta}}^{(\lambda)}$ is the solution of the dual problem of the antisparse coding problem with hyperparameter λ. In [80, 81] it is demonstrated that, in a similar fashion to safe screening rules, by introducing a region in the dual space that is guaranteed to contain the dual solution, the basic squeezing rules become computationally applicable. In particular, the case of the safe sphere i.e. ball $\mathcal{B}(\mathbf{c}, r)$ was studied, resulting in the safe sphere squeezing rules

$$\left|\mathbf{X}_{k,:}^T \mathbf{c}\right| > r\left\|\mathbf{X}_{k,:}\right\|_2 \rightarrow k \in \mathfrak{l}^{(\lambda)}_{\operatorname{sign}\left(\mathbf{X}_{k,:}^T \mathbf{c}\right)}, \tag{3.27}$$

where the center $\mathbf{c}$ and radius r can be determined as

$$r = \sqrt{2 \cdot \operatorname{gap}_\lambda(\mathbf{w}, w, \boldsymbol{\theta})} \tag{3.28a}$$

$$\mathbf{c} = \boldsymbol{\theta} = \begin{cases} \mathbf{z} & \text{if } \ \left\|\mathbf{X}_{\mathcal{C}}^T \mathbf{z}\right\|_1 + \bar{\mathbf{x}}^T \mathbf{z} \leq 0, \\ \frac{\lambda}{\left\|\mathbf{X}_{\mathcal{C}}^T \mathbf{z}\right\|_1 + \bar{\mathbf{x}}^T \mathbf{z}} \cdot \mathbf{z} & \text{if } \ \left\|\mathbf{X}_{\mathcal{C}}^T \mathbf{z}\right\|_1 + \bar{\mathbf{x}}^T \mathbf{z} > 0, \end{cases} \tag{3.28b}$$

where $\operatorname{gap}_\lambda$ is the duality gap between the reduced antisparse problem and its dual for the hyperparameter λ. Finally, it is important to mention that the extension to dynamic squeezing for gap safe sphere squeezing is also studied in [81], where dual points that are produced along the iterations of the optimization procedure can be used to update the safe sphere, resulting in a dynamic procedure with properties similar to dynamic screening.

Chapter 4

Proposed method

This chapter outlines the main contributions of this thesis. Initially, an intuitive exploration of the quantities employed to evaluate features is provided, offering insight into their roles and relationships. Following this, a practical approach for applying adaptive thresholding on this collection is introduced, aiming to enhance feature evaluation accuracy. It is worth to note that the contributions discussed in this chapter can be found in [6].

4.1 Screening metric

4.1.1 Definition

Screening rules become computationally feasible by incorporating a region $\mathcal{R}$ in the dual space, which is guaranteed to contain the solution to the dual problem (3.4b). This allows the selection of a dual point $\mathbf{P} \in \mathcal{R}$, enabling the derivation of computable quantities. Specifically, the screening metric to evaluate the k-th feature in the primal problem (3.4a) is denoted by $\mathbf{\Phi}_k$ and defined as

$$\mathbf{\Phi}_k = \left\|\mathbf{X}_{k,:}^T \mathbf{P}\right\| + \left\|\mathbf{X}_{k,:}\right\| \cdot d_{P_\lambda, D_\lambda, \mathcal{R}}\left(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda\right), \tag{4.1}$$

where $d_{P_\lambda, D_\lambda, \mathcal{R}}\left(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda\right)$ represents a parameter that depends on the geometry of the region $\mathcal{R}$ and the structure of the primal-dual pair (3.4a)-(3.4b). Then, the screening test to evaluate the k-th feature, using the defined screening metric, is given by

$$\mathbf{\Phi}_k < 1 \rightarrow \hat{\mathbf{\Omega}}_{k,:}^{(\lambda)} = \mathbf{0}_p, \tag{4.2}$$

where $\hat{\mathbf{\Omega}}^{(\lambda)}$ is the solution of the primal problem. This condition ensures that features not contributing to the optimal solution can be safely eliminated.

4.1.2 Analysis of the distribution of the screening metric

It is important to note how the dual problem (3.4b) is structured. It consists of two primary components: a data fidelity term, which ensures that its solution fits the observed information, and a regularization term, which imposes a dependency to the matrix of features $\mathbf{X}$. Following this reasoning, the dual solution $\hat{\mathbf{\Theta}}^{(\lambda)}$ represents the ideal or correct information that the primal solution $\hat{\mathbf{\Omega}}^{(\lambda)}$ should correspond to, effectively acting as a reference for the optimization problem. The correlation between $\hat{\mathbf{\Theta}}^{(\lambda)}$ and the k-th feature can be denoted as $\mathbf{\Gamma}_k$, which can be calculated as

$$\mathbf{\Gamma}_k = \left\| \mathbf{X}_{k,:}^T \hat{\mathbf{\Theta}}^{(\lambda)} \right\|, \tag{4.3}$$

and explains how relevant is the k-th feature with respect to $\hat{\mathbf{\Omega}}^{(\lambda)}$. In sparse optimization, the values collected in $\mathbf{\Gamma}$ can be interpreted as follows: non-contributing features have low $\mathbf{\Gamma}_k$ values, as their contribution to the final solution is minimal or non-existent. If the primal solution $\hat{\mathbf{\Omega}}^{(\lambda)}$ is highly sparse, meaning most features are not relevant, the distribution of $\mathbf{\Gamma}_k$ will have a large proportion of low values. As such, under the stated assumptions, the distribution of the quantities collected in $\mathbf{\Gamma}$ can be assumed to be unimodal, where the peak (mode) corresponds to non-contributing features.

While $\mathbf{\Gamma}$ provides crucial insights into the sparsity of the solution, its direct use is impractical because computing $\mathbf{\Gamma}$ requires the final solution $\hat{\mathbf{\Theta}}^{(\lambda)}$, which is only available after the optimization process is completed. To address this limitation, an alternative approach is needed. In the context of safe screening, the introduction of the quantities $\mathbf{\Psi}$, defined as

$$\mathbf{\Psi}_k = \max_{\mathbf{\Theta} \in \mathcal{R}} \ \left\| \mathbf{X}_{k,:}^T \mathbf{\Theta} \right\|, \tag{4.4}$$

aids into deriving computationally feasible alternatives to directly relying on $\mathbf{\Gamma}$. As demonstrated in [5], these quantities are bounded by the inequality

$$\mathbf{\Gamma}_k \leq \mathbf{\Psi}_k \leq \mathbf{\Gamma}_k + \left\| \mathbf{X}_{k,:} \right\| \cdot \operatorname{diam}(\mathcal{R}), \tag{4.5}$$

where $\operatorname{diam}(\mathcal{R})$ refers to the diameter of the region in the dual space $\mathcal{R}$, representing the distance between its farthest points. The following relation plays a crucial role in the final outcome, even if it is not explicitly stated in this reasoning

$$\max_{\mathbf{\Theta} \in \mathcal{R}} \ \|\mathbf{\Theta} - \mathbf{P}\| \leq d_{P_\lambda, D_\lambda, \mathcal{R}}(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda),$$

where $d_{P_\lambda, D_\lambda, \mathcal{R}}(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}, \mathbf{\Theta}, \lambda)$ denotes a distance related to the dual region $\mathcal{R}$, such as the radius of a ball. By leveraging this relationship and the expression in (4.5), along with the

following relations

$$\left\|\mathbf{X}_{k,:}\mathbf{P}\right\| \leq \boldsymbol{\Psi}_k \qquad d_{P_\lambda, D_\lambda, \mathcal{R}}\left(\mathbf{X}, \mathbf{Z}, \boldsymbol{\Omega}, \boldsymbol{\Theta}, \lambda\right) \leq \operatorname{diam}\left(\mathcal{R}\right) \qquad D = \max_k \ \left\|\mathbf{X}_{k,:}\right\|,$$

where $\mathbf{P} \in \mathcal{R}$, it can be shown that each entry of the screening metric $\boldsymbol{\Phi}$ is bounded as

$$\boldsymbol{\Gamma}_k \leq \boldsymbol{\Phi}_k \leq \boldsymbol{\Gamma}_k + 2D \cdot \operatorname{diam}\left(\mathcal{R}\right). \tag{4.6}$$

Following the unimodality assumption for $\boldsymbol{\Gamma}$, it is reasonable to assume that $\boldsymbol{\Phi}$ exhibits approximate unimodality, its mode is similarly bounded as

$$\boldsymbol{\Gamma}_{\text{mode}} \leq \boldsymbol{\Phi}_{\text{mode}} \leq \boldsymbol{\Gamma}_{\text{mode}} + 2D \cdot \operatorname{diam}\left(\mathcal{R}\right), \tag{4.7}$$

which supports the notion that $\boldsymbol{\Phi}$ inherits the unimodal structure of $\boldsymbol{\Gamma}$. This makes $\boldsymbol{\Phi}$ a valid surrogate that is feasible to compute, as it does not depend on the solution of the optimization problem, with exploitable properties for practical implementations.

4.2 Generalized screening rule

In the context of safe screening rules, one challenge lies in the occurrence of false negatives, where non-contributing features are mistakenly identified as contributing due to the use of surrogate quantities $\boldsymbol{\Phi}_k$ instead of the more accurate $\boldsymbol{\Gamma}_k$. Since $\boldsymbol{\Phi}_k \geq \boldsymbol{\Gamma}_k$, this discrepancy can lead to errors in feature selection. To mitigate this, the proposed approach leverages the approximate unimodality of $\boldsymbol{\Phi}$ to establish a threshold, thereby refining the screening rule. Specifically, the evaluation rule for the k-th feature reads as

$$\boldsymbol{\Phi}_k < t \rightarrow \hat{\boldsymbol{\Omega}}^{(\lambda)}_{k,:} = \mathbf{0}_p, \tag{4.8}$$

where t is determined based on the distribution of $\boldsymbol{\Phi}$. This generalization enhances the precision of the screening process.

