UNIVERSITÀ DEGLI STUDI DI PALERMO

Dottorato in Scienze Economiche e Statistiche
Dipartimento di Scienze Economiche, Aziendali e Statistiche
Settore Scientifico Disciplinare: SECS-S/01

Variable Selection in High Dimensions: the Adaptive Non-Convex Penalty Function

IL DOTTORE
Daniele Cuntrera

IL COORDINATORE
Prof. Vito M. R. Muggeo

IL TUTOR
Prof. Vito M. R. Muggeo

IL CO TUTOR
Prof. Luigi Augugliaro

CICLO XXXVI
ANNO CONSEGUIMENTO TITOLO 2024

Acknowledgements

If I were to depict the mood swings throughout these three years, it would undoubtedly be a series marked by high volatility.

The emotions tied to the experiences I've had were diverse in both type and intensity. But in drawing conclusions, I believe it has been the most intense and beautiful experience so far for what it has left me with. Although the journey was marked by moments of emotional and physical distress, I cannot help but define the overall balance in positive terms.

For this reason, first and foremost, I must thank my two supervisors, who guided me through these three years. I am grateful for their understanding and availability, especially at the beginning of the second year, which was undoubtedly the hardest time of my life.

I thank my girlfriend and my family, who have been close to me and have given me the strength to overcome certain moments, also being an active (albeit indirect) part of this work.

I thank my friends, with whom I could discuss and recount experiences, to share joys and overcome difficulties.

I thank the professors with whom I have shared various moments and works, all of whom I hold precious memories.

I thank my PhD colleagues: those who have been with me since the beginning of this journey that started from my bachelor's degree, with whom I shared some of the most fun moments of my life. I thank those I met or got to know better from the beginning of this journey: I will carry everything I have experienced with me forever,

both the moments linked to happy memories and the others.

I thank the thesis reviewers, who helped to improve the quality of the final result of this work.

I thank this discipline that I have studied for these eight years, which I have fallen in love with, which I am fascinated by, which I consider one of the purest forms of Science, which I consider one of the solid components of my identity.

Reflecting on the last three years, I cannot help but be moved.

Contents

List of Figures

List of Tables

Introduction

The 21st century has witnessed unprecedented technological advances that have revolutionised how we perceive and use data. This era of rapid technological development has led to exponential growth in collecting and storing vast amounts of information. With the advent of the digital age, the accumulation of data has become a phenomenon. Over the past two decades, the evolution of information technology has pushed us into a realm where traditional data storage and management methods are no longer sufficient. The sheer volume, velocity and variety of generated data required a paradigm shift in our approach. This paradigm shift paved the way for high and ultra-high dimensional data, encompassing diverse data types such as text, images, audio and video. To understand the scale of this data explosion, consider that the estimated amount of data stored at the beginning of 2003 was only 5 exabytes ($5 * 10^{18}$ bytes). Astonishingly, today, the same amount of data is generated in just two days, highlighting the staggering rate at which data is being generated (Sagiroglu and Sinanc, 2013). This unprecedented wave of data has ushered in a new era of opportunities and challenges. It has opened the door to cutting-edge technologies such as artificial intelligence, machine learning and big data analytics, enabling us to gain valuable insights and drive innovation across industries. However, it has also presented us with the immense challenge of efficiently storing, processing and extracting meaningful knowledge from these vast data repositories. To successfully navigate this era of

information abundance, individuals and organisations must harness the power of advanced data management techniques, data-driven decision making and scalable infrastructure. By harnessing the potential of big data analytics and using intelligent algorithms, we can unlock the transformative potential of these massive data sets. As we move further into the 21st century, the era of data-driven insights promises to reshape the world as we know it. By harnessing the ever-growing sea of data, we can revolutionise industries, solve complex problems and make impactful advances that can potentially improve the lives of individuals and societies on a global scale. This issue encompasses diverse domains of application, as noted by Donoho (2000): in recent years, there has been a significant increase in investments made towards data gathering and processing mechanisms across various industries. This growth primarily stems from developing, managing, and storing vast amounts of data for scientific, medical, engineering, and commercial purposes.

The availability of vast amounts of data, coupled with the emergence of new scientific problems, has fundamentally transformed the field of statistical thinking and data analysis. In particular, the rise of high-dimensional problems has made dimensionality reduction and feature extraction indispensable techniques (Fan and Li, 2006). So, the emergence of this challenge requires the development of new statistical methodologies to adapt to this new paradigm, which starkly contrasts with the one in which statistics was born. In the classical conception, statistics used to work in areas where the number of measured variables was small, or at least smaller than the number of observations that made up a sample. Statistical accuracy, model interpretability, and computational complexity are fundamental pillars of statistical procedures (Fan and Lv, 2010). Typically, in conventional studies, the number of observations n significantly exceeds the number of variables or parameters J. Consequently, maintaining all three aspects simultaneously does not require compromising efficiency. However, traditional methods encounter significant hurdles when the dimensionality J exceeds the sample size n. These challenges encompass the development of more efficient infer-

ence techniques, the establishment of asymptotic or nonasymptotic theory, the enhancement of interpretability for estimated models, and the achievement of computational efficiency and robustness in statistical procedures.

The scenario where the number of predictors significantly exceeds the number of observations presents numerous challenges. In classification contexts, attempting to utilize all dimensions for classification purposes would be counterproductive, as it would introduce noise and compromise the accuracy of the classification process (Fan and Fan, 2008; Hall et al., 2008). This issue becomes particularly pronounced when the dimensionality J greatly exceeds the sample size n, leading to many difficulties that need to be addressed (Fan and Lv, 2008). One of the primary challenges stems from unimportant predictors that exhibit a high correlation with the response variable due to the existence of significant associated predictors. This correlation often results in substantial spurious correlations, further complicating the task of variable selection. Distinguishing between the truly relevant and irrelevant predictors becomes inherently challenging in such cases. Addressing the issue of variable selection becomes imperative in scenarios where the number of predictors is overwhelmingly large. Identifying the subset of features that effectively capture the distinguishing characteristics between two groups is crucial for accurate classification. By carefully selecting the relevant predictors and excluding the irrelevant or noisy ones, we can improve the quality and reliability of the classification process.

Another context in which high and ultra-high dimensionality is tricky is model estimation. When the number of predictors or independent variables is much larger than the number of observations, this poses significant challenges and can lead to several problems in the estimation process. This is well-known as the curse of dimensionality (Donoho, 2000): as the number of predictors increases, the data becomes sparser in the high-dimensional space. This sparsity makes it difficult to estimate the relationships between predictors and the response variable accurately. The model may struggle to capture the true underlying structure of the data, leading to decreased pre-

dictive performance and reliability. Moreover, estimating models with high-dimensional data requires more computational resources and can be computationally intensive. As the number of predictors increases, the time and memory requirements for model estimation and evaluation also increase significantly. Dealing with large-scale datasets in high-dimensional settings can pose practical challenges regarding computational efficiency.

Classical statistical methods are not well-suited to address these emerging challenges in these cases. Utilizing such methods often necessitates compromising at least one of the three fundamental pillars mentioned earlier. Addressing these challenges requires advanced statistical techniques, regularization, and dimensionality reduction approaches. By carefully selecting informative predictors, managing model complexity, and employing appropriate strategies, it is possible to mitigate the adverse effects of high dimensionality and enhance the accuracy and interpretability of the estimated models. Different approaches exist to perform variable selection and produce sparse models, like test-based (e.g. Breaux (1967); Hendry and Richard (1987)) or screening-based approaches (e.g. Fan and Lv (2008)).

Outline of the thesis

This thesis first reviews of the best-known approaches to reduce the complexity of the models; next, it introduces a new proposal for selecting variables by a penalized approach.

The first chapter delves into the various methodologies employed for variable selection, covering test-based, screening-based, and penalty-based methods. It explains the fundamental principles behind each approach and underscores their significance in statistical modelling. The chapter also establishes the connection between these approaches and the context of the thesis.

The second chapter focuses on presenting the core of the thesis:

the Adaptive Non-Convex Penalty (ANP) function. It discusses the formalization of this function within the Generalized Linear Models (GLMs) and Gaussian Graphical Models (GGMs) frameworks and for grouped variables. The chapter emphasizes the unique features of the ANP function, striking a balance between convex and non-convex penalties for enhanced computational efficiency and variable selection.

The third chapter explains the methodology used to fit the penalized models, encompassing the ANP function in its different variations. It introduces the Alternating Directions Method of Multipliers (ADMM) algorithm as the primary estimation tool and elucidates the role of the Local Linear Approximation (LLA) technique in simplifying certain algorithmic steps.

This fourth chapter includes simulation studies conducted to evaluate the performance of the proposed ANP function against established penalization methods (LASSO, SCAD, MCP) across different scenarios. The simulations cover scenarios within GLM, grouped variables, and GGM frameworks.

The fifth chapter compares the proposal with traditional penalized models using real datasets. The chapter evaluates the effectiveness of the proposal in practical scenarios and discusses its application in GLM and GGM frameworks.

The concluding chapter summarizes the key findings and contributions of the thesis. It reiterates the significance of the proposed ANP function in addressing variable selection challenges and highlights its performance in various contexts.

Chapter 1

Variable selection

In the scientific literature, three primary approaches for variable selection are commonly considered: test-based, screening-based and penalty-based.

The first approach, known as test-based methods, involves automated statistical tests or computations to determine whether one or more variables should be included or excluded in the model. These methods were among the initial attempts at variable selection and aimed to assess the significance of variables. They provide a basis for deciding which variables should be included by conducting statistical tests or evaluating relevant quantities.

The second one, the screening-based approach, does not strictly focus on variable selection but instead aims to rank variables based on their importance. This approach is particularly useful in situations where the dimensionality is high, with a number of variables significantly larger than the number of statistical units. Screening-based methods often combine with other procedures to identify the most influential variables in ultra-high dimensional contexts.

Lastly, the penalty-based methods, refers to models that apply constraints on the estimated parameters to promote sparsity in the resulting model. These constraints encourage selecting a smaller set

of variables when the number of variables exceeds the number of statistical units. By penalizing the model for including unnecessary variables, penalty-based methods facilitate the identification of the most relevant variables.

In the context of this thesis, the proposed method belongs to the penalty-based approach. It utilizes penalties (or regularization) terms to promote the variable selection and to encourage sparsity in the fitted model.

1.1 Test-based approach

The primary aim of this method is to focus on explanatory and descriptive objectives. These methods determine the variables' significance and understand the underlying structure and relationships within the data. The goal is often to identify or to understand how different variables contribute to the variation in the response variable. The emphasis is on model interpretability and the ability to draw conclusions about the relationships between variables.

The most intuitive method belonging is the All-possible-regressions (Garside, 1965), that test all possible subsets of the set of potential covariates and selects the "best model" using some quantity that measures the model's goodness (Akaike Information Criterion (AIC, Akaike (1974)), Bayesian Information Criterion (BIC, Schwarz (1978)), Hannan-Quinn Information Criterion (HQIC, Hannan and Quinn (1979)), Mallows' Cp (Mallows, 1973)). If P potential covariates exist, the possible subsets to be tested are 2^P. Instinctively, this approach is computationally inefficient and time-consuming if many explanatory variables are available. In contexts where the number of variables exceeds the number of observations, the maximum number of estimated parameters must be less than or equal to the number of observations in the sample.

An alternative is a Stepwise regression, which consists of the iterative, step-by-step construction of a regression model involving the selection of covariates to be used. Operationally, it consists of adding

or removing potential explanatory variables in succession and checking the gain obtained at each step. The problem with these methods is that they involve estimating several regression models, and in high-numbered contexts, the time required can be considerable.

There are more efficient alternatives to be used in the literature, the best known of which are stepwise regression (Breaux, 1967) and Autometrics (Hendry and Richard, 1987).

1.1.1 Stepwise regression

The stepwise regression (Breaux, 1967) is one of the well-known techniques but currently is no longer widely used. It consists of a multistep technique. Each step evaluates whether to include or remove a variable based on a criterion defined by the person using the method. There were two possible approaches in its initial formulation: either estimate an initial null model and add a new variable at each step (forward step) or estimate a model with all variables and remove one at a time (backward step). At each step, the introduction or removal of a variable is done based on some criterion, which can be the minimization of some information criterion or considering the trend of the corrected R^2. It is important to note that the models identified by these stepwise methods need not be the same: Berk (1978) shows that the differences among the subset selection procedures are surprisingly small, with one exception, no more than 7%. Further, these procedures do not attempt to identify the 'best' models in that they do not necessarily locate the model with the minimum residual sum of squares. The first criticism of the method is that it explores only one of the possible paths of the model (since, at each step, one can add or remove a variable). Moreover, a common misconception when using stepwise regression is to assign greater importance to variables that are included or removed early in the model-building process. The order in which variables are included or removed does not necessarily reflect their true importance. An initially included variable may become irrelevant or redundant in the presence of other variables. Therefore, it is important to avoid misleading interpre-

tations based solely on the order of variable inclusion or removal in stepwise regression. To overcome this problem, a used solution is to evaluate, at each step, if one of the variables is present in the model or add a variable simultaneously. However, this does not entirely solve the problem. Hurvich and Tsai (1990) demonstrate with a simulation study the stepwise procedure in a simple setting often fails to select the true data-generating paradigm and suffers from undercoverage of the estimated parameter coverage rates. However, they also conclude that this is a broader problem and not only typical of this technique: they argue that this problem arises from using only one dataset for model selection and inference. They argue that a possible solution to this problem is to perform model selection and inference on separate parts of the dataset. Derksen and Keselman (1992) observed that the degree of correlation between the predictors dramatically influences the rate of variables correctly present in the final model. The degree of collinearity is the component that most influences the final result's goodness. For this reason, it is essential to choose the variables carefully to be used in the stepwise regression, avoiding selecting too many variables. The same study shows that the sample size is essential to the procedure's success, but this is not as important as the correct selection of the variables to be considered. A mistake often made when using stepwise regression is to consider the variables that are included (or removed) firstly in the model as more (or less) important. This can be misleading since, for example, it is not uncommon to find that the first variable included is useless in the presence of other variables (Hocking, 1976). It could happen that when a variable's introduction and removal are evaluated at each step, one of the first variables to enter the model is removed. This can also be attributed to the correlation between the variables considered.

1.1.2 Autometrics

Autometrics is a powerful approach to automated variable selection in econometric models. Developed by Hendry and Richard (1987),

it was designed to address the challenges associated with traditional manual variable selection methods, such as subjectivity and potential misspecification.

Autometrics employs an automated search algorithm to facilitate the iterative selection and evaluation of potential variables for model inclusion. It adheres to a general-to-specific approach, starting with a comprehensive General Unrestricted Model (GUM) encompassing all feasible specifications. The algorithm systematically assesses the significance and relevance of candidate variables through successive iterations, progressively refining the model to a simpler and more congruent specification. To address high-dimensional features and non-identifiability (when the number of variables exceeds the number of observations) in the General Unrestricted Model, Autometrics employs a straightforward approach called "block search". This technique involves dividing the regressors into smaller blocks, ensuring that the size of each block is less than the number of variables (J). Subsequently, tests are conducted on each block, leading to the removal of irrelevant variables. The remaining blocks are then merged, and the iterative process continues.

During the iterative process of model complexity reduction, Autometrics employs various criteria to evaluate the importance of each variable. Information criteria such as the AIC and BIC are considered to provide insights into the trade-off between model fit and complexity. Additionally, Autometrics utilizes statistical tests, including t-tests and F-tests, to assess the significance of individual variables. Variables with meaningful effects on the response variable are retained, while those deemed insignificant are discarded. Moreover, Autometrics recognizes the significance of outliers and incorporates their detection into the model selection process, using, for example, the Impulse Indicator Saturated Selection (IIS) developed by Santos et al. (2008). This method adds a set of indicators to the GUM for each observation. It applies tests in a block search manner to identify and remove observations that are inconsistent with the model, effectively identifying outliers. By integrating these techniques, Autometrics ensures the identification and poten-

tial exclusion of observations that do not align with the overall model structure.

One of the key advantages of Autometrics is its ability to deal with the problem of model overfitting. This occurs when a model becomes too complex and begins to capture noise or idiosyncratic data features, leading to poor out-of-sample performance. By reducing overly complex models, Autometrics helps prevent overfitting and promotes more reliable and robust model selection. In addition, Autometrics can handle endogeneity (i.e., when the relationship between variables is influenced by factors not adequately accounted for in the analysis) and simultaneity (i.e., when the explanatory variable is jointly determined with the dependent variable), which are common challenges in econometric modelling. Its iterative search algorithm considers the potential presence of endogenous variables and adjusts the model specification accordingly. This feature makes Autometrics particularly suitable for complex economic models involving interdependencies between variables. Another notable strength of Autometrics is its flexibility in dealing with structural breaks or regime shifts. Traditional econometric models often assume constant relationships over time, which may be false. Autometrics allows the inclusion of dummy variables or structural break indicators to capture relationship shifts, ensuring more accurate and dynamic modelling. Despite its many advantages, Autometrics is not without its limitations. The automated nature of the procedure means that the researcher must interpret the results carefully and be cautious in making conclusions. While Autometrics greatly simplifies the variable selection process, it still requires careful consideration of the underlying theory, data quality, and potential omitted variable bias. In addition, Autometrics assumes that the true data-generating process is well approximated by the candidate variables available for selection. If important variables are not included in the initial pool of candidates, Autometrics may fail to identify them, resulting in a misspecified model. Therefore, it is crucial to be careful in selecting candidate variables to ensure an accurate representation of the underlying relationships. Autometrics is available as an R package

named `gets` (Pretis et al., 2018).

1.2 Screening-based approach

Screening methods represent a class of alternative techniques that provide an intuitive and effective solution to the task of variable selection, especially in the context where $J >> n$. The main goal is to improve computational efficiency and manage the dimensionality of the data, particularly in ultra-high-dimensional contexts. These methods aim to reduce the feature space to a manageable size.

Introduced by Fan and Lv (2008) and subsequently extended, these methods are based on measures of association between the dependent variable and potential regressors. In this approach, variables are ranked according to their marginal association with the response variable, allowing for the rapid identification of promising candidates for inclusion in regression models. The convenience of the screening-based methods lies in their computational efficiency, enabling the rapid reduction of high-dimensional feature spaces without the exhaustive computational burden.

However, it is essential to note that this approach does not directly select variables to include in models. Instead, it progressively refines the candidate set by eliminating the less important variables, leaving the selection of the final model to subsequent procedures (e.g. with a penalty-based approach). This solution demonstrates particular strength when dealing with situations where the number of variables greatly surpasses the number of observations, a scenario that challenges conventional selection methods.

1.2.1 Sure Independence Screening

The Sure Independence Screening (SIS, Fan and Lv (2008)) technique is a feature selection method that evaluates the importance of each feature independently, primarily based on its correlation with the response variable. Features with stronger correlations are consid-

ered more important. One of the key advantages of SIS is its ability to maintain model sparsity, ensuring that only a subset of the available features is included in the selected model. This is particularly valuable in high-dimensional data scenarios. SIS also possesses the "sure screening property", which guarantees that important features are included with high probability, provided specific conditions are met. This property makes SIS a reliable technique for identifying relevant variables. After reducing the dimensionality with SIS, traditional variable selection methods can be applied to the reduced feature space. This combination of techniques allows for effective feature selection and model building in complex data analysis tasks.

However, SIS has some limitations. The most important is that it requires setting a threshold for variable selection, which determines the number of excluded variables. The choice of this threshold can be somewhat arbitrary and may benefit from optimization through techniques like cross-validation or other model selection methods.

The Iterative Sure Independence Screening (ISIS) is an extension of the Sure Independence Screening technique designed to improve variable selection accuracy while addressing the challenge of selecting weakly correlated variables. ISIS takes an iterative approach to variable selection: it considers the correlations between features and the residuals obtained from a model that primarily relies on the already selected variables. This conditional approach allows for a more refined ranking of variables, considering the dependencies among predictors. The core idea of ISIS aligns with the broader concept of reducing false positives in variable selection. Considering variables' conditional correlations within the selected subset enhances the screening process and provides a more accurate selection of relevant features.

1.2.2 Reduction of False Positive Rate

To mitigate false positive results, which often plague screening methods, a promising approach is to employ a resampling technique suggested by Fan et al. (2009b). This technique involves a simple yet

effective strategy: randomly splitting the samples into two halves. From these halves, two sets of active variables, denoted as $\hat{\mathcal{A}}_1$ and $\hat{\mathcal{A}}_2$, are derived.

The key insight here is that if both $\hat{\mathcal{A}}_1$ and $\hat{\mathcal{A}}_2$ exhibit a certain property related to screening, then the composite set $\hat{\mathcal{A}}$ also possesses this property. On the contrary, $\hat{\mathcal{A}}$ contains considerably fewer falsely selected variables. This reduction in false positives can be attributed to the rarity of an unimportant variable being randomly selected twice within the vast, high-dimensional space. Thus, $\hat{\mathcal{A}}$ effectively minimises the number of false positive variables.

Under specific exchangeability conditions, as explained by Fan et al. (2009b), it can be established that the likelihood of selecting at least r inactive variables is exceedingly low when n is substantially smaller than J. This is particularly applicable to scenarios discussed in the preceding two sections.

1.2.3 Covariate Assisted Screening and Estimation

The Covariate Assisted Screening and Estimation (CASE, Ke et al. (2014)) is a method proposed to address the challenges of variable selection when dealing with rare and weak signals and a non-sparse Gram matrix. The method is designed to exploit sparsity in the Gram matrix while mitigating the issues introduced by sparsifying.

CASE begins by constructing a sparse graph called the "Graph of Strong Dependence" (GOSD) using the Gram matrix G. This graph captures the relationships and dependencies between the design variables. This strategy effectively reduces the problem's dimensionality while retaining most of the signals. Unlike brute-force screening of all possible submodels, CASE uses information from G to decide which size-m submodels to consider. CASE includes a size-m submodel in the screening list if the m corresponding nodes in the GOSD form a connected subgraph. In other words, CASE screens submodels based on the connectivity of nodes in the graph, which allows for more efficient screening. This method is designed

for scenarios where signals are rare and weak. It adapts to the challenges posed by these types of signals and the specific characteristics of the Gram matrix.

1.3 Penalty-based approach

The penalty-based approach primarily aims to create parsimonious models that balance the trade-off between model complexity and predictive accuracy. These methods focus on selecting variables contributing to the model's predictive performance: they are particularly valuable in prediction contexts.

To overcome some of the theoretical limitations of the other approaches, the literature has focused on using sparse estimators derived from the idea of adding an appropriate penalty function to the objective function. If a penalty function is "well-defined", the model can perform model estimation and selection simultaneously. The sparsity assumption is formalized by saying that the J-dimensional vector of parameters, denoting by $\xi \in \Xi$, has a sparse structure, meaning that exists a set $\mathcal{A} \subset \{1, ..., J\}$ such that $\mathring{\xi}_j \in \mathcal{A}$ iff $\xi_j \neq 0$.

The basic idea of the penalized approach is to maximize the "loss function" of the model to the data (i.e., to maximize the fit of the model to the data), adding a "penalty" component that is a function of the estimated parameters. Let us denote a generic convex objective function by $f(\xi)$, such as residual deviance, log-likelihood or check function. Penalized models are usually grounded on estimating parameters and minimizing a new objective function

$$\hat{\xi} = \arg\min_{\xi} f(\xi) + p_\lambda(|\xi|),$$

where $p_\lambda(|\xi|)$ is a penalty function whose statistical properties of the estimator $\hat{\xi}$ are reflected.

λ is the tuning parameter, and the choice of it is crucial in penalized likelihood estimation. When the penalty parameter λ is set to 0, all variables are selected, which can lead to an unidentifiable

model when the number of variables J exceeds the sample size n. On the other hand, when λ is very large, and the penalty function satisfies certain conditions, no variables are selected. The optimal choice of λ lies between these extreme values. The value of λ controls the complexity of the selected model. A larger λ tends to result in a simpler model with smaller variance in the estimation, while a smaller λ leads to a more complex model with smaller bias. Balancing these biases and variances allows for an optimal choice of λ, often determined using techniques like multi-fold cross-validation. Wang et al. (2007a) found that generalized cross-validation selects models containing all important variables but may include some unimportant ones with nonzero probability. On the other hand, the model selected using BIC achieves model selection consistency and an oracle property, which means it correctly identifies the true model with high probability. When choosing a penalty, it is essential to consider model misspecification since missing true predictors or misspecifying the distribution family can lead to errors. Lv and Liu (2014) proposed a semi-Bayesian information criterion (SIC) to address this issue and improve model selection accuracy.

The literature on penalty functions is now boundless, and conducting an exhaustive review of them would require dedicated work. See, for example, Fan and Lv (2010) or Desboulets (2018) for broad perspectives on high-dimensional statistical problems. As can be guessed, the optimal inferential properties of $\hat{\xi}$ depend heavily on the penalty function. This leads to formalising which properties guarantee optimal properties for the estimator. Formally, an estimator produced by a proper penalty function should enjoy the three properties stated by Fan and Li (2001):

1. sparsity: the estimator can reduce the number of parameters, thus setting 0 for the noise-source parameters; a sufficient condition is that $\arg\min_{\xi} |\xi| + p'_{\lambda}(|\xi|)$ is positive;

2. continuity: the estimator is continuous in the data; in other words, a slight change in the data should not result in a significant change in the estimates; a necessary and sufficient

condition for the penalty function to be continuous is that $\arg\min_{\xi} |\xi| + p'_{\lambda}(|\xi|) = 0$;

3. unbiasedness: the estimator is nearly unbiased for large parameters; a sufficient condition is that $\lim_{\xi \to \infty} p'_{\lambda}(|\xi|) = 0$.

The first and second conditions imply that a penalty function must be discontinuous at the origin to meet the requirements.

The authors further introduce the concept of the oracle property, which is crucial to assess sparse estimator. To meet the oracle property asymptotically, the estimator must fulfil the following conditions:

- Identifying the correct subset of non-zero coefficients: the estimated coefficients should correctly identify the subset of non-zero coefficients, denoted as $\{\hat{\mathcal{A}} = j : \hat{\xi}_j \neq 0\}$, which should match the true subset of non-zero coefficients denoted as $\mathcal{A}$.

- Optimal convergence rate: the estimator should possess an optimal estimation rate, given by $\sqrt{n}(\hat{\xi}_{\mathcal{A}} - \xi_{\mathcal{A}}) \xrightarrow{d} N(0, \Sigma)$, where Σ represents the covariance matrix based on the knowledge of the true subset model.

When satisfied by a sparse estimator, these properties indicate that it performs optimally in identifying the correct non-zero coefficients and achieving the optimal estimation rate concerning the true underlying model. From a computational perspective, penalty functions should be chosen so that the resulting optimization problem is straightforward to solve. Focusing on the penalties able to perform variable selection, it is possible to divide the penalty functions into two classes, defined according to the geometric features of the penalty function, i.e. convex and non-convex. Table 1.1 lists some examples. Figures 1.1, 1.2 and 1.3 show the penalty functions together with their derivatives and the threshold operator. Figure 1.4 portraits the penalties function for 2 coefficients.

Table 1.1: Some penalty functions allow variable selection in regression models.

Name and reference	Penalty		
1. Hard thresholding Antoniadis (1997)	$p_\lambda(\lvert\xi\rvert) = \lambda I(\lvert\xi\rvert > 0)$		Convex
2. LASSO Tibshirani (1996)	$p_\lambda(\lvert\xi\rvert) = \lambda\lvert\xi\rvert$		Convex
3. Adaptive LASSO Zou (2006)	$p_\lambda(\lvert\xi\rvert) = \lambda w\lvert\xi\rvert$		Convex
4. Bridge Frank and Friedman (1993)	$p_\lambda(\lvert\xi\rvert) = \lambda\lvert\xi\rvert^\gamma$	$\gamma \in (0,1)$	Non convex
5. SCAD Fan and Li (2001)	$p'_\lambda(\lvert\xi\rvert) = \lambda\left\{I_{\lvert\xi\rvert\leq\lambda} + \frac{(\gamma\lambda-\lvert\xi\rvert)_+}{(\gamma-1)\lambda} I_{\lvert\xi\rvert>\lambda}\right\}$	$\gamma > 2$	Non convex
6. MCP Zhang (2010a)	$p'_\lambda(\lvert\xi\rvert) = \lambda\left(1 - \frac{\lvert\xi\rvert}{\gamma\lambda}\right) I_{\lvert\xi\rvert\leq\lambda}$	$\gamma > 1$	Non convex

For SCAD and MCP, the additional parameters are, usually, $\gamma_{\text{SCAD}} = 3.7$ and $\gamma_{\text{MCP}} = 3$.

1.3.1 Convex penalty function

Convex penalties were the first functions to be proposed and have the advantage of being easy to optimise due to their mathematical properties, allowing efficient algorithms. However, they have the disadvantage of providing biased estimates, which can affect the accuracy of the results.