4.3 Adaptive thresholding for the screening metric

4.3.1 Formulation of the thresholding rule

In order to compute the threshold t, the unimodality assumption for the distribution of $\boldsymbol{\Phi}$ is used, then, its histogram, from which an idealized though faithful representation can be

observed in Figure 1, can be analyzed to formulate a rule for the computation of t. A basic criteria would be to take a threshold greater than $\mathbf{\Phi}_{\text{mode}}$ to discard with great probability the values associated with non-contributing features. Nevertheless, in practice, this strategy can be either too conservative or too aggressive depending on the location of the threshold for safe screening (which is equal to one).

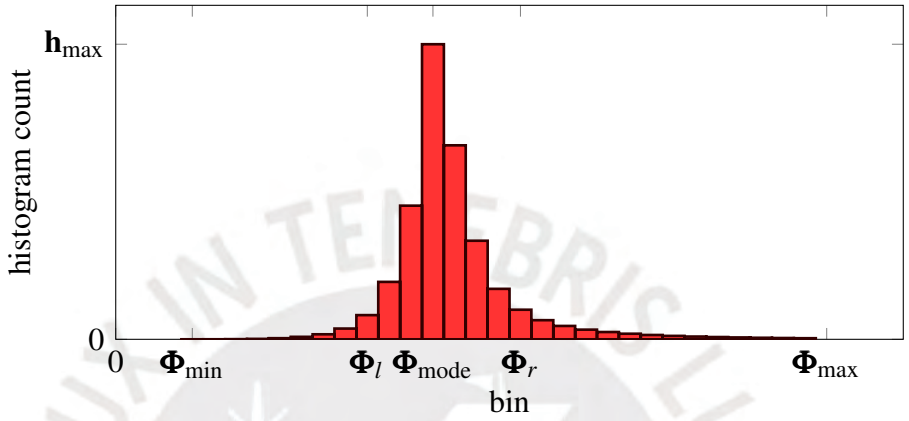


Figure 1: Idealized representation of the histogram of $\mathbf{\Phi}$

The following values can be extracted from the histogram of $\mathbf{\Phi}$

$$\mathbf{\Phi}_{\text{min}} \leq \mathbf{\Phi}_l \leq \mathbf{\Phi}_{\text{mode}} \leq \mathbf{\Phi}_r \leq \mathbf{\Phi}_{\text{max}}, \tag{4.9}$$

where $\mathbf{\Phi}_r$ is a value corresponding to the threshold for unimodal data that can be computed by using techniques such as [82], and $\mathbf{\Phi}_l$ can be computed in a similar fashion using the left part of the histogram.

The proposed criteria for the computation of t takes into account the shape of the histogram of $\mathbf{\Phi}$, and is crafted such that the resulting value of t is greater than one (to achieve more computational savings than safe screening). Before analyzing the histogram of $\mathbf{\Phi}$ to formulate a rule for the computation of t, it is important to note that if $\mathbf{\Phi}_{\text{max}} < 1$, then the solution of the problem, denoted by $\hat{\mathbf{\Omega}}^{(\lambda)}$, is all-zero, since safe thresholding should discard all features, while if $\mathbf{\Phi}_{\text{max}} = 1$, then, computing a threshold is not necessary. Thereby, the following cases for the computation of t can be distinguished:

- $\mathbf{\Phi}_{\text{mode}} \leq 1 < \mathbf{\Phi}_{\text{max}}$: This indicates that $\hat{\mathbf{\Omega}}^{(\lambda)}$ is highly sparse, then t can be chosen in $]\max(1, \mathbf{\Phi}_r), \mathbf{\Phi}_{\text{max}}[$. In practice, for $0 < \alpha_0 < 1$, the following rule can be applied

$$t = \alpha_0 \cdot \max(1, \mathbf{\Phi}_r) + (1 - \alpha_0) \cdot \mathbf{\Phi}_{\text{max}}. \tag{4.10}$$

- $1 < \mathbf{\Phi}_{\text{mode}}$ and the histogram of $\mathbf{\Phi}$ is not greatly left skewed: This indicates that $\hat{\mathbf{\Omega}}^{(\lambda)}$ is moderately sparse. If the histogram of $\mathbf{\Phi}$ is greatly right skewed, then an

aggressive strategy can be applied. Thus t can be reasonably chosen in $[\mathbf{\Phi}_{\text{mode}}, \mathbf{\Phi}_r]$. Then, by considering $0 \leq \alpha_1 \leq 1$, t can be calculated as

$$t = \alpha_1 \cdot \mathbf{\Phi}_{\text{mode}} + (1 - \alpha_1) \cdot \mathbf{\Phi}_r. \tag{4.11}$$

On the other hand, if the histogram of $\mathbf{\Phi}$ is not predominantly skewed, t can be reasonably chosen in $]\max(1, \mathbf{\Phi}_l), \mathbf{\Phi}_{\text{mode}}]$. Then, by considering $0 \leq \alpha_2 < 1$, t can be calculated as

$$t = \alpha_2 \cdot \max(1, \mathbf{\Phi}_l) + (1 - \alpha_2) \cdot \mathbf{\Phi}_{\text{mode}}. \tag{4.12}$$

- $\mathbf{\Phi}_l \leq 1 < \mathbf{\Phi}_{\text{mode}}$ and the histogram of $\mathbf{\Phi}$ is greatly left skewed: This indicates that $\hat{\mathbf{\Omega}}^{(\lambda)}$ could be moderately sparse, and $]1, \mathbf{\Phi}_{\text{mode}}]$ can be considered a reasonable range for t. In practice, it suffices to apply, for $0 \leq \alpha_3 < 1$, the following rule

$$t = \alpha_3 + (1 - \alpha_3) \cdot \mathbf{\Phi}_{\text{mode}}. \tag{4.13}$$

- $1 < \mathbf{\Phi}_l$ and the histogram of $\mathbf{\Phi}$ is greatly left skewed: This indicates that $\hat{\mathbf{\Omega}}^{(\lambda)}$ is not sparse or lowly sparse. Considering the latter case, t can be reasonably chosen in $]\max(1, \mathbf{\Phi}_{\text{min}}), \mathbf{\Phi}_l]$. Then, by considering $0 \leq \alpha_4 < 1$, t can be calculated as

$$t = \alpha_4 \cdot \max(1, \mathbf{\Phi}_{\text{min}}) + (1 - \alpha_4) \cdot \mathbf{\Phi}_l. \tag{4.14}$$

In summary, the rule for the calculation of the threshold is

$$t = \begin{cases} 1 & \text{if } \mathbf{\Phi}_{\text{max}} \leq 1, \\ \alpha_0 \cdot \max(1, \mathbf{\Phi}_r) + (1 - \alpha_0) \cdot \mathbf{\Phi}_{\text{max}} & \text{if } \mathbf{\Phi}_{\text{mode}} \leq 1 < \mathbf{\Phi}_{\text{max}}, \\ \alpha_1 \cdot \mathbf{\Phi}_{\text{mode}} + (1 - \alpha_1) \cdot \mathbf{\Phi}_r & \text{if } s > s_L, \\ \alpha_2 \cdot \max(1, \mathbf{\Phi}_l) + (1 - \alpha_2) \cdot \mathbf{\Phi}_{\text{mode}} & \text{if } |s| \leq s_L, \\ \alpha_3 + (1 - \alpha_3) \cdot \mathbf{\Phi}_{\text{mode}} & \text{if } \mathbf{\Phi}_l \leq 1 < \mathbf{\Phi}_{\text{mode}}, \\ \alpha_4 \cdot \max(1, \mathbf{\Phi}_{\text{min}}) + (1 - \alpha_4) \cdot \mathbf{\Phi}_l & \text{if } 1 < \mathbf{\Phi}_l, \end{cases} \tag{4.15}$$

where s is the skewness of the distribution of $\mathbf{\Phi}$, and s_L is a threshold used to determine the degree of skewness. It is worth remarking on the meaning of skewness in the context of the proposed method. By taking into account the unimodality assumption in the distribution of $\mathbf{\Phi}$, the magnitude and sign of the skewness indicate the relative position of the mode in the distribution of the screening metric $\mathbf{\Phi}$. Finally, it is important to mention that in this work the skewness is estimated using the second Pearson's skewness coefficient [83], which is defined, for a set of data points $\mathbf{x}$, as

$$s = \frac{3 \cdot (\mu_{\mathbf{x}} - m_{\mathbf{x}})}{\sigma_{\mathbf{x}}}, \tag{4.16}$$

where $\mu_{\mathbf{x}}$, $m_{\mathbf{x}}$ and $\sigma_{\mathbf{x}}$ represent the mean, median, and standard deviation of $\mathbf{x}$, respectively.

4.3.2 Complexity analysis

The following analysis assumes a matrix of features $\mathbf{X} \in \mathbb{R}^{m \times n}$ and an optimization variable $\mathbf{\Omega} \in \mathbb{R}^{n \times p}$. The main computational challenge arises from the histogram computation, which in its naive form requires $\mathcal{O}(m \cdot p)$ sums to accumulate frequencies. To determine $\mathbf{\Phi}_l$ and $\mathbf{\Phi}_h$ following the approach presented in [82] and assuming a histogram with r bins, $\mathcal{O}(r)$ pointwise operations are necessary. Additionally, the branches required in the threshold calculation (4.15) involve up to $\mathcal{O}(1)$ pointwise operations. As a result, the total computational complexity of the method is $\mathcal{O}(m \cdot p + r)$.

In terms of memory requirements, calculating a histogram with r bins for an input of size $m \cdot p$ results in a memory complexity of $\mathcal{O}(m \cdot p + r)$. The memory cost to determine $\mathbf{\Phi}_l$ and $\mathbf{\Phi}_h$ is $\mathcal{O}(1)$, and the memory complexity for the thresholding procedure is $\mathcal{O}(1)$. Therefore, the overall memory complexity of the method is also $\mathcal{O}(m \cdot p + r)$.

In comparison, safe screening methods typically involve matrix multiplications, which come with a computational complexity of $\mathcal{O}(m \cdot p)$ and a memory complexity of $\mathcal{O}(m \cdot p)$, where n represents the number of features. Considering that the proposed method consists basically in histogram computation and pointwise operations on the histogram and that $r << m \cdot p$, it introduces minimal overhead in terms of computation and memory and thus serves as a computationally lightweight enhancement to safe screening.