The Least Absolute Shrinkage and Selection Operator (LASSO) is undoubtedly the most popular convex penalty function. It is a special case of the more general penalty function, the Bridge penalty function, proposed by Frank and Friedman (1993) and studied deeply by Fu and Knight (2000). As seen from Table 1.1, the LASSO can be obtained from the Bridge penalty function by setting γ equal to 1, i.e.

$$p_\lambda(|\xi|) = \lambda|\xi|.$$

The LASSO estimator encourages both shrinkage and selection, providing a balance between model simplicity and predictive accuracy. It enjoys excellent computational properties: stable algorithms can be easily implemented and do not require complex optimizations. An example is the *Least Angle Regression* (LARS) algorithm Efron et al. (2004): it provides the whole solution path as a function of the tuning parameter λ. Another very used algorithm is the *Coordinate Descent* (Friedman et al., 2010b): it sequentially updates coefficients cyclically and separately, maintaining the other coefficients as constants. Augugliaro et al. (2013) propose a method that addresses monotonically decreasing sparsity for outcomes modelled by an exponential family. This generalizes the equiangular condition in a generalized linear model, where the Fisher information plays a crucial role. While the computation of solution paths is non-trivial, the method demonstrates favourable comparisons with other path-following algorithms.

On the other hand, the estimator suffers from a non-negligible bias proportional to the tuning parameter λ: the LASSO penalty

enjoys the properties of sparsity and continuity but not unbiasedness because

$$\lim_{\xi \to \infty} p'_{LASSO,\lambda}(|\xi|) = \lambda.$$

Several authors have carefully investigated the LASSO model consistency (Fu and Knight, 2000; Zou, 2006; Zhao and Yu, 2006): broadly speaking, the LASSO recovers the true set $\mathcal{A}$ under specific conditions on the covariance among covariates. In particular, Zou (2006) shows that the inequality below

$$|cov(X_{\bar{\mathcal{A}}}, X_{\mathcal{A}})cov(X_{\mathcal{A}})^{-1} sign_{\mathcal{A}}| \leq 1,$$

called *irrepresentable condition* or *neighbourhood stability* (Meinshausen and Bühlmann, 2006), must hold for LASSO to be consistent in variable selection where X can be the set of J covariates or the set of J variables for which the partial correlation structure is studied. The inequality is intended component-wise, and $sign_{\mathcal{A}}$ is the vector of signs of the true parameter values. The asymptotic results provide insights and guidance for utilizing LASSO as a feature selection tool, assuming that the standard regularity conditions on the design matrix, as presented in Fu and Knight (2000), are satisfied. So, verifying whether the Irrepresentable Conditions hold for a given dataset is crucial. If these conditions are not met, regardless of the λ value, LASSO may not accurately select the appropriate model.

The LASSO estimator exhibits a bias proportional to the size of the tuning parameter λ. This bias contributes to increased error in the estimator. Consequently, the LASSO estimator tends to select smaller values of λ during model selection, as it aims to mitigate the bias-related issues to reduce the mean squared error (Fan and Lv, 2010). However, opting for a smaller λ value leads to selecting a more complex model. This complexity arises from the trade-off between reducing bias and controlling the number of selected variables. Consequently, the LASSO estimator often includes numerous false

positive variables in the chosen model. This phenomenon can be attributed to the inherent bias of the LASSO estimator, which drives the preference for smaller λ values and, subsequently, the inclusion of additional variables that may not be truly significant.

Zou (2006) proposed a solution to address certain limitations of the LASSO method by introducing a modified version called the Adaptive LASSO (AdaLASSO). He proposes the Adaptive LASSO as a solution to address the inconsistency issues observed in the variable selection process of the traditional LASSO method. The Adaptive LASSO achieves variable selection consistency using a weighted LASSO penalty. This means that instead of assigning the same tuning parameter value for all coefficients, the Adaptive LASSO allows different λ_j, such that $\lambda_j = w_j\lambda$, for each coefficient. The weights are data-dependent and carefully chosen to optimize the variable selection performance. Zou (2006) define the weight vector as

$$w = \frac{1}{|\hat{\xi}|^\gamma},$$

where $\hat{\xi}$ is a root-n–consistent estimator of the true coefficient, usually coming from the unpenalized estimator (if $n > J$), and $\gamma > 0$. For this reason, the Adaptive LASSO is a two-step approach. By introducing this adaptively weighted penalty, the Adaptive LASSO can estimate the coefficients using different intensities of shrinkage. Note how this allows some of the characteristics of non-convex penalties to be enjoyed while retaining some of the advantages of convex penalties.

A notable advantage of the Adaptive LASSO is its computational efficiency. It can be solved using efficient path algorithms like those employed in the traditional LASSO method. This also ensures that the Adaptive LASSO can be applied to large-scale problems without sacrificing computational feasibility.

One key advantage of the Adaptive LASSO is its "oracle property," which implies superior statistical performance w.r.t. LASSO. However, its computational efficiency sets it apart from other ora-

cle methods. Unlike other penalty functions that enjoy the oracle property, the computational cost of Adaptive LASSO is comparable to that of LASSO. The estimation process involves solving a convex optimization problem based on an initial estimator, and the computational complexity remains the same as that of the LASSO. (Huang et al., 2008).

It is worth noting that while the Adaptive LASSO enjoys the oracle properties, which indicate its excellent performance when the true underlying model is known, optimal prediction performance is not automatically guaranteed.

1.3.2 Non-convex penalty function

The most well-known penalties for this group are SCAD (Fan and Li, 2001) and MCP Zhang (2010a).

Non-convex penalty functions can overcome some of the limitations of convex penalties. They can potentially provide almost unbiased solutions while allowing variable selection. On the other hand, there are two primary challenges in this context. First, computational issues arise, given the difficulty of optimizing a non-convex function. Second, there is the possibility of encountering multiple local solutions. It's crucial to note the interconnection between these two problems.

Two factors make a convex objective function desirable. First, convexity guarantees that any algorithm that converges to a critical point of the objective function also converges to the only global minimum. Second, convexity guarantees that $\hat{\xi}$ is continuous w.r.t. λ, which decreases the number of iterations the algorithm needs to converge. Additionally, this makes selecting a suitable regularization parameter value less difficult. From a statistical point of view, convexity can also be preferred. Without it, $\hat{\xi}$ is not necessarily continuous for the data, which means that a small change in the data could result in a significant difference in the estimate (Breheny and Huang, 2011).

Defining coordinate subspaces as a subspace of $\mathbb{R}^J$ (where each

component is referred to as a covariate), a possible solution to check what the global optima might is to consider the union of coordinate subspaces, which allows examining the global optimality. So, the union of all-dimensional coordinate subspaces of $\mathbb{R}^J$ is used to investigate the global optimality of $\hat{\xi}$. However, it is challenging to demonstrate the global optimality of a local maximizer when $J >> n$ (Fan and Lv, 2011; Bühlmann and Meier, 2008).

Numerous algorithms have been proposed in the literature to address penalized likelihood optimization issues using non-convex penalties. Despite having fast processing times and relatively easy implementation, none of these algorithms can address the case of multiple local optimums. The LLA (Zou and Li, 2008) and coordinate descent algorithms (Breheny and Huang, 2011) are not guaranteed to reach a global minimum, especially when λ is small. Despite having a non-convex penalty component, it is still possible for the objective function to be convex for ξ, under certain condition that changes for different penalty functions. Gasso et al. (2009) claimed that while a single iteration is undoubtedly computationally inexpensive, the optimality of such a plan is debatable because convergence to a local or global minimum of the optimization problem is not ensured. For this, the authors propose an algorithm based on the Difference of Convex (DC) functions programming (Horst and Thoai, 1999): a non-convex objective function is divided into the difference of two convex functions, and the resulting problem is then solved using a primal-dual strategy. Although the method ensures convergence to a local optimum, it does not ensure that it is the global optimum. In fact, from numerical studies, the authors believe that the algorithm being stuck in some "bad" local minimal is the primary cause of the worse performance.

Yet another crucial consideration is the non-separability of some of this penalty from the tuning parameter λ. Some penalties are separable from the tuning parameter, i.e. $p_\lambda(\cdot) = \lambda p(\cdot)$. The separable penalties are preferable from a computational perspective because path-following algorithms can be used, making it possible to quickly compute the entire regularization path. In contrast, the

path-following algorithm cannot be used for non-separable penalties, and the proposed algorithm appears to be much less effective for approximating all regularization paths (Bühlmann and Meier, 2008).

The first non-convex proposal was proposed by Fan and Li (2001), which introduces a penalty function called the Smoothly Clipped Absolute Deviation (SCAD) penalty

$$p_\lambda(|\xi|) = \begin{cases} \lambda|\xi| & \text{if } |\xi| \leq \lambda, \\ \frac{2\gamma\lambda|\xi| - \xi^2 - \lambda^2}{2(\gamma-1)} & \text{if } \lambda < |\xi| < \lambda\gamma, \\ \frac{\lambda^2(\gamma+1)}{2} & \text{otherwise.} \end{cases}$$

This penalty function is designed to improve upon the properties of both the L1 penalty and the hard thresholding penalty functions. As can also be seen graphically in Figures 1.1 and 1.2, the SCAD penalty in the neighbour of ξ is exactly LASSO, depending on λ; thereafter the degree of shrinkage decreases until it is exactly zero. It can be visualized as a quadratic spline function with knots in $\lambda\gamma$ and λ. The SCAD penalty offers several advantages over other penalty functions. Firstly, it avoids excessive penalization of large parameter values. Additionally, it ensures a continuous solution, which is desirable in many practical scenarios. The authors indicate that with the appropriate selection of regularization parameters, the proposed estimators perform on par with the oracle procedure for variable selection, enjoying the three properties of sparsity, continuity and unbiasedness.

Kim et al. (2008) studied SCAD's properties when $J >> n$, a case where assessing whether a solution is sub-optimal is more difficult. This shows that the prediction accuracy is inferior to the oracle estimator, partly due to the sub-optimal selection of the regularization parameter. Moreover, they provide sufficient conditions under which the global optimum of the SCAD penalty is asymptotically equivalent to that of the oracle estimator. Although this result is interesting, the conditions are relatively strong, assuming that $J \leq n$.

The Minimax Concave Penalty (MCP) is similar to that of SCAD: it is non-convex, but the derivative of the penalty function decreases from the beginning, as shown in Figure 1.2. The penalty is defined as

$$p_\lambda(|\xi|) = \begin{cases} \lambda|\xi| - \frac{\xi^2}{2\gamma} & \text{if } |\xi| \leq \lambda\gamma, \\ \frac{1}{2}\gamma\lambda^2 & \text{otherwise.} \end{cases}$$

In his work, Zhang (2010a) presents the MC+ algorithm, a new method for variable selection in high-dimensional linear regression. MC+ combines the MCP with a penalized linear unbiased selection (PLUS) algorithm. It overcomes biases in the LASSO and computational costs of subset selection. MCP achieves consistent variable selection without assuming strong conditions, especially when the number of variables is much larger than observations. It demonstrates a high probability of correct sign matching, attains convergence rates for coefficient estimation, estimates noise levels, and establishes continuity conditions. Simulation results validate MCP's effectiveness.

The MCP employs a specific formula to balance concavity and convexity, and it aims to minimize a metric of maximum concavity under certain conditions. These conditions ensure unbiasedness and selection features. The MCP operates by achieving a trade-off between unbiasedness and concavity through regularization.

The presence of the additional parameter γ gives rise to an unbroken sequence of penalty and threshold operators, ranging from $\gamma \to \infty$ (representing the soft threshold operator) to $\gamma \to 1+$, corresponding to the hard threshold operator (Mazumder et al., 2011). This property establishes the MCP has a continuous spectrum, bridging the domains of soft and hard thresholds.

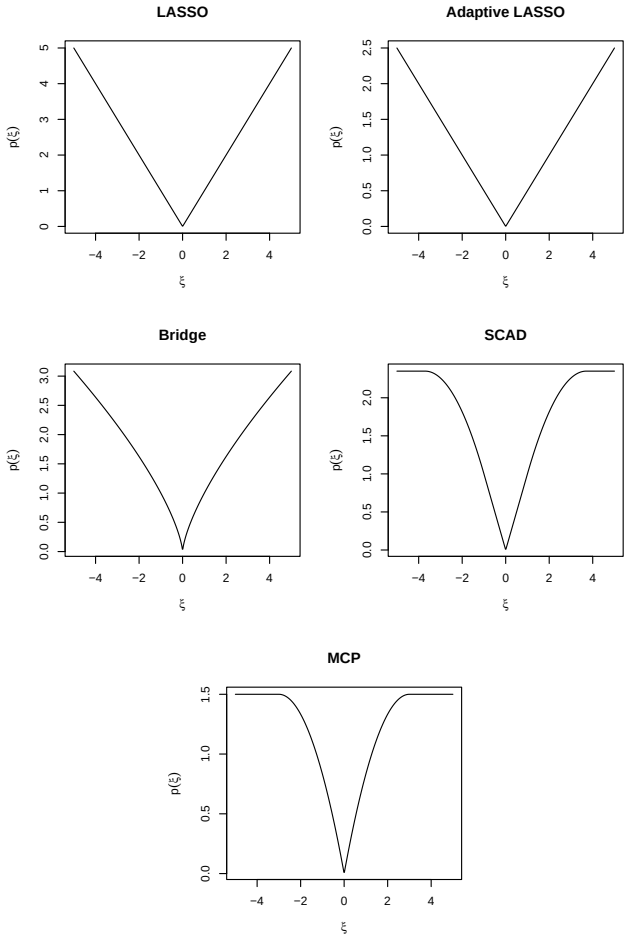


Figure 1.1: The four penalty functions. The values used for the additional parameters are $\gamma_{\text{Bridge}} = 0.7$, $\gamma_{\text{SCAD}} = 3.7$, $\gamma_{\text{MCP}} = 3$, $w_{AdaLASSO} = 0.5$ and the tuning parameter is $\lambda = 1$.

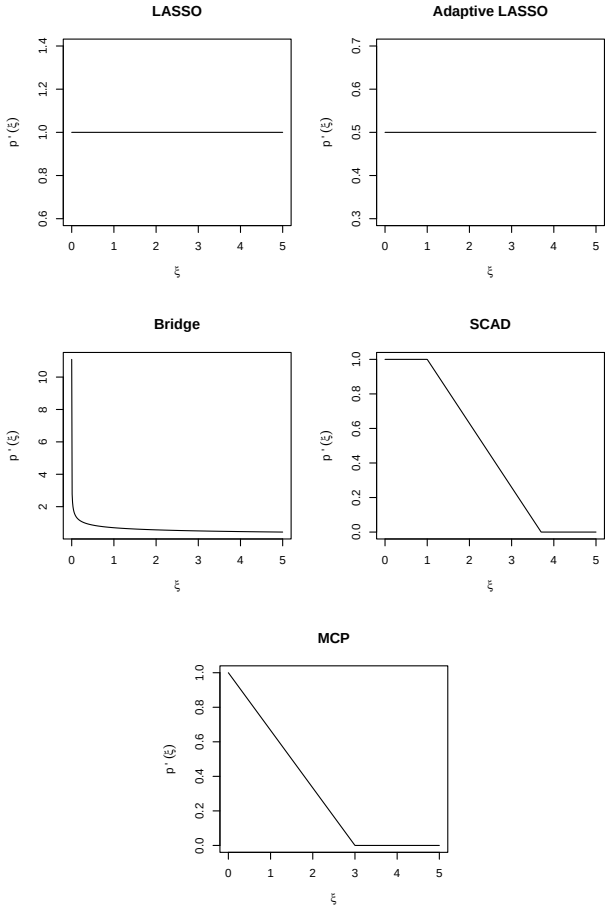


Figure 1.2: The derivatives of the four penalty functions, displayed over $\xi > 0$. The additional parameter values are set as $\gamma_{\text{Bridge}} = 0.7$, $\gamma_{\text{SCAD}} = 3.7$, $\gamma_{\text{MCP}} = 3$, $w_{AdaLASSO} = 0.5$ and the tuning parameter is $\lambda = 1$.

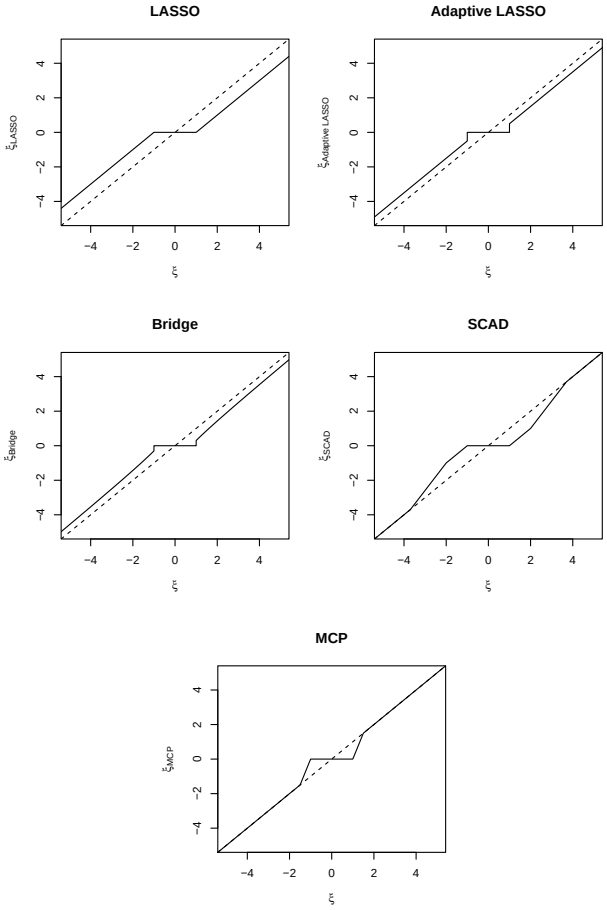


Figure 1.3: Thresholding operator of the four penalty functions, displayed over $\xi > 0$. The additional parameter values are set as $\gamma_{\text{Bridge}} = 0.7$, $\gamma_{\text{SCAD}} = 3.7$, $\gamma_{\text{MCP}} = 3$, $w_{AdaLASSO} = 0.5$ and the tuning parameter is $\lambda = 1$.

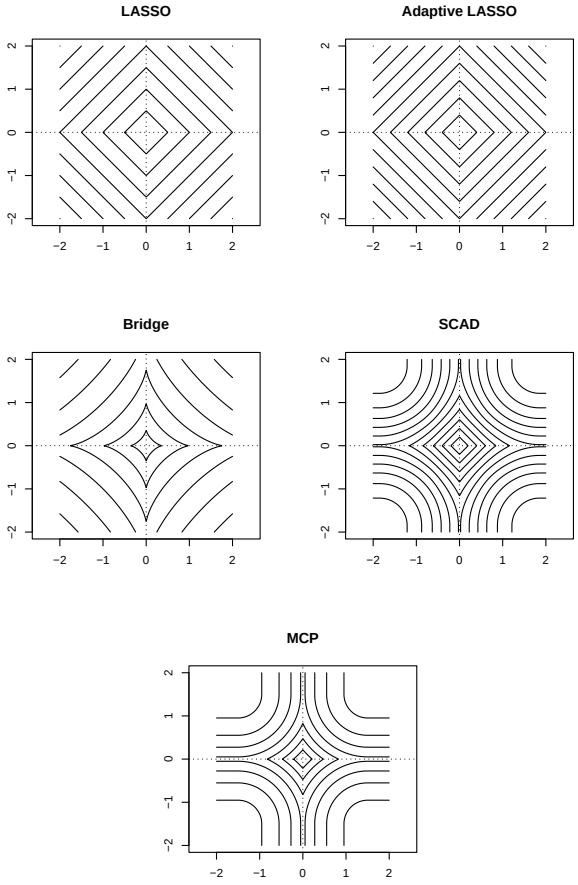


Figure 1.4: The four penalty functions in two dimensions. The values chosen for the additional parameters are $\gamma_{\text{Bridge}} = 0.7$, $\gamma_{\text{SCAD}} = 3.7$, $\gamma_{\text{MCP}} = 3$, $w_{AdaLASSO} = 0.5$ and the tuning parameter is set to $\lambda = 1$.

1.3.3 Other penalty function

The penalty functions described are the best-known and most studied. There are also countless other proposals.

Breiman (1995) introduces the Non-Negative Garrotte (NNG) as a novel method in subset selection regression. The penalty function is defined as

$$p_\lambda\left(|\xi|\right) = \left(1 - \frac{\lambda}{\hat{\xi}_j^{OLS^2}}\right)$$

The method is scale-invariant, and its performance is particularly noteworthy when a significant proportion of the true coefficients are non-zero. Comparatively, it performs as well or better than the competitors (known in 1995) in predictive accuracy. Simulation results highlight its stability and ability to find accurate predictors. While the NNG estimates models with more non-zero coefficients than conventional subset selection, the gains in accuracy compensate for the increase in complexity.

Candes et al. (2008) propose a penalty function that encourages sparsity more effectively than the traditional LASSO. The penalty function they introduce is a weighted logarithmic sum of absolute values that provides a more aggressive penalty for nonzero values than the LASSO, particularly for values close to zero. This encourages the optimization algorithm to push more coefficients to exactly zero, promoting sparsity. This approach aims to strike a balance between the sparsity-promoting properties of the Hard thresholding (which counts the number of nonzero entries but is computationally challenging) and the convexity and efficiency of the LASSO.

Lv and Fan (2009) presents the Smooth Integration of Counting and Absolute Deviation (SICA)

$$p_\lambda\left(|\xi|\right) = \lambda\frac{(a+1)\xi}{a+\xi},$$

where $a > 0$ is a shape parameter. It provides a unique method to finely adjust feature sparsity, considering both feature presence

and coefficient magnitudes. It achieves a balanced compromise between Hard thresholding and LASSO with an adjustable additional parameter.

Zhang (2010b) introduces the Capped-L1 penalty defined as

$$p_\lambda(|\xi|) = \lambda \min(a, |\xi|), \quad a \geq \xi.$$

This function encourages certain parameters to be zero while allowing others to retain their original magnitudes. The regularization increases linearly with the magnitude of the coefficients up to a predefined threshold or "cap." Beyond this cap, the penalty remains constant, preventing the coefficients from diminishing further. Capped-L1 effectively controls the sparsity of the solution, striking a balance between shrinking some coefficients towards zero while preserving the importance of others.

Belloni et al. (2011) introduces the Square-Root LASSO: this method overcomes the limitations of the conventional LASSO by formulating it as a conic programming problem. This allows for efficient algorithmic solutions. The proposal evaluates the performance of the square-root LASSO through Monte Carlo experiments, comparing it to the traditional LASSO and related methods.

Chapter 2

The Adaptive Non-Convex Penalty Function and the frameworks

As discussed in the first chapter, the primary objective is to address the challenge of variable selection. To achieve this goal, we present a novel penalty function to carry out variable selection. This chapter introduces the new penalty function designed to improve the variable selection process.

The following sections will first focus on formalizing our penalty approach; it occupies a position in the spectrum of penalty functions close to MCP and SCAD, striking a balance between the advantages of both convex and non-convex penalties. This equilibrium enables us to preserve the beneficial properties associated with non-convex penalty functions, including enhanced variable selection, while also harnessing the computational benefits of convex penalties. Some

comparisons are presented in the last chapter. Then, we will delve into the Generalized Linear Models (GLM) and Gaussian Graphical Model (GGM) domains. Within the GLM framework, we will also specifically formulate the penalty function for grouped variables. Once we have established this formal framework, we will proceed to introduce our novel penalty function.

2.1 Methodological proposal

We define the Adaptive Non-Convex Penalty function (ANP) acting on the generic model parameter ξ_j,

$$p_\lambda(|\xi_j|) = \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|\xi_j|}{\nu}\right). \tag{2.1}$$

where $\lambda > 0$ is the usual tuning parameter, $\Phi(\cdot)$ is the standard Normal cdf and ν is an additional scale parameter that affects both computational and inferential aspects. ν influences the "degree of nonconvexity" of the penalty. It determines the amount of bias of the resulting estimator. However, it turns out that ν also affects, to some extent, the non-uniqueness of the solution. More specifically, a large ν ensures the uniqueness of the solution, but the estimates will be biased. On the other hand, a small ν can lead to severe non-convexity, causing possible local optima in the resulting penalized likelihood. However, as we will discuss in Section 3.6, for any λ-value it is possible to find a lower bound for ν, denoted by $\nu_{\lambda,min}$, such that the solution is unique for any $\nu(\lambda) \geq \nu_{\lambda,min}$. This turns out to be a non-trivial advantage, as the non-convex penalties suffer from the non-uniqueness issue when the degree of non-convexity of the penalty function dominates the degree of convexity of the loss function.

It is worth stressing that the choice of the standard Normal distribution is not due to some assumption about the distribution of coefficients but only to the penalty shape.

The Fan and Li (2001) conditions discussed in Section 1.3 are fulfilled. In particular, the absolute value of the parameter ensures the singularity at the origin so that the penalty can perform the variable selection as in Table 1.1; moreover the non-convexity, obtained at small ν, ensures unbiasedness of the non-zero estimates as in SCAD and MCP. However, with respect to SCAD and MCP, the proposed ANP (2.1) offers some additional advantages:

1. it is multiplicative respect to λ, namely $p_\lambda(\cdot) = \lambda\, p(\cdot)$ making the penalty shape independent of λ unlike SCAD and MCP penalties whose shape itself does depend on λ (and also on the additional γ, see Table 1);

2. is infinitely differentiable throughout the domain, except one point of discontinuity in zero; in particular, the second derivative is also continuous, which leads to some advantages (e.g. on the computation of standard errors);

3. it is very flexible, i.e. the amount of non-convexity can be easily tuned by the additional parameter ν.

In fact, it is easy to see that as $\nu \to \infty$ the first derivative of the proposed penalty (2.1) approaches to the sign function

$$\lim_{\nu\to\infty} p'_\lambda(|\xi_j|) = \lambda \operatorname{sgn}(\xi_j),$$

which is exactly the LASSO. Figures 2.1 and 2.2 show the shape of the ANP penalty functions for three different ν-values, along with the first derivative. The proposed ANP, as ν varies, moves between two limiting cases: convex, i.e. LASSO, and a non-convex penalty. The penalty takes the form and characteristics of a non-convex penalty for intermediate values, and it exhibit some of the optimal features of SCAD and MCP penalties (nearly unbiased) while keeping the LASSO appealing from a computational point of view.

While the proposed penalty function can be applied to different frameworks wherein sparseness and unbiasedness are requested, we will discuss its application in GLM's, grouped variables' and GGM's.

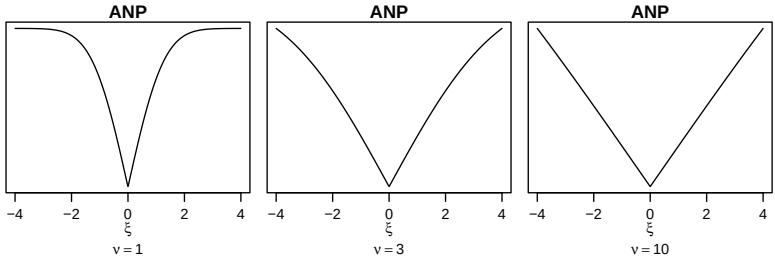


Figure 2.1: The shape of the ANP for different values of ν.

2.2 Generalized Linear Model framework

We assume that the ith observation of the outcome of interest, denoted by y_i, is drawn from a distribution belonging to the family of exponential dispersion models (Jørgensen, 1987), meaning that the probability distribution function takes the following form:

$$f(y_i; \vartheta_i, \phi) = \exp\left\{ \frac{y_i \vartheta_i - b(\vartheta_i)}{a(\phi)} + c(y_i, \phi) \right\},$$

where $a(\cdot)$ and $c(\cdot, \cdot)$ are specific functions, whereas $b(\cdot)$ denotes the cumulant generating function. In the jargon of exponential dispersion models, $\vartheta \in \mathbb{R}$ and $\phi \in \mathbb{R}^+$ are called canonical and dispersion parameter, respectively, and are related to the expected value and variance of Y_i by the following identities:

$$E(Y_i) = b'(\vartheta_i) = \mu(\vartheta_i), \qquad V(Y_i) = a(\phi) b''(\vartheta_i) = a(\phi) V(\vartheta_i),$$

where $\mu(\cdot)$ and $V(\cdot)$ are called mean and variance function, respectively. GLMs postulate that a J-dimensional vector of covariates $x_i = (x_{i1}, \ldots, x_{ip})^\top$, affects the expected value of the response Y_i by the known link function $g(\cdot)$,

$$g\{E(Y_i)\} = x_i^\top \beta = \eta_i,$$

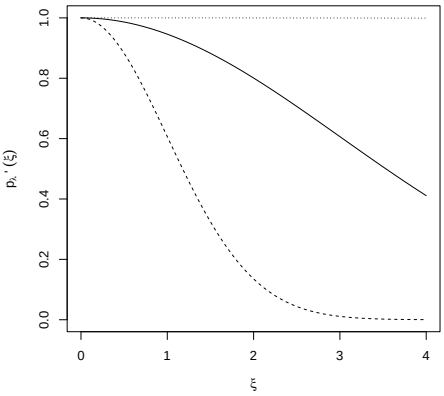


Figure 2.2: Derivative of ANP using ν 1 (dashed line), 3 (continuous line) and 10 (dotted line).

where β is the vector of regression coefficients and η_i is the linear predictor.