Chapter 5

Computational experiments and results

This chapter presents the computational evaluation of the proposed method using a range of signal processing and machine learning models. The purpose of the proposed experimental validation is to assess the effectiveness of the method under different sparsity-promoting regularization regimes and across diverse datasets. To this end, several case studies are considered, reflecting both element-wise and group-wise sparsity scenarios, and selected to illustrate the versatility and robustness of the method in practical settings. The chapter begins by outlining the cases of study and their mathematical details, and the datasets used, followed by a description of the experimental setup. Then it presents the numerical results and concludes with a discussion of the key findings.

5.1 Cases of study

The application of the proposed method spans several domains in signal processing and machine learning, particularly in relation to how different sparsity-promoting regularization techniques influence task performance. These applications can be broadly grouped into two categories:

- models using individual sparsity regularization to recover signals or extract features by enforcing component-level sparsity, such as
  - basis pursuit denoising,
  - binary classification;
- models leveraging group sparsity where sparsity is enforced across multiple signals or labels to promote consistency in sparsity patterns, including
  - joint sparse reconstruction,

- MEG / EEG source imaging,
- multiclass classification.

Each one of these categories is discussed in more detail in the following subsections.

5.1.1 Element-wise sparse regularization

The problem of inducing sparsity in model parameters can be framed as an optimization problem that minimizes a loss function subject to regularization promoting sparse solutions. In general, the class of problems addressed here involves minimizing an objective function of the form

$$\min_{\boldsymbol{\omega}\in\mathbb{R}^n} \; f(\mathbf{X}\boldsymbol{\omega}) + \lambda \cdot g(\boldsymbol{\omega}), \tag{5.1}$$

where $\boldsymbol{\omega} \in \mathbb{R}^n$ is the optimization variable, $\mathbf{z} \in \mathbb{R}^m$ is the observed data, $\mathbf{X} \in \mathbb{R}^{m\times n}$ is the design matrix, and the regularization term considered in this work is the ℓ_1-norm i.e.

$$g(\boldsymbol{\omega}) = \|\boldsymbol{\omega}\|_1 = \sum_{k=0}^{n-1} |\boldsymbol{\omega}_k|. \tag{5.2}$$

The two models considered within this class are the least absolute shrinkage and selection operator (LASSO) and sparse binary logistic regression. LASSO, originally introduced in [84], is widely applied in contexts such as signal recovery and denoising. It applies ℓ_1-norm regularization to the least squares problem, effectively driving some coefficients to zero and promoting sparse solutions. This property makes LASSO particularly useful in feature selection, especially when only a subset of features is expected to significantly influence the outcome. On the other hand, sparse binary logistic regression [85] is tailored for binary classification tasks, such as image classification. Like traditional logistic regression, it models the probability of binary outcomes using a logistic function. However, by incorporating sparsity-inducing regularization, it highlights the most relevant predictors, enhancing both interpretability and computational efficiency in cases with several potential predictors. Thus, sparse binary logistic regression not only aids in accurate prediction but also helps identify the most important features in classification tasks.

A concise overview of the mathematical elements used in gap safe sphere screening rules, as described in [86], for problems of the form (5.1) is given in Table I. A critical component is the sigmoid function $\boldsymbol{\sigma}$, which is applied to a vector $\boldsymbol{\omega} \in \mathbb{R}^n$, defined as

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\omega}) = \frac{1}{1+\exp(\boldsymbol{\omega})}. \tag{5.3}$$

This transformation ensures that each element of the vector $\mathbf{x}$ is mapped to a positive value lower than one, a characteristic often used in binary classification tasks. Another essential element is the negative binary entropy function, denoted nh. For a scalar value $x \in \mathbb{R}$, the function $\mathrm{nh}(x)$ is defined as

$$\mathrm{nh}(x) = \begin{cases} x \cdot \log(x) + (1-x) \cdot \log(1-x) & \text{if } 0 \leq x \leq 1, \\ +\infty & \text{if otherwise.} \end{cases} \tag{5.4}$$

Table I: Mathematical elements for safe screening rules for pointwise sparse norm regularized problems. It is worth to note that the dual variable $\boldsymbol{\theta} \in \mathbb{R}^m$ is a vector.

Mathematical element	**LASSO**	**Sparse binary logistic regression**
$f(\boldsymbol{\omega})$	$\frac{1}{2}\|\boldsymbol{\omega} - \mathbf{z}\|_2^2$	$-\sum_k \mathbf{z}_k \cdot \log(\boldsymbol{\sigma}_k) + (1-\mathbf{z}_k) \cdot \log(1-\boldsymbol{\sigma}_k)$
$f^*(\boldsymbol{\theta})$	$\frac{1}{2}\|\boldsymbol{\theta} + \mathbf{z}\|_2^2 - \frac{1}{2}\|\mathbf{z}\|_2^2$	$\sum_k \mathrm{nh}(\boldsymbol{\theta}_k + \mathbf{z}_k)$
$\nabla f(\boldsymbol{\omega})$	$\boldsymbol{\omega} - \mathbf{z}$	$\boldsymbol{\sigma} - \mathbf{z}$
$\lambda_{\max}$	$\left\|\mathbf{X}^T \mathbf{z}\right\|_\infty$	$\left\|\mathbf{X}^T \left(\frac{1}{2} \cdot \mathbf{1}_m - \mathbf{z}\right)\right\|_\infty$
L	1	$\frac{1}{4}$

5.1.2 Group sparse regularization

The problems addressed here involve learning structured sparsity patterns across multiple related tasks, typically by solving a regularized optimization problem of the form

$$\min_{\boldsymbol{\Omega} \in \mathbb{R}^{n \times p}} \quad f(\mathbf{X}\boldsymbol{\Omega}) + \lambda \cdot g(\boldsymbol{\Omega}), \tag{5.5}$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{n \times p}$ is the optimization variable, $\mathbf{Z} \in \mathbb{R}^{m \times p}$ is the observed data, $\mathbf{X} \in \mathbb{R}^{m \times n}$ is the design matrix. In particular, this work analyzes the case where the applied regularization is the $\ell_{2,1}$-norm, defined as

$$g(\boldsymbol{\Omega}) = \|\boldsymbol{\Omega}\|_{2,1} = \sum_{k=0}^{m-1} \left\|\boldsymbol{\Omega}_{:,k}\right\|_2. \tag{5.6}$$

Two primary models of interest within this framework are multitask LASSO and sparse multinomial logistic regression. Multitask LASSO builds on the original LASSO by extending it to scenarios where multiple related tasks are solved simultaneously. Unlike traditional LASSO, multitask LASSO [85] employs group sparsity (also known as group

LASSO), enforcing sparsity at the level of groups of coefficients rather than individual entries. This group-level regularization encourages the selection or exclusion of entire groups of related features, which is particularly useful in situations where tasks are expected to share common predictive features. By doing so, multitask LASSO ensures a more interpretable solution, as it identifies features that are relevant across multiple tasks. In contrast, sparse multinomial logistic regression [85] is designed for classification problems that involve more than two classes. Similarly to its binary counterpart, this model estimates the probability distribution over several classes using a multinomial logistic function. However, what distinguishes sparse multinomial logistic regression is its focus on sparsity, making it more interpretable by selecting a subset of relevant predictors for each class. This approach proves valuable in applications where there are many potential features, as it not only predicts the most likely class, but also highlights which features are most influential for each category, improving both model interpretability and generalization.

The elements required for gap safe sphere screening rules, as described in [86], to be applied to problems of the form (5.5) are presented in Table II. One key component is the softmax function, which transforms a matrix $\mathbf{X}$ by converting its rows into probability distributions. The softmax function is defined as

$$\mathbf{softmax}\left(\mathbf{X}\right) = \begin{bmatrix} \frac{\mathbf{exp}\left(\mathbf{X}_{0,:}\right)^T}{\sum_{k=0}^{n-1}\mathbf{exp}\left(\mathbf{X}_{0,k}\right)} & \cdots & \frac{\mathbf{exp}\left(\mathbf{X}_{m-1,:}\right)^T}{\sum_{k=0}^{n-1}\mathbf{exp}\left(\mathbf{X}_{m-1,k}\right)} \end{bmatrix}^T, \tag{5.7}$$

where each row is normalized, making them suitable for use as probabilities in classification models. Another important concept is the negative entropy function, denoted as **NH**, which for a vector $\mathbf{x} \in \mathbb{R}^n$, is defined as

$$\mathbf{NH}\left(\mathbf{x}\right) = \begin{cases} \sum_{k=0}^{n-1}\mathbf{x}_k \cdot \log\left(\mathbf{x}_k\right) & \text{if } \sum_{k=0}^{n-1}\mathbf{x}_k = 1 \text{ and } \mathbf{x}_k > 0, \\ +\infty & \text{if otherwise.} \end{cases} \tag{5.8}$$

Table II: Mathematical elements for safe screening rules for group sparse norm regularized problems. It is worth to note that dual variable $\mathbf{\Theta} \in \mathbb{R}^{m\times p}$ is a matrix.