Using the notation introduced in Fan and Li (2001), the penalized log-likelihood takes the form (Fan and Li, 2001)

$$\mathcal{L}_\lambda(\beta) = \ell(\beta) - \sum_{j=1}^{J} p_\lambda(|\beta_j|),$$

where $\ell(\beta)$ denotes the usual log-likelihood and $p_\lambda(|\beta_j|)$ the penalty function. In GLM framework, a lot of algorithms have been proposed for estimating models with LASSO penalty function: an example are the *Least Angle Regression* (LARS) algorithm (Efron et al., 2004) and the *Differential Geometric Least Angle Regression* (dgLARS) (Augugliaro et al., 2013), providing the whole solution path as a function of the tuning parameter λ. Another popular algorithm is the *Coordinate Descent* (Friedman et al., 2010b), which continuously

updates coefficients cyclically and separately, maintaining the other coefficients as constants.

Various optimisation algorithms have also been proposed for non-convex penalty functions: approximations such as Local Quadratic Approximation (LQA) (Fan and Li, 2001) and Local Linear Approximation (LLA) (Zou and Li, 2008) have been proposed; we will discuss these deeply in Chapter 3. Other contributions include the PLUS algorithm by Zhang (2010a), different version of Coordinate Descent (Mazumder et al., 2011), and some algorithms which relies on coordinate descent algorithms (Breheny and Huang, 2011).

In this framework, we can write our proposal as

$$p_\lambda(|\beta_j|) = \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|\beta_j|}{\nu}\right).$$

Moreover, the standard errors for the estimated parameters can be easily derived as we simultaneously estimate parameters and select variables. In accordance with established practices in likelihood-based modeling, we can employ the sandwich formula as an estimator for the covariance of the estimates of $\hat{\beta}_{\hat{\mathcal{A}}}$. That is

$$\begin{aligned}\hat{\text{cov}}(\hat{\beta}_{\hat{\mathcal{A}}}) = \{\nabla^2\ell(\hat{\beta}_{\hat{\mathcal{A}}}) + \Sigma_\lambda(\hat{\beta}_{\hat{\mathcal{A}}})\}^{-1}\hat{\text{cov}}(\ell(\hat{\beta}_{\hat{\mathcal{A}}}))\\ \times\{\nabla^2\ell(\hat{\beta}_{\hat{\mathcal{A}}}) + \Sigma_\lambda(\hat{\beta}_{\hat{\mathcal{A}}})\},\end{aligned}$$

where $\Sigma_\lambda(\hat{\beta}_{\hat{\mathcal{A}}})$ is a diagonal matrix with diagonal elements

$$\lambda\exp\{-\frac{\hat{\beta}^2_{\hat{\mathcal{A}}}}{2\nu^2}\}\frac{\hat{\beta}_{\hat{\mathcal{A}}}}{\nu^2}.$$

This formula exhibits reliable accuracy, especially when dealing with moderately sized samples.

2.2.1 Grouped variable

The formalization so far assumes that each variable is associated with only one coefficient. However, the explanatory variables, such

as gene expression research or basis spline, may sometimes be organised into groups. When a "grouped" structure exists in the covariates, variable selection should account for that. Here, the challenge is to select relevant groups of variables that collectively contribute to explaining the response variable. A model may also include grouping to use previously acquired information with scientific value.

Following Huang et al. (2012), we consider a common J-factor regression problem:

$$Y = \sum_{j=1}^{J} X_j \beta_j + \epsilon,$$

where Y is an $n \times 1$ vector, X_j is an $n \times p_j$ matrix corresponding to the j-th predictor group, β_j is the p_j-dimensional coefficient vector, $\epsilon \sim N(0, \sigma I_n)$, and p_j denotes the number of factors within the j-th covariate group. Here, the response variable is centred, and each X_j is orthonormalized using Gram-Schmidt orthonormalization, equivalent to group-level standardization.

In this context, selecting important variables translates into identifying essential groups of variables or factors. Traditional variable selection methods are designed primarily for individual variable selection and may not be suitable for group-factor selection. In addition, these methods may select more factors than necessary, and their solutions may depend on how the elements are represented. Indeed, Huang and Zhang (2010) introduced the concept of strong group sparsity to evaluate the signal recovery performance of the group LASSO. They found that group LASSO outperforms standard LASSO when the signal is strongly group-sparse due to the stability associated with a group structure and requires a smaller sample size.

Let us define the generic penalized objective function for grouped variables to be optimised

$$\mathcal{L}_\lambda(\beta) = \ell(\beta) - \sum_{g=1}^{G} p(\|\beta_g\|; c_g\lambda, \gamma). \tag{2.2}$$

Here, $p(\cdot)$ represents a versatile penalty function, and its characteristics directly influence the properties of the estimator. The parameter γ is an optional parameter that allows adjustments to the penalty function but is not always provided.

Notably, in the generic objective function expressed in (2.2), when $p_g = 1$ for all G, it reduces to the well-known objective function without grouped variables. Furthermore, it's worth noting that (2.2) can be used to estimate a penalized Generalized Linear Model.

Different specifications of the penalty function lead to variations in the model and influence the properties of the estimator. For instance, Bakin et al. (1999) introduced the group LASSO along with a computational algorithm. Subsequent advancements in group selection methods and algorithms were made by Yuan and Lin (2006). The group LASSO, which incorporates the L_2 norm of coefficients associated with groups of variables into the penalty function, stands as a natural extension of the LASSO method. This extension retains computational advantages due to its convex nature, ensuring the objective function possesses a unique optimum. Several alternative penalty functions have been adapted to accommodate data with a group structure. For instance, Antoniadis (1997) delved into block-wise shrinkage techniques for regularized wavelet estimation in non-parametric regression problems. Their study explored various methods for shrinking wavelet coefficients within their natural blocks, encompassing block-wise hard and soft thresholding rules.

Wang et al. (2007b) introduced a group SCAD penalisation method for selecting variables with time-varying coefficients in functional response models. In a review article, Huang et al. (2012) extended the concept to include 2-norm group SCAD, 2-norm group MCP, and 2-norm group bridge penalties. They demonstrated that the MCP variant of this penalty function exhibits the desirable oracle property. This property implies that the estimator has the potential to

precisely identify the true underlying model, representing an ideal scenario.

Depending on the context, individual group factors may or may not possess direct scientific significance. In cases where specific factors within groups lack individual scientific importance, the primary focus often shifts toward selecting groups as a whole rather than individual factors. This concept, often called 'bi-level selection,' involves considering both significant individual variables and significant groups when individual variables are relevant. Various methods have been proposed in the literature to facilitate this dual selection process, targeting not only groups of variables but also coefficients within these groups. Two notable approaches are additive penalties (Friedman et al., 2010a) and composite penalties (Breheny and Huang, 2009).

In the case of additive penalties, authors augment the objective function with a secondary penalty function, enabling the selection of both groups and individual variables. Conversely, the concept of composite penalties presents an alternative perspective. It decomposes the penalty functions associated with groups of variables into a composite structure comprising two components: an internal penalty function (f_I), which penalizes coefficients within groups, and an external penalty function (f_O), which penalizes entire groups.

Notably, if the internal penalty function can effectively select specific coefficients, the global penalty function shrinks some coefficients within groups to zero. Conversely, the coefficients within groups are cancelled out collectively. Furthermore, researchers have derived a local coordinate descent algorithm to facilitate this approach.

To adapt our penalty function for data structured into groups, we shift the focus from individual coefficients to the L2 norm of coefficient groups. Consequently, the penalty function in equation (2.1) is redefined as:

$$p(\|\beta_g\|; c_g\lambda, \nu) = c_g\lambda\sqrt{2\pi}\nu\Phi\left(\frac{\|\beta_g\|_2}{\nu}\right). \tag{2.3}$$

This is the grouped Adaptive Non-Convex Penalty (grANP). Here, the coefficient c_g is calculated as the square root of p_g to account for variations in the sizes of grouped variables.

Following the framework introduced by Breheny and Huang (2009), the outer penalty f_O corresponds to the Adaptive Non-Convex Penalty, while the inner penalty f_I takes the form of a ridge penalty. Moreover, this approach allows for considering distinct ν_g values, one for each of the G different groups.

Indeed, the penalty function (2.3) does not shrink individual coefficients towards zero. However, if a bi-level selection approach is deemed advantageous, one can explore the use of the Adaptive Non-Convex Penalty as the inner penalty f_I, defined as:

$$p(\|\beta_j\|; c_j\lambda, \nu) = p_{\lambda,\nu_O}\left(\sum_{k=1}^{K_j} p_{\lambda,\nu_I}(|\beta_k|)\right).$$

Here, the function p represents the ANP, and the additional parameter ν is indexed by I and O, allowing for the use of different values in the two components. It's worth noting that in this thesis, we won't delve deeply into exploring this bi-level formulation. Once the scopes of application of the new penalty function are defined, the challenge arises of estimating the model coefficients in different cases, as well as making the appropriate selection of the additional parameter ν. For this discussion, we defer to the next chapter.

2.3 Gaussian Graphical Model framework

Let Y be a J-dimensional random variable with joint distribution $f(y)$ and $\mathcal{V} = \{1, \ldots, J\}$ is the associated set of vertices. Each element in this set corresponds to a random variable within the model. A probabilistic graphical model can be defined as the triplet $\{Y, f(y), \mathcal{G}\}$, where $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is a graph whose edge set $\mathcal{E}$ encodes the conditional dependence and independence structure among the

J random variables. That is, Y_h and Y_k are stochastically independent given the other random variables if $(h,k) \notin \mathcal{E}$. The interested reader may refer to Lauritzen (1996) for a comprehensive treatment of probabilistic graphical models.

The Gaussian Graphical Model is grounded on the assumption that $Y \sim N_p(\mu, \Sigma)$, i.e.

$$f(y;\mu,\Sigma) = (2\pi)^{-J/2}|\Sigma|^{-1/2}\exp\{-1/2(y-\mu)^\top\Sigma^{-1}(y-\mu)\}, \quad (2.4)$$

where μ is the vector of expected values of Y, whereas Σ is the covariance matrix. The inverse of the covariance matrix, denoted by $\Theta = \Sigma^{-1}$, is called precision matrix and its off-diagonal elements, denoted as θ_{hk}, are the parametric tools by which density (2.4) factorizes according to $\mathcal{G}$, formally:

$$(h,k) \notin \mathcal{E} \Leftrightarrow Y_h \perp\!\!\!\perp Y_k \mid Y_{\mathcal{V}\setminus\{h,k\}} \Leftrightarrow \theta_{hk} = 0.$$

The pairwise relationships between the variables are encoded by Θ: if there is no edge between two nodes, the variables are conditionally independent, given all the other variables in the graph. GGMs are widely used in biology, finance and social sciences because they provide a flexible tool to model complex relationships between variables.

Suppose that a set of n independent and identically distributed observations is drawn from the distribution (2.4) and denote the corresponding GGM by $\{Y, f(y;\mu,\Theta), \mathcal{G}\}$. In principle, inference on the factorization of the density (2.4), and consequently on $\mathcal{G}$, can be carried out by maximizing the log-likelihood function:

$$\ell(\Theta) \propto \log\det\Theta - \mathrm{tr}(S\Theta),$$

where S denotes the empirical covariance matrix. Then, the edge-set $\mathcal{E}$ can be estimated by $\widehat{\mathcal{E}} = \{(h,k) \mid \hat{\theta}_{hk} \neq 0\}$.

Hence, the problem of estimating the edge set $\hat{\mathcal{E}}$ is equivalent to the problem of selecting the non-zero entries of Θ. There are several ways to reduce the number of parameters in a GGM. Applying

some structural constraints on the precision matrix (such as symmetries, Højsgaard and Lauritzen (2008); Ranciati et al. (2021)), using a model selection procedure based on AIC, BIC or test statistics (Dempster, 1972), or the *Neighborhood Selection* approach (Meinshausen and Bühlmann, 2006) which is one of the earliest sparse inference techniques for the undirected Gaussian graph.

The Penalized Gaussian Graphical Model (pGGM) extends the traditional GGM by introducing a penalty term to select a sparse set of edges representing the dependencies between variables. This approach allows only the most relevant edges to be selected, which helps to mitigate the effects of overfitting, improves the interpretability of the model, and makes the model scalable to large datasets.

Yuan and Lin (2007) proposed the *Graphical LASSO (gLASSO)*, which is defined by adding the LASSO penalty to the log-likelihood, namely:

$$\log\det\Theta - \mathrm{tr}(S\Theta) - \lambda\|\Theta\|_1,$$

where $\|\Theta\|_1 = \sum_{i\neq j}|\theta_{ij}|$ and λ is the tuning parameter regulating the sparseness: as usual, the higher the value, the sparser the estimate. Fan et al. (2009a) propose to use SCAD or Adaptive LASSO. Interestingly, MCP has not been proposed in this framework. Many efficient algorithms were developed to estimate the coefficients of gLASSO. Among this, Friedman et al. (2008) suggest transforming the objective function using the dual and optimising the function using the coordinate block descent method, optimizing one row and one matrix column at a time. The dual problem is equivalent to optimising a quadratic loss function with a LASSO penalty, so it is possible to use all the well-known algorithms proposed in the literature to estimate each coefficient (Banerjee et al., 2008). In this framework, we write our proposal as

$$p_\lambda(|\theta_{hk}|) = \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|\theta_{hk}|}{\nu}\right).$$

Chapter 3

Model estimation

To estimate the penalized models (in their three different variations), we propose using a version of the alternating directions method of multipliers (ADMM) algorithm Boyd et al. (2011). In the algorithm's framework, a linear approximation will simplify certain steps, known as the Local Linear Approximation (LLA) (Zou and Li, 2008). Before delving into the algorithm's specifics, we will provide an overview of how ADMM and LLA operate.

3.1 The ADMM algorithm

Optimization in mathematics refers to searching for optimal parameters of a - usually complex - system. Optimization problems are found in many scientific fields, such as physics, chemistry, economics and statistics.

In mathematics, an optimization problem is formulated as a problem of minimizing or maximizing a function of one or more variables. While in the minimization (or maximization) of single-variable functions, analytical and algebraic methods can be used to define minima (or maxima) precisely, in the study of multi-variable

functions, mainly numerical methods are used for an approximate definition of minima (or maxima). Several optimisation problems impose additional constraints that the solutions must satisfy: these can be equality or inequality.