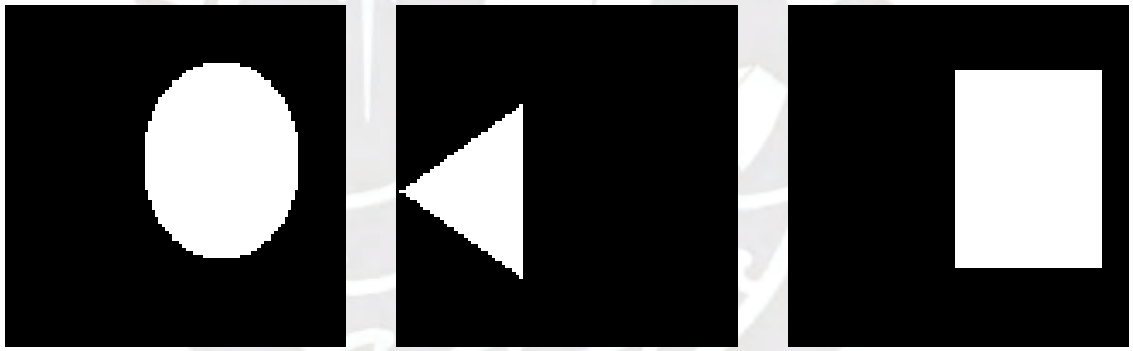
Mathematical element	**Multitask LASSO**	**Sparse multinomial logistic regression**
$f\left(\mathbf{\Omega}\right)$	$\frac{1}{2}\left\|\mathbf{\Omega}-\mathbf{Z}\right\|_F^2$	$-\sum_k\sum_l\left[\mathbf{Z}\odot\mathbf{softmax}\left(\mathbf{\Omega}\right)\right]_{k,l}$
$f^*\left(\mathbf{\Theta}\right)$	$\frac{1}{2}\left\|\mathbf{\Theta}+\mathbf{Z}\right\|_F^2-\frac{1}{2}\left\|\mathbf{Z}\right\|_F^2$	$\sum_k\mathbf{NH}\left(\mathbf{\Theta}_{k,:}+\mathbf{Z}_{k,:}\right)$
$\nabla f\left(\mathbf{\Omega}\right)$	$\mathbf{\Omega}-\mathbf{Z}$	$\mathbf{softmax}\left(\mathbf{\Omega}\right)-\mathbf{Z}$
$\lambda_{\max}$	$\left\|\mathbf{X}^T\mathbf{Z}\right\|_{2,\infty}$	$\left\|\mathbf{X}^T\left(\frac{1}{p}\cdot\mathbf{1}_{n\times p}-\mathbf{Z}\right)\right\|_{2,\infty}$
L	1	1

5.2 Datasets

The experiments detailed in this chapter make use of data coming from different sources. In the following, important details of the datasets used in this work are briefly presented.

5.2.1 Basic shapes dictionary

A synthetic dictionary consisting of m atoms is computationally generated, where each atom represents a $n \times n$ grayscale image. The dictionary is overcomplete, meaning that the number of atoms exceeds twice the total number of pixels in each image, i.e. $m \geq 2 \cdot n^2$. Each atom in the dictionary corresponds to a basic geometric shape: ellipse, triangle, rectangle, pentagon, hexagon, or heptagon. The variations between atoms come from differences in size and centering (location) for the same geometric shape. In this specific work, a dictionary of 20000 atoms is generated, with each atom being a binary image of size 100×100, where white shapes are placed against a black background. One example of each geometric shape that was generated for the dictionary can be observed in Figure 2.



(a) Ellipse (b) Triangle (c) Rectangle

(d) Pentagon (e) Hexagon (f) Heptagon

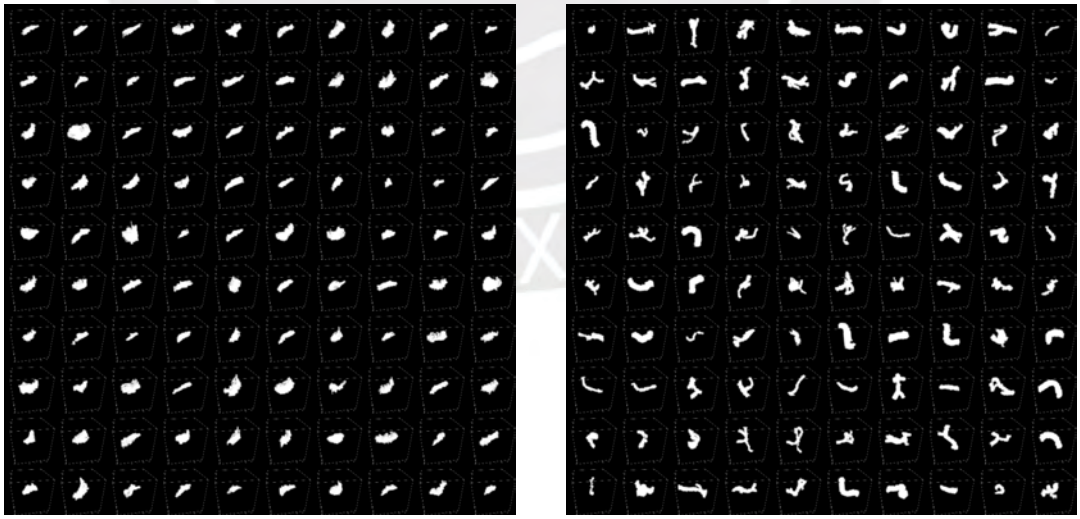
Figure 2: Examples of the generated atoms for the basic shapes dictionary

5.2.2 MIT-BIH Arrhythmia Database

The MIT-BIH Arrhythmia Database [87] contains 48 half-hour fragments of two-channel ECG recordings, obtained from 47 subjects studied by the BIH Arrhythmia Laboratory between 1975 and 1979. The database recordings were sampled at 360 Hz with a resolution of 11 bits over a range of 10 mV. Cardiologists annotated each record; disagreements were resolved to obtain computer-readable reference annotations for each heartbeat. In total, the database includes approximately 110000 annotated beats.

5.2.3 MedMNIST

MedMNIST [88] is a large-scale MNIST-like collection of standardized biomedical data, comprising 12 2D image datasets and 6 3D image datasets. All samples in MedMNIST have been preprocessed into 28×28 matrices and $28 \times 28 \times 28$ tensors for 2D and 3D datasets, respectively, and paired with their corresponding classification labels. MedMNIST encompasses 708069 2D images and 9998 3D images in total, and covers key biomedical imaging modalities and is designed for classification tasks on lightweight 2D and 3D images, accommodating a variety of data scales (ranging from 100 to 100000 images) and task types, including binary and multiclass classification, ordinal regression, and multilabel classification. In this work, the AdrenalMNIST3D and VesselMNIST3D datasets are selected for analysis. These datasets are briefly described in the following, and samples of each dataset are provided in Figure 3, offering visual insight.



(a) AdrenalMNIST3D (b) VesselMNIST3D

Figure 3: Examples of the subsets of the MedMNIST dataset that are used in this work

5.2.3.1 AdrenalMNIST3D

The AdrenalMNIST3D dataset, as described in [88], comprises 3D shape masks of 1,084 adrenal glands (left and right) obtained from 792 patients. Data acquisition was performed at Zhongshan Hospital, affiliated with Fudan University. Abdominal computed tomography (CT) scans were analyzed by an endocrinologist to provide annotations for each 3D shape of an adrenal gland i.e. a binary label indicating whether the gland is normal or has an adrenal mass. For consistency, the center of each adrenal gland was computed, and the center-cropped 64 mm $\times$ 64 mm $\times$ 64 mm cubes were resized into $28 \times 28 \times 28$ voxel tensors. It is important to mention that this dataset is imbalanced and is split into training, validation, and test sets, consisting of 1188, 98, and 298 samples, respectively.

5.2.3.2 VesselMNIST3D

The VesselMNIST3D dataset, as described in [88], is derived from IntrA [89], an open-access 3D intracranial aneurysm dataset. It includes 103 3D models (meshes) of entire brain vessels, reconstructed from MRA (magnetic resonance angiography) images. From these complete models, 1694 healthy vessel segments and 215 aneurysm segments were automatically generated, thus making this dataset imbalanced. To prepare the data for analysis, the non-watertight meshes were corrected, and the watertight meshes were voxelized into $28 \times 28 \times 28$ voxel grids. It is important to mention that this dataset is split into training, validation, and test sets using a ratio of $7 : 1 : 2$.

5.2.4 UCI Air Quality Dataset

The UCI Air Quality Dataset [90] is a dataset from that provides information on air quality measurements from an array of sensors located in a station in Italy recorded from March 2004 to April 2005. This dataset is used to study and model the relationship between various environmental factors and air pollution levels. It includes data collected over several months in the city of Pisa, which is often used to predict air quality parameters or to analyze the impact of certain variables on air pollution levels. The variables collected in this dataset can be categorized into three groups: environmental variables, pollutant concentrations, and sensor responses to target pollutants, with some missing values due to sensor malfunctions. The data can be applied to develop predictive models, perform time-series analysis, and study air pollution patterns.

5.2.5 MEG / EEG data

Electroencephalography (EEG) and magnetoencephalography (MEG) are prominent brain imaging techniques employed to identify and characterize active regions of the brain. A

composite MEG / EEG dataset that was computationally generated following the specifications outlined in [86] was used in this work. The dataset consists of $n = 360$ sensors, with 301 MEG sensors and 59 EEG sensors, used to capture neural activity. The total number of possible sources is $p = 22494$, and the data spans $q = 20$ time instants.

5.2.6 MNIST

The Modified National Institute of Standards and Technology (MNIST) dataset [91] is a cornerstone in machine learning and computer vision research, widely used as a benchmark for evaluating image classification and recognition algorithms. This dataset is derived from the NIST's Special Database 3 (SD-3) and Special Database 1 (SD-1) considered to be the training and test sets of NIST, respectively. This dataset consists of size-centered and normalized 28×28 grayscale images, each representing a handwritten digit, resulting in 10 distinct classes. The MNIST dataset is divided into two sets: (i) a training set containing 60000 images, and (ii) a test set with 10000 images, enabling comprehensive model training and performance evaluation. Table III presents ten sample images from each class, providing a visual overview of this dataset.

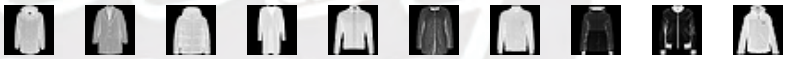
Table III: Examples for each class of the MNIST dataset

Label	Description	Examples
0	Digit 0	
1	Digit 1	
2	Digit 2	
3	Digit 3	
4	Digit 4	
5	Digit 5	
6	Digit 6	
7	Digit 7	
8	Digit 8	
9	Digit 9	

5.2.7 Fashion-MNIST

Fashion-MNIST [92] is a dataset used for machine learning and computer vision tasks. It provides a more challenging and realistic dataset for training and testing machine learning models than MNIST. This dataset consists of a collection of 70000 grayscale images of clothing items, each associated with a corresponding label indicating the type of clothing. It contains 10 different categories of clothing and accessories, specifically items such as t-shirts, trousers, pullovers, dresses, coats, sandals, shirts, sneakers, bags, and ankle boots. The examples of the Fashion-MNIST dataset consist of 28×28 size-centered grayscale images, making this dataset similar in structure to the original MNIST dataset Ten examples for each class in the Fashion-MNIST dataset figure can be observed in Table IV.