In our context, the minimization problem can be rewritten as an equality-constrained convex optimization minimization problem. Among the different algorithms proposed in the literature, the one we choose to use is the Alternating Direction Method of Multiplier (ADMM) (Gabay, 1983), which is a simple but powerful algorithm for convex and large-scale problems (as we shall see, it allows us to treat J different problems in parallel and independently). The method was introduced in the mid-1970s and studied throughout the 1980s. It is based on combining two different algorithms (Dual Ascent and Method of Multiplier). Before illustrating the ADMM, the Dual Ascent and Multiplier Method will be briefly reviewed.

3.1.1 Dual Ascent and Dual Decomposition

Consider the following problem

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{3.1}$$

where $x \in R^n$, $A \in R^{m \times n}$ and $f : R^n \to R$ is convex. The Lagrangian problem is

$$L(x, y) = f(x) + y^T(Ax - b),$$

while the dual function is

$$g(y) = \inf_x L(x, y) = -f^*(-A^T y) - b^T y.$$

y is the dual variable, and f^* is the convex conjugate function of f. Then, the problem is to maximize $g(y)$, which is equal to minimize (3.1). After that we maximize $g(y)$ finding the dual optimal point y^*, we can find the optimal point x^* as

$$x^* = \arg\min_x L(x, y^*).$$

If $f(x, y^*)$ is strictly convex, only one minimizer of $L(x, y^*)$ exists. The dual ascent method consists of iterating the updates.

$$x^{k+1} = \arg\min_x L(x, y^k) \tag{3.2}$$

$$y^{k+1} = y^k + \alpha^k(Ax^{k+1} - b). \tag{3.3}$$

The first step is an x-minimization step; the second is the update id dual variable. The term α is a step size, and if it is correctly chosen, the dual function increases in each step. As stated by Boyd et al. (2011), the conditions under which the iteration is guaranteed to converge to the optimal solution are quite conservative.

One of the greatest strengths of the Dual Ascent is that it can lead to parallelising the algorithm if f is separable, or

$$f(x) = \sum_{i=1}^{N} f_i(x_i),$$

where $x = (x_1, ..., x_N)$ and $x_i \in R^{n_i}$ are subvectors of x. Partitioning the matrix A, the Lagrangian can be rewritten as

$$L(x, y) = \sum_{i=1}^{N} L_i(x_i, y) = \sum_{i=1}^{N}(f_i(x_i) + y^T A_i x_i - \frac{1}{N} y^T b).$$

So, the x-minimization step (3.2) can be treated in parallel and independently, allowing the algorithm's efficiency to be greatly increased. So, the new iterating updates are

$$x_i^{k+1} = \arg\min_{x_i} L_i(x_i, y^k) \tag{3.4}$$

$$y^{k+1} = y^k + \alpha^k(Ax^{k+1} - b). \tag{3.5}$$

In this case, the name of the algorithm becomes Dual Decomposition. From a computational point of view, the Dual decomposition is structured in a phase of collecting the results for the calculation of the residuals $A_i x_i^{k+1}$ at step (3.5), which are combined and then redistributed for the N optimisations conducted in parallel at step (3.4).

3.1.2 Augmented Lagrangians and the Method of Multipliers

Augmented Lagrangian methods (Hestenes, 1969; Powell, 1969) are a class of algorithms used to solve optimization problems with constraints. They transform the original problem into an unconstrained optimization problem by transforming the constraints into penalty terms of the objective function. They add a term to the objective function to mimic the Lagrangian multiplier, which is different from the Lagrangian multiplier. The unconstrained objective function is the Lagrangian dual of the constrained problem with an additional penalty term.

it brings robustness to the dual ascent method to yield convergence without assuming the strict convexity of f. The augmented Lagrangian for (3.1) is

$$L_\rho(x, y) = f(x) + y^T(Ax - b) + \frac{\rho}{2}\|Ax - b\|_2^2. \tag{3.6}$$

ρ is the penalty parameter. We can see that, for $\rho = 0$, the augmented Lagrangian becomes the standard Lagrangian problem. Note that the added penalty term is equal to 0. This can be viewed as a Lagrangian problem

$$\begin{aligned} \min \quad & f(x) + \frac{\rho}{2}\|Ax - b\|_2^2 \\ \text{s.t.} \quad & Ax = b. \end{aligned}$$

Applying the dual ascent method to the problem, the algorithm is

$$x^{k+1} = \arg\min_x L_\rho(x, y^k) \tag{3.7}$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} - b). \tag{3.8}$$

which is called Method of Multipliers. This is very similar to the Dual Ascent, except for (3.7), where a penalty term is involved on Lagrangian. ρ is the step size (similar to α in (3.3)).

The improvement in terms of convergence comes at some costs. In fact, when f is separable, L_ρ is not separable, so the x-minimization step cannot be parallelized for each x_i.

3.1.3 Alternating Direction Method of Multipliers

The Alternating Direction Method of Multipliers (ADMM) is an algorithm that combines the decomposition property of Dual Ascent and the best convergence properties of the multiplier method. The idea is to split an optimization problem into two parts, separable across the splitting. The algorithm solves the problem

$$\begin{aligned} \min \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c, \end{aligned}$$

where $x \in R^n$, $z \in R^m$, $A \in R^{J\times n}$, $B \in R^{J\times m}$ and $c \in R^J$. As done in the Method of Multipliers, we form the augmented Lagrangian

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2.$$

The optimal points of the problem are denoted by

$$p^* = \inf\{f(x) + g(z) \,|\, Ax + Bz = c\}. \tag{3.9}$$

The steps of the ADMM algorithm are as follows

$$x^{k+1} = \arg\min_{x} L_\rho(x, z^k, y^k) \tag{3.10}$$

$$z^{k+1} = \arg\min_{z} L_\rho(x^{k+1}, z, y^k) \tag{3.11}$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} + Bz^{k+1} - c), \tag{3.12}$$

The role of $\rho > 0$ is the same of (3.6). The algorithm consists of a x-minimization step (3.10), a z-minimization step (3.11) and a dual variable update (3.12). In this algorithm, the variables x and z are updated in an alternating way; for this reason, the algorithm is called *Alternating Direction*. When f and g are separable, splitting x and z allows for decomposition.

One of the problems of ADMM is the convergence speed to high accuracy. It is often the case that ADMM converges with a modest accuracy in a few steps, but it requires a large number of iterations for high accuracy. On the other hand, one of the significant benefits of ADMM is its convergence property, inherited from the Method of Multipliers. Defining the residual r of k-iteration as

$$r^k = Ax^k + Bz^k - c,$$

it is possible to define a stopping criterion based on residuals. It can be considered that the algorithm has reached convergence when

$$\|r^k\|_2 < \epsilon,$$

where ϵ is the tolerance value. One of the possible extensions of the ADMM algorithm concerns the parameter ρ: it is possible to establish an updating scheme along with the iterations of the algorithm, with the benefit of a higher convergence speed and lower dependence on the chosen parameter ρ.

3.2 Quadratic and Local Linear Approximation

In their work introducing the SCAD penalty, Fan and Li (2001), to estimate the model suggested iteratively, propose to approximate the penalty function by a quadratic function locally and referred to such approximation as Local Quadratic Approximation (LQA). Consider the following penalized likelihood function

$$\mathcal{Q}(\xi) = \sum_{i=1}^{n} \ell_i(\xi) - n \sum_{j=1}^{J} p_{\lambda_j}(|\xi_j|). \tag{3.13}$$

Fan and Li (2001) propose to approximate the penalty function as

$$p_{\lambda_j}(|\xi_j|) \approx p_{\lambda_j}(|\xi_j^{(0)}|) + \frac{1}{2}\{p'_{\lambda_j}(|\xi_j^{(0)}|)/|\xi_j^{(0)}|\}(\xi_j^2 - \xi_j^{(0)2}), \tag{3.14}$$

where $\xi_j^{(0)}$ is an initial value close to the true value of ξ_j. Substituting (3.14) in (3.13) and setting the initial value $\xi_j^{(0)}$ to the un-penalized maximum likelihood estimate, the maximization of the objective function is solved repeating

$$\hat{\xi}^{k+1} = \arg\max_{\xi} \left\{ \sum_{i=1}^{n} \ell_i(\xi) - n \sum_{j=1}^{p} \frac{p'_{\lambda_j}(|\xi_j^{(k)}|)}{2|\xi_j^{(k)}|} \xi_j^2 \right\}. \tag{3.15}$$

Applying the LQA to the likelihood function, the penalized likelihood equation (3.15) transforms into a problem that can be solved analytically using a closed-form solution. The algorithm is stopped when $\hat{\xi}^{(k)}$ converges. The problem of LQA is that it cannot have a sparse representation: Fan and Li (2001) suggests to set a generic $\hat{\xi}_j = 0$ if $\hat{\xi}_j$ is very close to 0, say $|\hat{\xi}_j| < \epsilon_0$. Hunter and Li (2005) studied the convergence property of the LQA: their findings demonstrate

that it performs a similar role to the E-step in the Expectation-Maximization (EM).

To eliminate the instability of LQA, Zou and Li (2008) propose a Local Linear Approximation (LLA), which guarantees a natural sparse representation. They propose to approximate the penalty function as

$$p_{\lambda_j}(|\xi_j|) \approx p_{\lambda_j}(|\xi_j^{(0)}|) + p'_{\lambda_j}(|\xi_j^{(0)}|)(|\xi_j| - |\xi_j^{(0)}|). \tag{3.16}$$

As shown in Figure 3.1, both LLA and LQA perform as convex majorants for the concave penalty function $p_{\lambda_j}(|\xi_j|)$. However, LLA is a superior approximation since it represents the minimum (tightest) convex majorant for the concave function.

Substituting (3.16) in (3.13) and setting the initial value $\xi^{(0)}$ to the un-penalized maximum likelihood estimate, the maximization of the new objective function is solved repeating

$$\hat{\xi}^{k+1} = \arg\max_{\xi} \left\{ \sum_{i=1}^{n} \ell_i(\xi) - n \sum_{j=1}^{J} p'_{\lambda_j}(|\xi_j^{(k)}|)|\xi_j| \right\}. \tag{3.17}$$

The algorithm is stopped when $\hat{\xi}^{(k)}$ converges. This problem is a concave optimization problem if the likelihood function is concave. So, in this form, the function to be optimized enjoys all the computational characteristics of a (weighted) LASSO, and can be solved with the best-known algorithms in the literature, such as LARS (Efron et al., 2004). (Zou and Li, 2008) demonstrate that, for a concave penalty function p_λ on $[0, \infty)$, the LLA has the ascent property, so

$$\mathcal{Q}(\xi^{(k+1)}) \geq \mathcal{Q}(\xi^{(k)}), \tag{3.18}$$

or

$$\mathcal{Q}(\xi^{(k+1)}) > \mathcal{Q}(\xi^{(k)}), \tag{3.19}$$

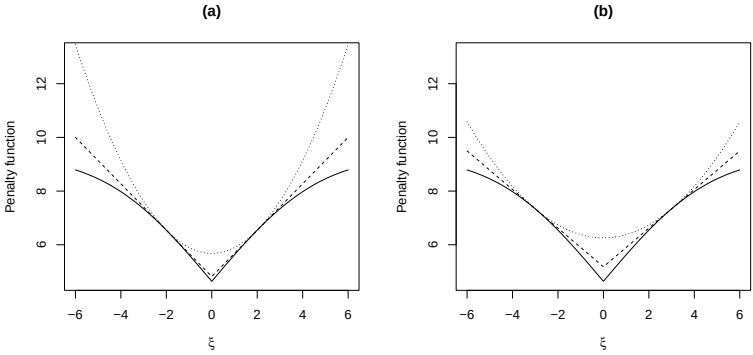


Figure 3.1: Local Quadratic Approximation (thin dotted lines) and Local Linear Approximation (thin broken line) at $\xi = 2$ (a) and 3 (b). ν fixed to 1.

if the penalty function is strictly concave.

However, carious penalty functions yield different weighting schemes: the LASSO utilizes a constant weighting scheme, while the nonconcave penalized likelihood can be seen as an iterative reweighted penalized problem, where the weights are adjusted at each iteration based on the specific penalty function employed (Fan and Lv, 2010).

In Figure 3.1, the ANP (continuous line) for ν fixed at 1, LQA (dotted line) and LLA (broken line) are reported. From a graphical point of view, it is clear that the Local Quadratic Approximation is not capable of shrinking some coefficient to 0 due to the absence of singularity at the origin; on the other hand, the Local Linear Approximation maintain the singularity at 0, so it estimates naturally some coefficients equal to 0.

3.3 Generalized Linear Model

Using a local quadratic approximation of the log-likelihood function, it is well known that parameter estimation is carried out iteratively via the IWLS (see for example McCullagh and Nelder (1989)), namely via minimization of the following

$$\sum_{i=1}^{n} V_i^t (z_i^t - x_i^\top \beta)^2 + \lambda\sqrt{2\pi}\nu \sum_{j=1}^{J} \Phi\left(\frac{|\beta_j|}{\nu}\right),$$

where $V_i^t = V(\eta_i^t)$ is the weight matrix, $z_i^t = \eta_i^t + \{y_i - \hat{\mu}(\eta_i^t)\}/V_i^t$ the working response vector and η_i the linear predictor. The superscript emphasizes the dependence on the previous value $\beta^{(t-1)}$. To apply the ADMM we first set

$$f(\beta) = \sum_{i=1}^{n} V_i^t (z_i^t - x_i^\top \beta)^2,$$

$$g(\tilde{\beta}) = -\lambda\sqrt{2\pi}\nu \sum_{j=1}^{J} \Phi\left(\frac{|\tilde{\beta}_j|}{\nu}\right),$$

then parameter estimation is performed via the following constrained minimization problem

$$\begin{aligned} \min_{\beta,\tilde{\beta}} \quad & f(\beta) + g(\tilde{\beta}) \\ \text{s.t.} \quad & \beta - \tilde{\beta} = 0, \end{aligned}$$

According to the standard ADMM theory, the augmented scaled Lagrangian function takes the form:

$$\mathcal{Q}_\tau(\beta, \tilde{\beta}, \gamma) = f(\beta) + g(\tilde{\beta}) + \frac{\tau}{2}\|\beta - \tilde{\beta} + \gamma\|_2^2,$$

where γ is a dual variable whereas τ is a non-negative penalty parameter that controls the algorithm's convergence rate. The ADMM algorithm consists in repeating the following three steps until a convergence criterion is met:

1: $\beta^{k+1} = \arg\min_{\beta} \mathcal{Q}_{\tau}(\beta, \tilde{\beta}^k, \gamma^k)$
2: $\tilde{\beta}^{k+1} = \arg\min_{\tilde{\beta}} \mathcal{Q}_{\tau}(\beta^{k+1}, \tilde{\beta}, \gamma^k)$
3: $\gamma^{k+1} = \beta^{k+1} - \tilde{\beta}^{k+1} + \gamma^k$.