Table IV: Examples for each class of Fashion-MNIST dataset

Label	Description	Examples
0	T-shirt	
1	Trouser	
2	Pullover	
3	Dress	
4	Coat	
5	Sandals	
6	Shirt	
7	Sneaker	
8	Bag	
9	Ankle boots	

5.3 Experimental framework

All experiments were conducted using datasets that were deemed appropriate for the specific tasks. It is important to note that the proposed method was developed based on the gap safe screening framework [86]. The performance of the proposed method was com-

pared against a vanilla implementation i.e. without screening, as well as implementations incorporating either gap safe or strong screening rules in each experiment. The following tests were employed to validate the proposed method:

1. **Basis pursuit denoising:** This task refers to the problem of reconstructing a signal by finding the sparsest solution while allowing for noise. In the first case, a dictionary of basic shapes is used to form a composite image as a linear combination of a few of its atoms. The observed data is this composite image, contaminated with additive Gaussian noise ($\mu = 0$, $\sigma^2 = 2.56 \times 10^{-2}$), and the objective is to recover the sparse representation associated with the image from the noisy observations. In the second case, the task involves using the MIT-BIH Arrhythmia Database, where ECG signals are represented using a dictionary built upon the inverse type-II discrete cosine transform (DCT), an approach that has been exploited for ECG compression [93]. A 20-second segment of the right-channel ECG signal is contaminated with additive Gaussian noise ($\mu = 0$, $\sigma^2 = 2.56 \times 10^{-2}$), and the objective is to eliminate the noise from the contaminated ECG signal.

2. **Binary classification:** This task is approached using binary logistic regression, a statistical method that models the relationship between a binary dependent variable and one or more independent variables. The AdrenalMNIST3D and VesselMNIST3D datasets are utilized for this experiment. For each dataset, the training, validation, and test subsets were merged and subsequently re-split into new training and test sets using a randomized 85% / 15% split.

3. **Joint-sparse reconstruction:** Joint-sparse signals are those that share the same support in a specific domain. In this experiment, the variables temperature, relative humidity, and absolute humidity are extracted from the UCI Air Quality Dataset and treated as jointly sparse due to their cyclical, weather-driven patterns. To achieve an efficient representation under additive Gaussian noise ($\mu = 0$, $\sigma^2 = 2.56 \times 10^{-2}$), a dictionary built using the inverse type-II DCT is employed within a multitask LASSO regression framework. This approach effectively captures both smooth baseline trends and sharp transient features.

4. **MEG / EEG source imaging:** Typically, this problem consists of solving a multitask regression problem with squared loss where every task corresponds to a time instant using the MEG / EEG data. It is valid to impose a temporal stationary assumption i.e. the recovered sources are identical during a short time interval, then, this task can be modeled as a multitask LASSO [94].

5. **Multiclass classification:** This task is modeled as sparse multinomial logistic regression, an extension of logistic regression using the softmax function to predict category probabilities. The model assigns each observation to the category with the highest probability based on input features. The MNIST and Fashion-MNIST datasets are used for this task.

Each experiment consists of performing the task for a sequence of hyperparameter values $(10^k \cdot \lambda_{\max},\ \ k = -2, -1.9, \ldots, -0.1)$, 100 times for each hyperparameter value in the sequence in order to reduce variability. The selection of this hyperparameter range is based on the observation that all tasks yield meaningful results without using screening within this range. For each experiment (for all values in the sequence of hyperparameters), the set of values $\alpha_k,\ \ k = 0,1,2,3,4$ used by the proposed screening method is selected by grid search. The resulting values for α_k used in each experiment are indicated in Table V.

Table V: Values of $\alpha_k,\ \ k = 0,1,2,3,4$ for the validation experiments

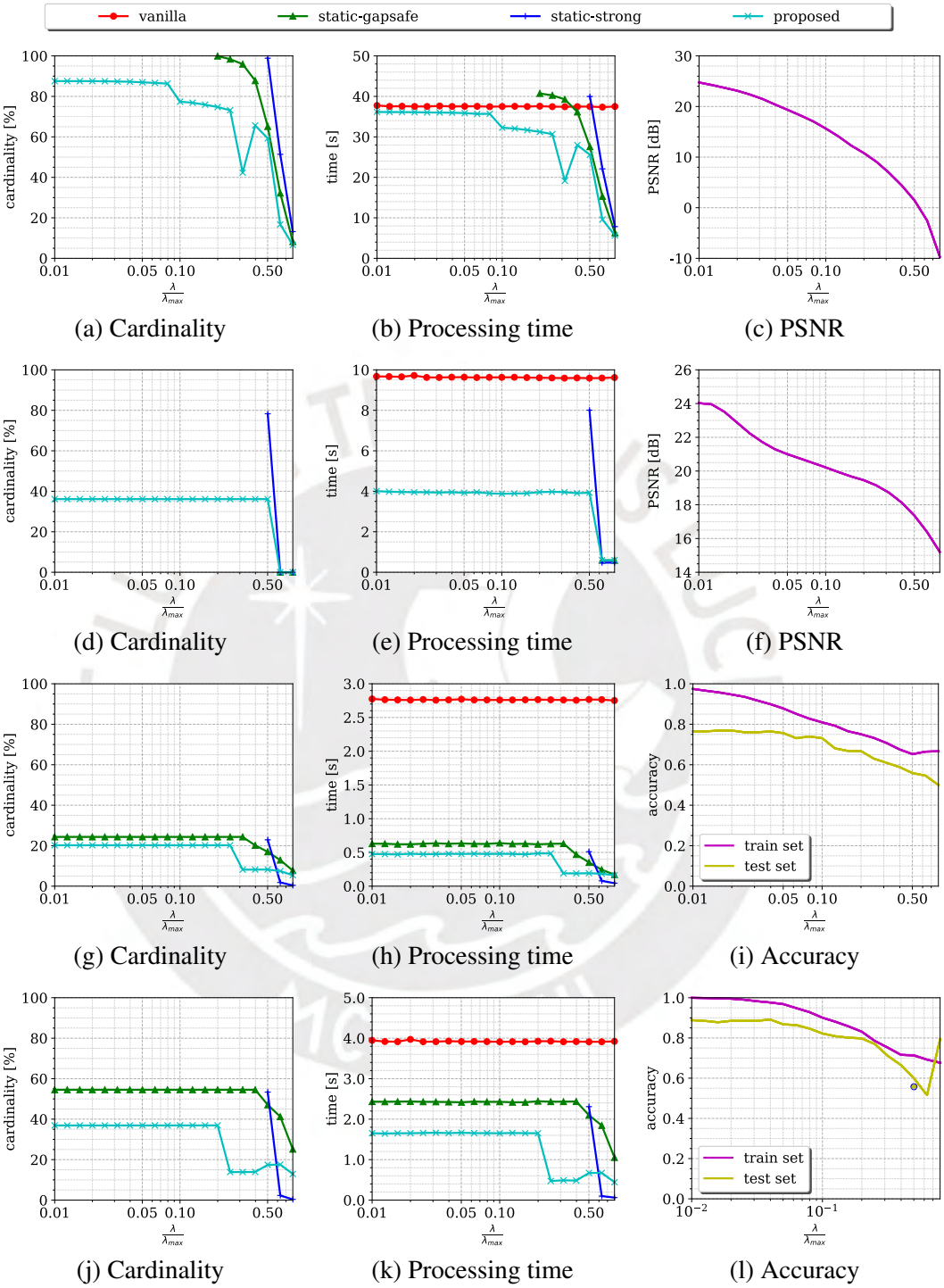
Experiment	α_0	α_1	α_2	α_3	α_4
BPDN (Basic shapes dictionary)	0.9	0.9	0.5	0.5	0.5
BPDN (MIT-BIH Arrythmia Database)	0.8	0.8	0.8	0.8	0.8
Image classification (AdrenalMNIST3D)	0.9	0.7	0.7	0.7	0.7
Image classification (VesselMNIST3D)	0.9	0.2	0.2	0.2	0.2
Joint sparse reconstruction (UCI Air Quality Dataset)	0.85	0.85	0.85	0.85	0.85
MEG / EEG source imaging	0.75	0.25	0.5	0.5	0.5
Image classifcation (MNIST)	0.9	0.2	0.5	0.5	0.5
Image classifcation (Fashion-MNIST)	0.85	0.85	0.85	0.85	0.85

It is important to mention that all experiments employed the APG optimization algorithm with a fixed step size. This choice was deliberate, as adaptive step size techniques rely on the gradient of iterative solution values. Since the tested screening techniques discard entries prior to the optimization procedure, they disrupt the gradient value, thereby affecting convergence properties and complicating the analysis if adaptive step size techniques were used. It is also important to mention that the imbalanced datasets were handled by applying the Synthetic Minority Over-sampling Technique (SMOTE) [95], using the implementation provided in the widely adopted imbalanced-learn library [96].

5.4 Numerical results

All experiments were carried out on a desktop computer equipped with a 12th Gen Intel$^{(R)}$ Core$^{(TM)}$ i7-12700K CPU (4.7 GHz, 25 MB cache, 64 GB RAM) using Python 3. It is important to mention that the figures only display results where there is actual discarding i.e. cardinality lower than 100% due to screening, that is, the results corresponding to tests with hyperparameter values where no discarding was done are not displayed. The methods included in the experiment are gap safe screening, strong screening, and the proposed method which are labeled as static-gapsafe, static-strong, and proposed, respectively.

Cardinality, processing time (in seconds), and a quality metric (depending on the task) for the basis pursuit denoising and binary classification experiments are presented in Figure 4. Similarly, the same evaluation metrics for the joint sparse reconstruction of environmental variables, MEG / EEG source imaging, and multiclass classification experiments are shown in Figure 5. Finally, the speedups achieved by all screening-enabled optimizers compared to their vanilla counterparts in all the models evaluated in the experiments are summarized in Tables VI, VII, VIII, and IX.



(a) Cardinality (b) Processing time (c) PSNR

(d) Cardinality (e) Processing time (f) PSNR

(g) Cardinality (h) Processing time (i) Accuracy

(j) Cardinality (k) Processing time (l) Accuracy

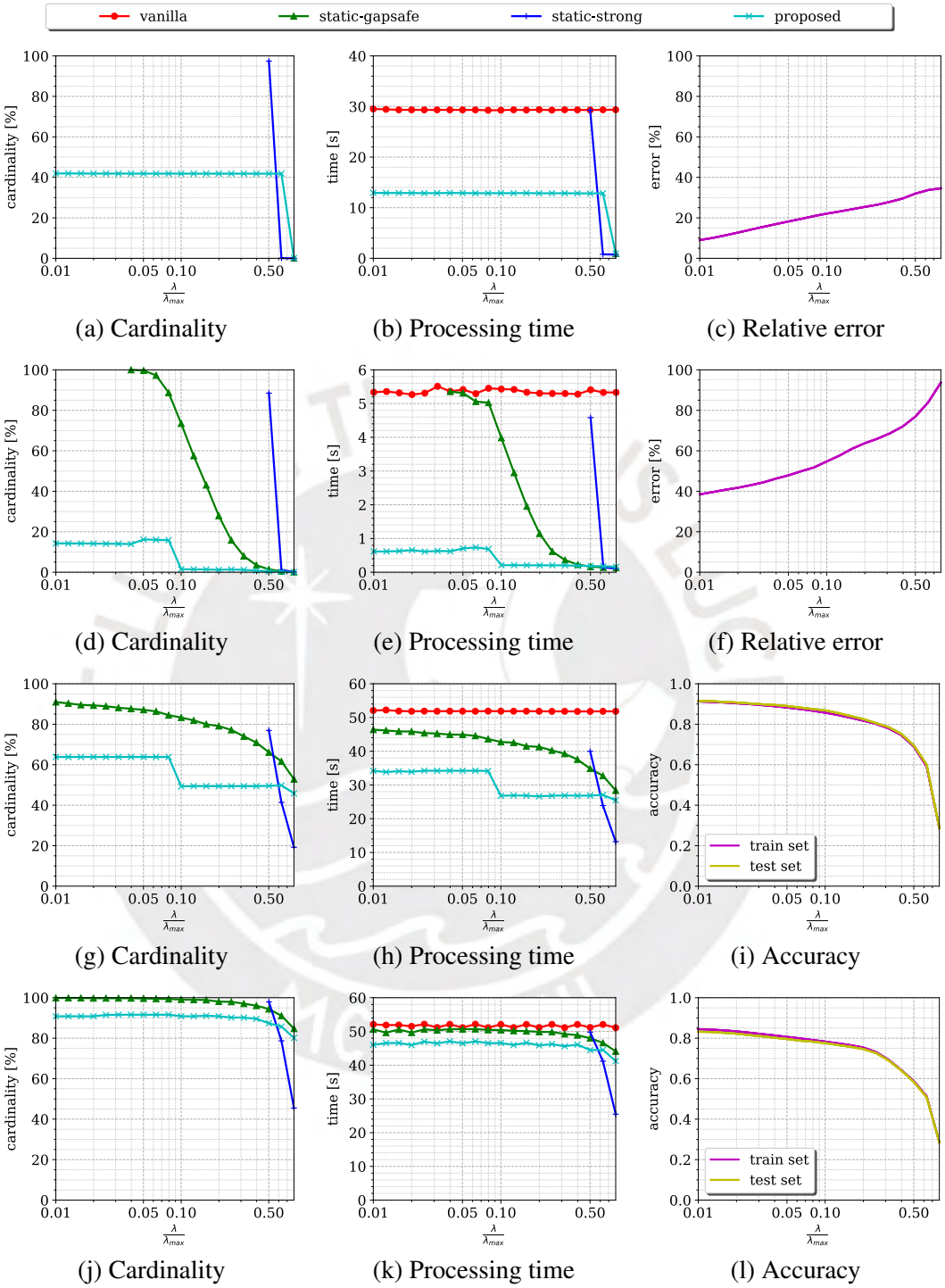
Figure 4: Evaluation metrics for the experiments using element-wise sparse regularization. From top to bottom: Basis pursuit denoising using the basic shapes dictionary (first row) and for MIT-BIH Arrythmia Dataset (second row), binary classification for AdrenalMNIST3D (third row) and VesselMNIST3D (fourth row).

Table VI: Speedups for the BPDN experiments (both with basic shapes dictionary and with the MIT-BIH Arrythmia Dataset). Entries with '−' correspond to data points in the experiment where the respective screening methodology does not discard features, thus they were not considered for calculating speedup. Best results are highlighted in bold.

$\log_{10}\left(\frac{\lambda}{\lambda_{\max}}\right)$	**BPDN (Basic shapes dictionary)**			**BPDN (MIT-BIH Arrythmia Dataset)**		
	Static gap safe	**Static strong**	**Proposed**	**Static gap safe**	**Static strong**	**Proposed**
-2	–	–	**1.04**	–	–	**2.42**
-1.9	–	–	**1.04**	–	–	**2.43**
-1.8	–	–	**1.04**	–	–	**2.44**
-1.7	–	–	**1.04**	–	–	**2.46**
-1.6	–	–	**1.04**	–	–	**2.43**
-1.5	–	–	**1.05**	–	–	**2.44**
-1.4	–	–	**1.04**	–	–	**2.44**
-1.3	–	–	**1.05**	–	–	**2.46**
-1.2	–	–	**1.05**	–	–	**2.43**
-1.1	–	–	**1.05**	–	–	**2.47**
-1	–	–	**1.16**	–	–	**2.48**
-0.9	–	–	**1.17**	–	–	**2.48**
-0.8	–	–	**1.18**	–	–	**2.47**
-0.7	0.92	–	**1.2**	–	–	**2.43**
-0.6	0.93	–	**1.22**	–	–	**2.42**
-0.5	0.95	–	**1.95**	–	–	**2.42**
-0.4	1.04	–	**1.34**	–	–	**2.46**
-0.3	1.36	0.94	**1.47**	–	1.2	**2.44**
-0.2	2.44	1.69	**3.9**	16.26	**20.26**	16.06
-0.1	5.97	4.78	**6.61**	16.32	**20.26**	16.15

Table VII: Speedups for the image classification experiments (both with AdrenalMNIST3D and with VesselMNIST3D). Entries with '−' correspond to data points in the experiment where the respective screening methodology does not discard features, thus they were not considered for calculating speedup. Best results are highlighted in bold.

$\log_{10}\left(\frac{\lambda}{\lambda_{\max}}\right)$	**Binary classification (AdrenalMIST3D)**			**Binary classification (VesselMNIST3D)**		
	Static gap safe	**Static strong**	**Proposed**	**Static gap safe**	**Static strong**	**Proposed**
-2	4.41	–	**5.83**	1.62	–	**2.39**
-1.9	4.39	–	**5.82**	1.61	–	**2.38**
-1.8	4.44	–	**5.88**	1.61	–	**2.37**
-1.7	4.44	–	**5.74**	1.63	–	**2.4**
-1.6	4.4	–	**5.84**	1.61	–	**2.36**
-1.5	4.34	–	**5.8**	1.61	–	**2.35**
-1.4	4.4	–	**5.76**	1.62	–	**2.37**
-1.3	4.37	–	**5.81**	1.62	–	**2.35**
-1.2	4.4	–	**5.74**	1.61	–	**2.37**
-1.1	4.41	–	**5.79**	1.61	–	**2.37**
-1	4.32	–	**5.75**	1.61	–	**2.37**
-0.9	4.41	–	**5.78**	1.62	–	**2.36**
-0.8	4.39	–	**5.86**	1.62	–	**2.37**
-0.7	4.45	–	**5.68**	1.61	–	**2.38**
-0.6	4.4	–	**5.7**	1.61	–	**8.31**
-0.5	4.37	–	**14.47**	1.61	–	**8**
-0.4	5.87	–	**14.6**	1.61	–	**8.1**
-0.3	7.86	5.41	**14.43**	1.86	1.69	**5.82**
-0.2	11.34	**35.02**	15.03	2.12	**39.49**	5.82
-0.1	16.12	**61.98**	16.85	3.73	**62.99**	8.88



(a) Cardinality (b) Processing time (c) Relative error

(d) Cardinality (e) Processing time (f) Relative error

(g) Cardinality (h) Processing time (i) Accuracy

(j) Cardinality (k) Processing time (l) Accuracy

Figure 5: Evaluation metrics for the experiments using group sparse regularization. From top to bottom: Joint sparse reconstruction of environmental variables (first row), MEG / EEG source imaging (second row), multiclass classification for MNIST (third row) and Fashion-MNIST (fourth row).

Table VIII: Speedups for the joint sparse reconstruction (for environmental variables) and MEG / EEG source imaging. Entries with '−' correspond to data points in the experiment where the respective screening methodology does not discard features, thus they were not considered for calculating speedup. Best results are highlighted in bold.

$\log_{10}\left(\frac{\lambda}{\lambda_{\max}}\right)$	**Joint sparse reconstruction (UCI Air Quality Dataset)**			**MEG / EEG source imaging**		
	Static gap safe	**Static strong**	**Proposed**	**Static gap safe**	**Static strong**	**Proposed**
-2	–	–	**2.28**	–	–	**8.68**
-1.9	–	–	**2.28**	–	–	**8.68**
-1.8	–	–	**2.27**	–	–	**8.46**
-1.7	–	–	**2.27**	–	–	**8**
-1.6	–	–	**2.28**	–	–	**8.68**
-1.5	–	–	**2.28**	–	–	**8.71**
-1.4	–	–	**2.27**	1	–	**8.65**
-1.3	–	–	**2.27**	1.02	–	**7.68**
-1.2	–	–	**2.28**	1.05	–	**7.2**
-1.1	–	–	**2.27**	1.09	–	**7.93**
-1	–	–	**2.27**	1.36	–	**26.01**
-0.9	–	–	**2.28**	1.84	–	**26.09**
-0.8	–	–	**2.27**	2.73	–	**25.86**
-0.7	–	–	**2.29**	4.63	–	**25.95**
-0.6	–	–	**2.28**	8.63	–	**25.72**
-0.5	–	–	**2.28**	14.67	–	**26.18**
-0.4	–	–	**2.28**	23.31	–	**27.51**
-0.3	–	1	**2.29**	**32.61**	1.18	29.66
-0.2	–	**37.55**	2.28	37.42	**36.9**	29.94
-0.1	30.91	**39.05**	31	40.73	**47.89**	33.9

Table IX: Speedups for the image classification experiments (both with MNIST and Fashion-MNIST datasets). Entries with '–' correspond to data points in the experiment where the respective screening methodology does not discard features, thus they were not considered for calculating speedup. Best results are highlighted in bold.