Updating β. Step 1 involves the optimization of the augmented scaled Lagrangian function with respect to β, thus it is equivalent to the following problem:

$$\beta^{k+1} = \min_{\beta} \frac{1}{n} \sum_{i=1}^{n} V_i^t (y_i^t - x_i^{\top}\beta)^2 + \frac{\tau}{2} \|\beta - \tilde{\beta}^k + \gamma^k\|_2^2,$$

which admits solution in closed form:

$$\beta^{k+1} = \left(\frac{1}{n} X^{\top} V^t X + \tau I\right)^{-1} \left\{\frac{1}{n} X^{\top} V^t y^t + \tau(\tilde{\beta}^k - \gamma^k)\right\},$$

where $V^t = \text{diag}(V_1^t, \ldots, V_n^t)$.

Updating $\tilde{\beta}$. Given β^{k+1} and letting $\mathring{\beta}_j^k = \beta_j^{k+1} + \gamma_j^k$, the minimization problem in Step 2 can be written as follows:

$$\tilde{\beta}^k = \min_{\tilde{\beta}} \sum_{j=1}^{J} \left\{ \frac{\tau}{2} (\mathring{\beta}_j^k - \tilde{\beta}_j)^2 - \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|\tilde{\beta}_j|}{\nu}\right)\right\}, \tag{3.20}$$

which shows that the objective function in Step 2 is additive, implying that the optimization problem can be solved in parallel. Given the non-linear structure of the proposed penalty function, to solve the jth sub-problem, we propose to use the Local Linear Approximation (LLA) proposed by Zou (2006). So, the second Step is computed as the solution of a sequence of new minimization problems involving a new objective function that replaces the penalty function with a suitable local approximation. Formally, $\tilde{\beta}$ is obtained by the following iterative procedure:

1: Let $\tilde{\beta}_t$ be a starting value

2: **repeat**
3: Let $w_t = \exp\{-(\tilde{\beta}_t/\nu)^2/2\}$
4: $\tilde{\beta}_{t+1} = \arg\min_{\tilde{\beta}} \frac{\tau}{2}(\mathring{\beta}^k - \tilde{\beta})^2 + \lambda w_t|\tilde{\beta}|$
5: **until** convergence criterion is met
6: Return $\tilde{\beta}_{k+1} = \tilde{\beta}_{t+1}$

It is easy to recognize that Step 4 is a weighted LASSO problem; therefore, using the results given in Friedman et al. (2007), we have that the updating step of $\tilde{\beta}^{k+1}$ admits the following solution in closed form, i.e., $\tilde{\beta}^{k+1} = S(\mathring{\beta}^k; w_k\lambda/\tau)$, where $S(x;\lambda) = \text{sign}(x)(|x|-\lambda)_+$ is the soft-thresholding operator. We will refer to the support identified by the estimated coefficients as $\hat{\mathcal{A}} = \{j : \hat{\beta}_j \neq 0\}$.

Regarding step 5 (convergence criterion), for each value of the parameter λ, two convergence criteria are evaluated based on two quantities:

$$R_1 = \frac{1}{J}\sqrt{(\tilde{\beta}_t - \tilde{\beta}_{t+1})^2}$$

$$R_2 = \frac{1}{J}\sqrt{(\beta_{t+1} - \tilde{\beta}_t)^2}$$

The algorithm stops if both quantities are below a convergence threshold of ϵ.

3.4 Grouped variable

Our approach extends the optimization process to accommodate varying values of ν_j. Again, we consider the standard local quadratic approximation of the log-likelihood function, following the principles outlined by McCullagh and Nelder (1989). Specifically, let $\hat{\beta}^t$ represent an appropriate initial point, and define $\eta_i^t = x_i^\top \hat{\beta}^t$. With this, we can approximate the minimizer of the objective function (2.2), incorporating the grouped ANP, as follows:

$$\hat{\beta}^{t+1} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} V_i^t (y_i^t - x_i^\top \beta)^2 - c_j \lambda \sqrt{2\pi} \nu \sum_{j=1}^{J} \Phi \left(\frac{\|\beta_j\|_2}{\nu} \right),$$

where $V_i^t = V(\eta_i^t)$ and $y_i^t = \eta_i^t + \{y_i - \mu(\eta_i^t)\}/V_i^t$ denotes the ith working response of the IWLS algorithm. The above approximation shows that estimating $\hat{\beta}$ involves solving a series of penalised weighted least squares regression problems, which can be efficiently solved using the ADMM algorithm. For a comprehensive discussion of ADMM algorithms, we refer the interested reader to Boyd et al. (2011). We begin defining $\hat{\beta}^{t+1}$ as solution of the following linear equality-constrained problem:

$$\begin{aligned} \min_{\beta, \tilde{\beta}} \quad & f(\beta) + g(\tilde{\beta}) \\ \text{s.t.} \quad & \beta - \tilde{\beta} = 0, \end{aligned}$$

where:

$$f(\beta) = \frac{1}{n} \sum_{i=1}^{n} V_i^t (y_i^t - x_i^\top \beta)^2, \quad \text{and} \quad g(\tilde{\beta}) = -c_j \lambda \sqrt{2\pi} \nu \sum_{j=1}^{J} \Phi \left(\frac{\left\| \tilde{\beta}_j \right\|_2}{\nu} \right).$$

The augmented scaled Lagrangian function is:

$$\mathcal{Q}_\tau(\beta, \tilde{\beta}, \gamma) = f(\beta) + g(\tilde{\beta}) + \frac{\tau}{2} \|\beta - \tilde{\beta} + \gamma\|_2^2.$$

γ is the dual variable, and τ is a non-negative penalty parameter that governs the rate of convergence of the algorithm. The ADMM algorithm involves the following three steps, which are repeated until a convergence criterion is satisfied:

1: $\beta^{k+1} = \arg\min_\beta \mathcal{Q}_\tau(\beta, \tilde{\beta}^k, \gamma^k)$
2: $\tilde{\beta}^{k+1} = \arg\min_{\tilde{\beta}} \mathcal{Q}_\tau(\beta^{k+1}, \tilde{\beta}, \gamma^k)$
3: $\gamma^{k+1} = \beta^{k+1} - \tilde{\beta}^{k+1} + \gamma^k$.

Update for β: step 1 involves optimizing the augmented scaled Lagrangian function with respect to β. Hence, it is equivalent to solving the following problem:

$$\beta^{k+1} = \min_{\beta} \frac{1}{n} \sum_{i=1}^{n} V_i^t (y_i^t - x_i^\top \beta)^2 + \frac{\tau}{2} \|\beta - \tilde{\beta}^k + \gamma^k\|_2^2,$$

which admits a closed-form solution:

$$\beta^{k+1} = \left(\frac{1}{n} X^\top V^t X + \tau I \right)^{-1} \left\{ \frac{1}{n} X^\top V^t y^t + \tau (\tilde{\beta}^k - \gamma^k) \right\},$$

where $V^t = \text{diag}(V_1^t, \ldots, V_n^t)$.

Update for $\tilde{\beta}$: having β^{k+1}, we update $\tilde{\beta}$ in Step 2 by setting $\mathring{\beta}_j^k = \beta_j^{k+1} + \gamma_j^k$ and then solving the following minimization problem:

$$\tilde{\beta}^k = \min_{\tilde{\beta}} \sum_{j=1}^{J} \left\{ \frac{\tau}{2} (\mathring{\beta}_j^k - \tilde{\beta}_j)^2 - c_j \lambda \sqrt{2\pi} \nu \Phi \left(\frac{\left\| \tilde{\beta}_j \right\|_2}{\nu} \right) \right\}. \quad (3.21)$$

The objective function in Step 2 can be broken down into smaller parts, meaning the optimization problem can be solved in parallel for each group. However, solving these sub-problems separately can be challenging because the proposed penalty function is non-linear. To overcome this, we suggest using the Local Linear Approximation (LLA) method proposed by Zou (2006) to solve the j-th sub-problem. Thus, Step 2 solves a new sequence of minimization problems by replacing the penalty function with an appropriate local approximation. This procedure is iterative and results in obtaining $\tilde{\beta}$ by:

1: Let $\tilde{\beta}_t$ be a starting value
2: **repeat**
3: Let $w_t = c_j \exp\left\{ - \left(\left\| \tilde{\beta}_j \right\|_2 / \nu \right)^2 / 2 \right\}$

4: $\tilde{\beta}_{t+1} = \arg\min_{\tilde{\beta}} \frac{\tau}{2}(\mathring{\beta}^k - \tilde{\beta})^2 + \lambda w_t |\tilde{\beta}|$
5: **until** convergence criterion is met
6: Return $\tilde{\beta}_{k+1} = \tilde{\beta}_{t+1}$

Step 4 involves a weighted LASSO problem. We can use the findings shown in Boyd et al. (2011) to compute the solution as a closed-form for updating $\tilde{\beta}^{k+1}$. Specifically, the update step for $\tilde{\beta}^{k+1}$ can be obtained using $S(\mathring{\beta}^k; w_k\lambda/\tau)$, where $S_z(a) = (1 - z/\|a\|_2)_{+a} : \mathcal{R}^m \to \mathcal{R}^m$ is the vector soft thresholding operator. When a is a scalar, the formula simplifies to its scalar form. Furthermore, this expression extends the formula provided by Friedman et al. (2007).

3.5 Gaussian Graphical Model

We start by redefining the solution of our minimization problem as the solution to the following equality-constrained minimization problem, with matrix variables Θ and Z:

$$\begin{aligned} \min_{\Theta, Z \succ 0} \quad & -\ell(\Theta) + \lambda \sum_{h,k=1} \sqrt{2\pi}\nu\Phi\left(\frac{|Z_{hk}|}{\nu_{hk}}\right) \\ \text{s.t.} \quad & \Theta - Z = 0. \end{aligned}$$

The augmented scaled Lagrangian function takes the form:

$$\begin{aligned} \mathcal{Q}(\Theta, Z, U) = -\ell(\Theta) + \lambda \sum_{h,k=1} \sqrt{2\pi}\nu\Phi\left(\frac{|Z_{hk}|}{\nu_{hk}}\right) \\ + \frac{\tau}{2}\|\Omega - Z + U\|_F^2 - \frac{\tau}{2}\|U\|_F^2, \end{aligned}$$

where $\tau > 0$ is a penalty parameter, $U \succ 0$ is the scaled dual matrix and $\|\cdot\|_F$ denotes the Frobenius norm, respectively. The solution of the problem can be computed by the following procedure:

1: **repeat**
2: $\Theta^{k+1} = \arg\min_{\Theta \succ 0} -\ell(\Theta) + \frac{\tau}{2}\|\Theta - Z^k + U^k\|_F^2,$

3: $Z^{k+1} = \arg\min_{Z \succ 0} \frac{\tau}{2}\|\Theta^{k+1} - Z + U^k\|_F^2 + \lambda \sum_{h,k}^{p} \sqrt{2\pi}\nu\Phi\left(\frac{|Z_{hk}|}{\nu_{hk}}\right)$,

4: $U^{k+1} = U^k + \Theta^{k+1} - Z^{k+1}$

5: **until** convergence criterion is met

Updating Θ. The problem in Step 2 has been studied in Boyd et al. (2011), where the authors show that updating the precision matrix estimator admits a closed-form solution. To gain more insight into the updating formula, consider the first-order optimality condition of the problem in Step 2, which can be rewritten in the following more convenient form:

$$\tau\Theta - \Theta^{-1} = \tau(Z^k - U^k) - S. \tag{3.22}$$

Let $Q\Lambda Q^\top$ be the spectral decomposition of $\tau(Z^k - U^k) - S$, then from the equation (3.22) we can immediately conclude that Θ^{k+1} can be written as $Q\widetilde{\Lambda}Q^\top$, where $\widetilde{\Lambda}$ is a diagonal matrix whose elements are the solutions of the equation $\tau\widetilde{\Lambda} - \widetilde{\Lambda}^{-1} - \Lambda = 0$, i.e.:

$$\tilde{\lambda}_{ii} = \frac{\lambda_{ii} + \sqrt{\lambda_{ii}^2 + 4\tau}}{2\tau},$$

which is always positive because $\tau > 0$.

Updating Z. Before going into the technical details of updating the matrix Z, note that the objective function in Step 3 has the following additive structure:

$$\frac{\tau}{2}\sum_{h,k=1}^{J}\left\{(\theta_{hk}^{k+1} + U_{hk}^k - Z_{hk})^2 + \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|Z_{hk}|}{\nu_{hk}}\right)\right\},$$

which implies that the minimization problem in Step 3 can be decomposed into $J(J+1)/2$ univariate optimization problems that can be solved in parallel. Therefore, the rest of this section will focus on how to solve the sub-problem:

$$Z_{hk}^{k+1} = \arg\min_{Z_{hk}} \frac{\tau}{2}(\theta_{hk}^{k+1} + U_{hk}^{k} - Z_{hk})^2 + \lambda\sqrt{2\pi}\nu\Phi\left(\frac{|Z_{hk}|}{\nu_{hk}}\right).$$

Even in the GGM framework, we solve the problem using the Local Linear Approximation (LLA) method (Zou and Li, 2008), i.e. Z_{hk}^{k+1} is computed as the solution to a sequence of new minimisation problems with a new objective function obtained by replacing the penalty function with a suitable local approximation. Formally, Z_{hk}^{k+1} is obtained by the following iterative procedure:

1: Let $\widetilde{Z}_{hk}^{k}$ be a starting value
2: **repeat**
3: Let $w_{hk} = \exp\{-(\widetilde{Z}_{hk}^{k}/\nu_{hk})^2/2\}$
4: $\widetilde{Z}_{hk}^{k+1} = \arg\min_{\widetilde{Z}_{hk}} \frac{1}{2}(\theta_{hk}^{k+1} + U_{hk}^{k} - \widetilde{Z}_{hk})^2 + \frac{\lambda}{\tau} w_{hk}|\widetilde{Z}_{hk}|$
5: **until** convergence criterion is met
6: Return $Z_{hk}^{k+1} = \widetilde{Z}_{hk}^{k+1}$

The Step 4 is a weighted LASSO problem; therefore, using the results given in Friedman et al. (2007), we have that the updating step of $\widetilde{Z}_{hk}^{k+1}$ admits the following closed-form solution:

$$\widetilde{Z}_{hk}^{k+1} = S(\theta_{hk}^{k+1} + U_{hk}^{k}; \frac{\lambda}{\tau} w_{hk}),$$

where $S(x;\lambda) = \text{sign}(x)(|x| - \lambda)_+$ is the soft-threshold operator.