$\log_{10}\left(\frac{\lambda}{\lambda_{\max}}\right)$	**Multiclass classification (MNIST)**			**Multiclass classification (Fashion-MNIST)**		
	Static gap safe	**Static strong**	**Proposed**	**Static gap safe**	**Static strong**	**Proposed**
-2	1.12	–	**1.52**	1.03	–	**1.13**
-1.9	1.13	–	**1.54**	1.05	–	**1.11**
-1.8	1.13	–	**1.52**	1.03	–	**1.11**
-1.7	1.13	–	**1.53**	1.04	–	**1.12**
-1.6	1.14	–	**1.52**	1.03	–	**1.11**
-1.5	1.15	–	**1.52**	1.02	–	**1.1**
-1.4	1.15	–	**1.52**	1.03	–	**1.11**
-1.3	1.15	–	**1.52**	1.01	–	**1.1**
-1.2	1.16	–	**1.51**	1.03	–	**1.11**
-1.1	1.19	–	**1.52**	1.01	–	**1.1**
-1	1.21	–	**1.94**	1.03	–	**1.12**
-0.9	1.22	–	**1.93**	1.02	–	**1.11**
-0.8	1.25	–	**1.93**	1.04	–	**1.12**
-0.7	1.26	–	**1.95**	1.03	–	**1.12**
-0.6	1.29	–	**1.93**	1.04	–	**1.13**
-0.5	1.32	–	**1.93**	1.04	–	**1.12**
-0.4	1.38	–	**1.93**	1.07	–	**1.13**
-0.3	1.49	1.3	**1.93**	1.07	1.03	**1.15**
-0.2	1.58	**2.17**	1.92	1.12	**1.26**	1.17
-0.1	1.83	**3.93**	2.03	1.16	**2**	1.24

5.5 Discussion

Computational results demonstrate that the proposed screening methodology effectively discards a sufficient number of features to achieve significant time savings, while maintaining the same level of quality as the baseline solution i.e. the proposed method maintains negligible distortion, as can be observed in the plots of quality metrics (PSNR, error, and accuracy) in Figures 4 and 5 where all plots practically overlap. Numerically, there was no loss in quality metrics for all experiments with the exception of BPDN with geometric shapes dictionary and MEG / EEG source imaging, where the relative error in quality metrics remained at most 0.22% and 0.05%, respectively. In contrast, strong screening leads to a noticeably different level of test accuracy in the binary classification task on the VesselMNIST3D dataset as can be observed in Figure 4(l). This is expected, as the strong screening rules may discard features that are important for the solution. Notably, the proposed method screens out more features than both the gap safe sphere and strong screening methods in several cases, except at high values of the hyperparameter (i.e., values close to $\lambda_{\max}$). This explains the shorter processing times when the proposed method is applied, compared to the processing times associated with other screening techniques.

Furthermore, for low values of the regularization hyperparameter λ, gap safe screening and strong screening either discard very few features or fail to discard any features at all. In contrast, the proposed adaptive thresholding technique enables feature discarding in these cases, yielding computational savings with low or zero quality degradation quality in the solution. This is supported by quality evaluation metrics (refer to the metrics in each figure) and by observing the speedup (see Tables VI, VII, VIII and IX).

It is also worth mentioning that the cardinality behavior of the solutions generated by the proposed screening technique is irregular. Specifically, more features can sometimes be discarded at lower hyperparameter values. This behavior is expected because the thresholding mechanism employed in the proposed screening method depends on the histogram shape rather than directly on the hyperparameter value.

Conclusions

This thesis introduces a novel method for enhancing safe screening rules by interpreting the screening procedure as a thresholding operation on a measure associated with the relevance of each feature in the optimization problem, which can be improved through adaptive thresholding. The proposed method introduces a thresholding rule to be used over safe screening rules to discard more features, thus enabling greater computational savings. Its efficiency was assessed through experiments on both synthetic and real datasets, covering tasks such as basis pursuit denoising, joint-sparse signal reconstruction, MEG / EEG source imaging, and image classification.

Experimental results demonstrate that the proposed method effectively accelerates optimization solvers for optimization problems with sparsity-promoting regularization with minimal impact on quality metrics, provided the method's parameters are appropriately tuned. It is important to mention that the proposed screening method, as a static preprocessing step, facilitates screening for low values of the sparsity-controlling hyperparameter, a capability not observed in other state-of-the-art techniques, hence leveraging sparsity more efficiently. Furthermore, while the proposed method introduces more computational overhead compared to traditional safe screening, it outperforms state-of-the-art screening methods, making it a promising option for accelerating sparse models widely used in signal processing and machine learning.

References

[1] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, 1st ed. Springer New York, 2010.

[2] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with Sparsity-Inducing Penalties," *Foundations and Trends® in Machine Learning*, vol. 4, no. 1, pp. 1–106, 2012.

[3] L. El Ghaoui, Vivian Viallon, and Tarek Rabbani, "Safe Feature Elimination in Sparse Supervised Learning," EECS Dept., University of California at Berkeley, Tech. Rep. UC/EECS-2010-126, 2010.

[4] A. Raj, J. Olbrich, B. Gärtner, B. Schölkopf, and M. Jaggi, "Screening Rules for Convex Problems," *arXiv preprint arXiv:1609.07478*, 2016.

[5] E. Ndiaye, "Safe optimization algorithms for variable selection and hyperparameter tuning," Ph.D. dissertation, Université Paris-Saclay, 2018.

[6] H. Chahuara and P. Rodriguez, "Enhancing safe screening rules with adaptive thresholding for non-overlapping group sparse norm regularized problems," in *2023 24th International Conference on Digital Signal Processing (DSP)*, 2023, pp. 1–5.

[7] J. Sevilla, L. Heim, A. Ho, T. Besiroglu, M. Hobbhahn, and P. Villalobos, "Compute Trends Across Three Eras of Machine Learning," in *2022 International Joint Conference on Neural Networks (IJCNN)*, 2022, pp. 1–8.

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. The MIT Press, 2001.

[9] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[10] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics Letters*, vol. 44, pp. 800–801(1), 2008.

[11] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[12] A. Beck, *First-Order Methods in Optimization*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017.

[13] R. T. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.

[14] J. Nocedal and S. Wright, *Numerical optimization*, 2nd ed., ser. Springer series in operations research and financial engineering. New York, NY: Springer, 2006.

[15] D. P. Bertsekas, *Nonlinear Programming*, 3rd ed. Belmont, MA: Athena Scientific, 2016.

[16] A.-L. Cauchy, "Méthode générale pour la résolution des systèmes d'équations simultanées," *Comptes Rendus de l'Academie des Science*, vol. 25, pp. 536–538, 1847.

[17] Y. E. Nesterov, "A method for solving the convex programming problem with convergence rate $O(1/k^2)$," *Dokl. Akad. Nauk SSSR*, vol. 269, pp. 543–547, 1983.

[18] N. Parikh and S. Boyd, "Proximal Algorithms," *Foundations and Trends® in Machine Learning*, vol. 1, no. 3, pp. 127–239, 2014.

[19] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[20] M. A. T. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906–916, 2003.

[21] P. L. Combettes and V. R. Wajs, "Signal Recovery by Proximal Forward-Backward Splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[22] A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," *SIAM J. Imaging Sciences*, vol. 2, pp. 183–202, 2009.

[23] M. Raydan and B. Svaiter, "Relaxed Steepest Descent and Cauchy-Barzilai-Borwein Method," *Computational Optimization and Applications*, vol. 21, pp. 155–167, 2002.

[24] T. Blumensath and M. E. Davies, "Normalized Iterative Hard Thresholding: Guaranteed Stability and Performance," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 298–309, 2010.

[25] J. Barzilai and J. Borwein, "Two-Point Step Size Gradient Methods," *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[26] T. Li and Z. Wan, "New adaptive Barzilai-Borwein step size and its application in solving large-scale optimization problems," *The ANZIAM Journal*, vol. 61, no. 1, p. 76–98, 2019.

[27] Y.-H. Dai, "Alternate step gradient method," *Optimization*, vol. 52, no. 4-5, pp. 395–415, 2003.

[28] A. L. Brearley, G. Mitra, and H. P. Williams, "Analysis of mathematical programming problems prior to applying the simplex algorithm," *Mathematical Programming*, vol. 8, pp. 54–83, 1975.

[29] C. Mészáros and U. Suhl, "Advanced preprocessing techniques for linear and quadratic programming," *OR Spectrum*, vol. 25, pp. 575–595, 2003.

[30] J. M. Borwein and H. Wolkowicz, "Facial reduction for a cone-convex programming problem," *Journal of the Australian Mathematical Society. Series A. Pure Mathematics and Statistics*, vol. 30, no. 3, p. 369–380, 1981.

[31] D. Drusvyatskiy and H. Wolkowicz, "The Many Faces of Degeneracy in Conic Optimization," *Foundations and Trends® in Optimization*, vol. 3, no. 2, p. 77–170, 2017.

[32] N. Dinh, G. Vallet, and M. Volle, "Functional inequalities and theorems of the alternative involving composite functions," *Journal of Global Optimization*, vol. 59, no. 4, p. 837–863, 2014.

[33] F. Permenter and P. Parrilo, "Partial facial reduction: Simplified, equivalent sdps via approximations of the psd cone," *arXiv preprint arXiv:1408.4685*, 2014.

[34] H. Hu, R. Sotirov, and H. Wolkowicz, "Facial reduction for symmetry reduced semidefinite and doubly nonnegative programs," *Math. Program.*, vol. 200, no. 1, p. 475–529, 2022.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Foundations and Trends in Machine Learning*, vol. 3, pp. 1–122, 2011.

[36] S. Huang and H. Wolkowicz, "Low-rank matrix completion using nuclear norm minimization and facial reduction," *Journal of Global Optimization*, vol. 72, pp. 5 – 26, 2017.

[37] S. Ma, F. Wang, L. Wei, and H. Wolkowicz, "Robust principal component analysis using facial reduction," *Optimization and Engineering*, vol. 21, pp. 1195 – 1219, 2019.

[38] H. Waki and M. Muramatsu, "A facial reduction algorithm for finding sparse SOS representations," *Operations Research Letters*, vol. 38, no. 5, pp. 361–365, 2010.

[39] J. Fan and J. Lv, "Sure independence screening for ultrahigh dimensional feature space," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 5, pp. 849–911, 2008.

[40] W. Zhong, "Robust sure independence screening for ultrahigh dimensional non-normal data," *Acta Mathematica Sinica-English Series*, vol. 30, pp. 1885–1896, 2014.

[41] L. Z. Xue and H. Zou, "Sure independence screening and compressed random sensing," *Biometrika*, vol. 98, no. 2, pp. 371–380, 2011.

[42] E. Barut, J. Fan, and A. Verhasselt, "Conditional sure independence screening," *Journal of the American Statistical Association*, vol. 111, no. 515, pp. 1266–1277, 2016, publisher Copyright: © 2016 American Statistical Association.

[43] J. Liu, Z. Zhao, J. Wang, and J. Ye, "Safe Screening with Variational Inequalities and Its Application to Lasso," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32. Beijing, China: PMLR, 2014, pp. 289–297.

[44] Z. J. Xiang, Y. Wang, and P. J. Ramadge, "Screening tests for lasso problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 5, pp. 1008–1027, 2017.

[45] S. Ren, S. Huang, J. Ye, and X. Qian, "Safe Feature Screening for Generalized LASSO," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2992–3006, 2018.

[46] X. Xiao, Y. Xu, Y. Zhang, and P. Zhong, "A novel self-weighted Lasso and its safe screening rule," *Applied Intelligence*, vol. 52, no. 12, p. 14465–14477, 2022.

[47] J. Wang, J. Zhou, J. Liu, P. Wonka, and J. Ye, "A safe screening rule for sparse logistic regression," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 1053–1061.

[48] X. Pan and Y. Xu, "A Safe Feature Elimination Rule for L_1–Regularized Logistic Regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 4544–4554, 2022.

[49] A. Deza and A. Atamturk, "Safe screening for logistic regression with ℓ_0-ℓ_2 regularization," *arXiv preprint arXiv:2202.00467*, 2022.

[50] X. Pang and Y. Xu, "A safe screening rule for accelerating weighted twin support vector machine," *Soft Computing*, vol. 23, pp. 7725–7739, 2019.

[51] H. Wang, J. Zhu, and F. Feng, "Elastic net twin support vector machine and its safe screening rules," *Information Sciences*, vol. 635, pp. 99–125, 2023.

[52] H. Wang, J. Zhu, and S. Zhang, "Safe screening rules for multi-view support vector machines," *Neural Networks*, vol. 166, pp. 326–343, 2023.

[53] Z. Xiang, H. Xu, and P. J. Ramadge, "Learning sparse representations of high dimensional data on large scale dictionaries," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Eds., vol. 24. Curran Associates, Inc., 2011.

[54] L. Dai and K. Pelckmans, "An ellipsoid based, two-stage screening test for BPDN," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 654–658.

[55] Z. J. Xiang and P. J. Ramadge, "Fast lasso screening tests based on correlations," *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2137–2140, 2012.

[56] T.-L. Tran, C. Elvira, H.-P. Dang, and C. Herzet, "Beyond GAP screening for Lasso by exploiting new dual cutting half-spaces," in *2022 30th European Signal Processing Conference (EUSIPCO)*, 2022, pp. 2056–2060.

[57] J. Wang, J. Zhou, P. Wonka, and J. Ye, "Lasso Screening Rules via Dual Polytope Projection," in *Advances in Neural Information Processing Systems*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., vol. 26. Curran Associates, Inc., 2013.

[58] R. Tibshirani, J. Bien, J. Friedman, T. Hastie, N. Simon, J. Taylor, and R. J. Tibshirani, "Strong rules for discarding predictors in lasso-type problems," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 74, no. 2, pp. 245–266, 2012.

[59] E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon, "Gap Safe Screening Rules for Sparsity Enforcing Penalties," *J. Mach. Learn. Res.*, vol. 18, no. 1, p. 4671–4703, 2017.

[60] A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval, "A dynamic screening principle for the Lasso," in *2014 22nd European Signal Processing Conference (EUSIPCO)*, 2014, pp. 6–10.

[61] ——, "Dynamic Screening: Accelerating First-Order Algorithms for the Lasso and Group-Lasso," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5121–5132, 2015.

[62] O. Fercoq, A. Gramfort, and J. Salmon, "Mind the duality gap: safer rules for the Lasso," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 2015, pp. 333–342.

[63] H. Yamada and M. Yamada, "Dynamic Sasvi: Strong Safe Screening for Norm-Regularized Least Squares," in *Advances in Neural Information Processing Systems*,

M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 14 645–14 655.

[64] A. Rakotomamonjy, G. Gasso, and J. Salmon, "Screening Rules for Lasso with Non-Convex Sparse Regularizers," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and M. Sugiyama, Eds., vol. 97. Long Beach, California, USA: PMLR, 2019, pp. 5341–5350.

[65] A. Atamturk and A. Gomez, "Safe screening rules for L0-regression from Perspective Relaxations," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. Dhua and A. Singh, Eds., vol. 119. Virtual: PMLR, 2020, pp. 421–430.

[66] T. Guyard, C. Herzet, and C. Elvira, "Node-Screening Tests for the ℓ_0-Penalized Least-Squares Problem," in *ICASSP 2022 - IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 5448–5452.

[67] S. Lee and E. Xing, "Screening Rules for Overlapping Group Lasso," *arXiv preprint arXiv:1410.6880*, 2014.

[68] J. Larsson, M. Bogdan, and J. Wallin, "The Strong Screening Rule for SLOPE," 2020.

[69] C. Elvira and C. Herzet, "Safe Rules for the Identification of Zeros in the Solutions of the SLOPE Problem," *SIAM Journal on Mathematics of Data Science*, vol. 5, no. 1, pp. 147–173, 2023.

[70] C. Dantas, E. Soubies, and C. Févotte, "Expanding boundaries of gap safe screening," *Journal of Machine Learning Research*, vol. 22, no. 236, pp. 1–57, 2021.

[71] S. Lee, N. Görnitz, E. P. Xing, D. Heckerman, and C. Lippert, "Ensembles of lasso screening rules," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2841–2852, 2018.

[72] Y. Zeng, T. Yang, and P. Breheny, "Hybrid safe–strong rules for efficient optimization in lasso-type problems," *Computational Statistics & Data Analysis*, vol. 153, p. 107063, 2021.

[73] T. Guyard, C. Herzet, and C. Elvira, "Screen & Relax: Accelerating The Resolution Of Elastic-Net By Safe Identification of The Solution Support," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 5443–5447.

[74] C. Herzet, C. Elvira, and H.-P. Dang, "Region-free Safe Screening Tests for ℓ_1-penalized Convex Problems," in *2022 30th European Signal Processing Conference (EUSIPCO)*, Belgrade, Serbia, 2022.

[75] A. Calderbank and I. Daubechies, "The pros and cons of democracy," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1721–1725, 2002.

[76] Z. Cvetkovic, "Resilience properties of redundant expansions under additive noise and quantization," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 644–656, 2003.

[77] B. Farrell and P. Jung, "A Kashin Approach to the Capacity of the Discrete Amplitude Constrained Gaussian Channel," in *International Conference on Sampling Theory and Applications (SAMPTA)*, Marseille, France, 2009.

[78] C. Studer and E. G. Larsson, "PAR-Aware Large-Scale Multi-User MIMO-OFDM Downlink," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 303–313, 2013.

[79] C. Studer, T. Goldstein, W. Yin, and R. G. Baraniuk, "Democratic representations," *arXiv preprint arXiv:1401.3420*, 2014.

[80] C. Elvira and C. Herzet, "Short and Squeezed: Accelerating the Computation of Antisparse Representations with Safe Squeezing," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 5615–5619.

[81] ——, "Safe Squeezing for Antisparse Coding," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3252–3265, 2020.

[82] P. L. Rosin, "Unimodal thresholding," *Pattern Recognition*, vol. 34, pp. 2083–2096, 2001.

[83] D. P. Doane and L. E. Seward, "Measuring skewness: A forgotten statistic?" *Journal of Statistics Education*, vol. 19, no. 2, 2011.

[84] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[85] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[86] E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon, "GAP safe screening rules for sparse multi-task and multi-class models," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., 2015, pp. 811–819.

[87] G. Moody and R. Mark, "The impact of the MIT-BIH Arrhythmia Database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[88] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "MedMNIST v2-A large-scale lightweight benchmark for 2D and 3D biomedical image classification," *Scientific Data*, vol. 10, no. 1, p. 41, 2023.

[89] X. Yang, D. Xia, T. Kin, and T. Igarashi, "IntrA: 3D Intracranial Aneurysm Dataset for Deep Learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2653–2663.

[90] S. De Vito, "Air Quality Data Set," https://archive.ics.uci.edu/ml/datasets/Air+Quality, 2008, UCI Machine Learning Repository.

[91] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[92] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," 2017.

[93] V. Allen and J. Belina, "ECG data compression using the discrete cosine transform (DCT)," in *Proceedings Computers in Cardiology*, 1992, pp. 687–690.

[94] A. Gramfort, M. Kowalski, and M. Hämäläinen, "Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods," *Physics in Medicine & Biology*, vol. 57, no. 7, p. 1937, 2012.

[95] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *J. Artif. Int. Res.*, vol. 16, no. 1, p. 321–357, 2002.

[96] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning," *Journal of Machine Learning Research*, vol. 18, no. 17, pp. 1–5, 2017.