After describing the algorithms for coefficient estimation, the next chapter will evaluate the performance of the proposed penalty function against the leading competitors known in the literature.

3.6 About the choice of ν

The role of ν is crucial in the proposed approach. While sparseness is preserved, ν affects both computational and inferential aspects. As

discussed in section 2, as ν increases the ANP approaches to LASSO, making the estimates biased. On the other hand, the lower ν the less biased the solution. In other words, the parameter ν determines the convergence rate of the proposed estimator to the maximum likelihood one: the lower the value, the higher the convergence rate. However, using lower values of ν does not come without drawbacks, as the objective function will have local minima. To understand how ν may affect the number of local solutions of the objective function, consider the gradient of (3.20)

$$-\tau(\mathring{\beta} - \tilde{\beta}) + \lambda \exp\left\{-\frac{\tilde{\beta}^2}{2\nu^2}\right\} \operatorname{sgn}(\tilde{\beta}). \tag{3.23}$$

From (3.23), it is easy to see how ν influence gradient, from which it is possible to guess its influence in both computational terms and estimation results.

In Figure 3.2, it is possible to see, from a graphical point of view, how ν acts on the gradient. The gradient is not monotonic for parameter values that are too small: the solutions are more than one. By increasing the value of ν, the gradient will become monotonic, and the solution will be unique. Conversely, the convergence speed decreases as the value of ν increases. Hence arises the problem of defining the lowest value of ν such that only one solution can be obtained (typical property of convex penalties), having the highest convergence rate to the maximum likelihood estimates and thus the lowest bias (property of nonconvex penalties).

Due to the decomposition obtained with the ADMM (which makes it possible to split the estimate of the non-sparse solution from the sparse one), it is possible to go into the detail of the $\tilde{\beta}$ estimate, making its diagnosis possible: in this way it is possible to study what component introduces the presence of local multiple solutions, that is, the excessive degree of the non-convexity of the penalty (as already known in the literature). By tying the degree of non-convexity to the additional parameter ν, it is possible to work on the degree of the non-convexity of the penalty, so that it does not

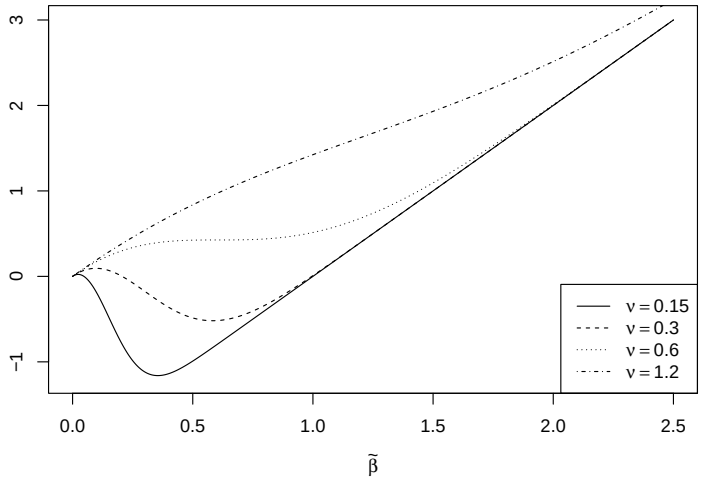


Figure 3.2: Gradient (3.23). $\lambda = 2$, $\tau = 2$, $\mathring{\beta} = 1$. The parameter ν influences the shape of the gradient, determining the number of solutions. Beyond a certain threshold, the problem will have a unique solution.

dominate over the convexity of the likelihood.

The crucial point is to find the lowest value of ν that guarantees the monotonicity of (3.23), i.e. its first derivative must always be greater than or equal to 0. Since the optimal value of ν depends on λ, we will make the dependence explicit by writing $\nu_{\lambda,min}$. Let us consider the derivative of (3.23) with respect to $\tilde{\beta}$

$$1 - \frac{\lambda}{\tau} \exp\left\{-\frac{\tilde{\beta}^2}{2\nu_\lambda^2}\right\} \frac{|\tilde{\beta}|}{\nu_\lambda^2}, \tag{3.24}$$

we want that (3.24) is greater than or equal to 0, so

$$\frac{\lambda}{\tau}\exp\left\{-\frac{\tilde{\beta}^2}{2\nu_\lambda^2}\right\}\frac{|\tilde{\beta}|}{\nu_\lambda^2} \leq 1.$$

To solve the inequality only with respect to ν_λ, we consider the maximum value the quantity can assume, which occurs when $\tilde{\beta}$ is equal to ν_λ.

$$\frac{\lambda}{\tau}\exp\left\{-\frac{\tilde{\beta}^2}{2\nu_\lambda^2}\right\}\frac{|\tilde{\beta}|}{\nu_\lambda^2} \leq \max_{\tilde{\beta}}\frac{\lambda}{\tau}\exp\left\{-\frac{\tilde{\beta}^2}{2\nu_\lambda^2}\right\}\frac{|\tilde{\beta}|}{\nu_\lambda^2} \leq 1.$$

Finding the solution for the maximum value of $\tilde{\beta}$ implies the solution for any value that $\tilde{\beta}$ can assume. The maximum value occurs when $\tilde{\beta}$ takes on a value equal to ν_λ, i.e.

$$\frac{\lambda}{\tau}\exp\left\{-\frac{1}{2}\right\}\frac{1}{\nu_\lambda} \leq 1. \tag{3.25}$$

From (3.25) is easy to obtain

$$\nu_{\lambda,min} \geq \frac{\lambda}{\tau}\exp\left\{-\frac{1}{2}\right\}.$$

This important result makes it possible to find the smallest value of λ for each value of $\nu_{\lambda,min}$ that guarantees the highest convergence rate to maximum likelihood estimates, avoiding numerical instability problems during coefficient estimation. We will refer to $\nu_{\lambda,min}$ when $\nu_{\lambda,min}$ is exactly equal to $\frac{\lambda}{\tau}\exp\left\{-\frac{1}{2}\right\}$.

In the context of grouped variables, the value is slightly different: this value does not depend only on λ, but also on $\sqrt{p_j}$, i.e.

$$\nu_{\lambda,min} = \nu_{\lambda,j,min} = \frac{\lambda\sqrt{p_j}}{\tau}\exp\left\{-\frac{1}{2}\right\}. \tag{3.26}$$

If the size of the groups is the same, this value is the same for each group; on the other hand, if p_j is different in the groups, the

value depends on the different groups. Since the value is proportional to $\sqrt{p_j}$, one can consider a single minimum value, fixing it to the maximum value across the J different values of $\sqrt{p_j}$.

From here, it is clear why we define our penalty as *adaptive*: thanks to this result; it is possible to adapt the degree of non-convexity of the penalty based on the level of shrinkage applied (i.e., on the level of λ).

Additional considerations should be made, however. The $\nu_{\lambda,min}$ value found should not be regarded as the best value to estimate the penalized model; it should be understood as the lower bound of a range of possible values for finding the best ν value to use. However, based on numerical studies carried out (and not reported), no substantial differences can be appreciated for small variations in the parameter.

Chapter 4

Simulation

In this chapter, we undertake an extensive simulation study to thoroughly assess the performance of the proposed penalty function compared to well-established alternatives from the existing literature, namely LASSO, SCAD, and MCP. The simulations are designed to cover a wide array of scenarios and are carried out within both the Generalized Linear Model (with and without grouped variables) and the Gaussian Graphical Model frameworks. Additionally, we also present simulation studies to investigate the influence of the parameter ν.

4.1 Mean Squared Error and Area Under Curve

We employ two key evaluation metrics to present the results of our simulation study comprehensively: the Mean Squared Error (MSE) and the Area Under Curve (AUC).

Let ξ be the generic parameter to be estimated, regardless of the framework. The AUC derived from the Receiver Operating Characteristic (ROC) curve. The ROC curve is constructed using the False

Positive Rate (FPR) and the True Positive Rate (TPR), defined as follows:

$$\text{FPR}(\lambda) = \frac{\text{Card}(\hat{\xi}_{bh}(\lambda) \neq 0 \,|\, \xi_h = 0)}{\text{Card}(\xi_h = 0)},$$
$$\text{TPR}(\lambda) = \frac{\text{Card}(\hat{\xi}_{bh}(\lambda) \neq 0 \,|\, \xi_h \neq 0)}{\text{Card}(\xi_h \neq 0)}.$$

AUC provides a comprehensive assessment of methods' abilities to select the subset of coefficients and estimate parameters accurately.

Our second evaluation metric is MSE, which quantifies the accuracy of the estimator. MSE is calculated as the mean of the squared errors and is defined by the following formula

$$\text{MSE}(\lambda) = \frac{1}{B} \sum_{h=1}^{J} \sum_{b=1}^{B} \left(\hat{\xi}_{b,h}(\lambda) - \xi_h \right)^2,$$

where b is the index related to the replicates, ξ_h are the true coefficients, and $\hat{\xi}_{b,h}$ are the estimated coefficients. We evaluate MSE at different point along the estimated path and across a range of tuning parameter values, including 80%, 60%, 40% and 20% of λ_{max}, i.e. the maximum value within each simulation . Moreover, we compute MSE for the entire set of coefficients ξ and separately on the subset of non-null coefficient ($\xi \in \mathcal{A}$) and null coefficients ($\xi \notin \mathcal{A}$). This approach enables us to discern performance variations across different parameter settings.

These two metrics offer a widespread means of evaluating the performance of the penalty methods under investigation, allowing us to draw meaningful conclusions about their effectiveness.

4.2 GLM framework

While our method works for GLMs such as Poisson or Binomial, only the Gaussian case is examined. The underlying model for generating the response variable is given by:

$$y_i = \beta_0 + \sum_{h=1}^{J} x_{ih}\beta_h + \sigma_s \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0,1).$$

We explored two main simulation scenarios:

1. Low-dimensional setting: We considered sample sizes n of 30, 60, and 90 observations with $J = 100$ variables. Two different values $\sigma_s = 1$ and $\sigma_s = 2$, were examined.

2. High-dimensional setting: Here, we kept the sample size n fixed at 100 observations while increasing the number of variables to $J = 1000$. We again studied two noise levels: $\sigma_s = 1$ and $\sigma_s = 5$.

The covariates x_i were generated from a Normal distribution with mean 0 and a Toeplitz correlation matrix $\Sigma_{jk} = 0.5^{|j-k|}$. For each case, only ten coefficients were set as non-null, and their positions were randomly chosen. We considered two scenarios for generating non-zero coefficients: in the former, the values of these non-null coefficients were randomly drawn from a uniform distribution $U(1,2)$, while in the latter, they were drawn from $U(2,3)$. Consequently, we had 16 different scenarios, with 100 replicates for each run.

We considered the minimum allowable value for the additional parameter, i.e. ν_{min}. We set additional parameters for SCAD and MCP to 3.7 and 3, respectively.

Table 4.1 presents the median Area Under the Curve based on the results, where the dataset comprises $J = 100$ variables.

The table is structured to showcase performance across different combinations of parameters. Specifically, it investigates the influence

Table 4.1: Median AUC of simulation results with $J = 100$, varying n, β and σ_s

			β values	
n	σ_s	Penalty	$\beta \sim U(1,2)$	$\beta \sim U(2,3)$
30	1	LASSO	0.790	0.800
		SCAD	0.671	0.680
		MCP	0.665	0.666
		ANP	0.687	0.689
	5	LASSO	0.761	0.792
		SCAD	0.659	0.667
		MCP	0.655	0.666
		ANP	0.645	0.681
60	1	LASSO	0.979	0.983
		SCAD	0.985	0.988
		MCP	0.994	0.995
		ANP	0.991	0.991
	5	LASSO	0.958	0.978
		SCAD	0.955	0.982
		MCP	0.939	0.992
		ANP	0.932	0.988
90	1	LASSO	0.993	0.993
		SCAD	0.994	0.993
		MCP	1.000	1.000
		ANP	1.000	1.000
	5	LASSO	0.983	0.992
		SCAD	0.988	0.994
		MCP	0.996	1.000
		ANP	0.992	1.000

of sample size and noise level on the AUC for each penalty method under two distinct scenarios of coefficient distributions.

LASSO excels in scenarios with low and high noise levels, showcasing its effectiveness in situations with a limited sample size. Notably, our proposed penalty slightly outperforms other non-convex penalty functions, indicating its superior performance in this context. In this context, leveraging a higher value of ν could enhance our proposed method's AUC performance. Section 4.5 will provide a more in-depth exploration of strategies to improve outcomes in these specific scenarios.

In scenarios with a sample size of $n = 60$, under both low ($\sigma_s = 1$) and high noise levels ($\sigma_s = 5$), the ANP consistently outperforms LASSO and SCAD, achieving higher AUC values. This indicates that the proposed penalty function demonstrates a superior ability to accurately identify the subset of non-null coefficients in situations characterised by moderate sample sizes and varied noise levels. In the case of the largest sample size tested, all penalization methods yield excellent results, with our proposal and MCP standing out, achieving perfect AUC values of 1. This underscores the robust performance of our approach in scenarios with larger sample sizes, showcasing its ability to distinguish between true positives and false positives accurately.

Tables 4.2, 4.3, and 4.4 show the Mean Squared Errors for all scenarios across four different values of λ, focusing on subsets of coefficients ($\beta \in \mathcal{A}$), subsets of null coefficients ($\beta \notin \mathcal{A}$), and the entire vector of coefficients. The tuning parameter λ varies at different levels.

Table 4.2: MSE of simulation results with $J = 100$ and $n = 30$, varying β and σ_s

			$\beta \in \mathcal{A}$				$\beta \notin \mathcal{A}$				$\forall \beta$			
			λ				λ				λ			
$\beta \sim U$	σ_s	Penalty	80%	60%	40%	20%	80%	60%	40%	20%	80%	60%	40%	20%
Unif(1,2)	1	LASSO	1.520	1.467	1.383	1.254	0.007	0.036	0.075	0.118	0.481	0.466	0.445	0.414
		SCAD	1.520	1.467	1.383	1.288	0.007	0.036	0.075	0.135	0.481	0.466	0.445	0.432
		MCP	1.515	1.451	1.372	1.324	0.011	0.046	0.100	0.189	0.480	0.463	0.449	0.465
		ANP	1.464	1.446	1.377	1.296	0.139	0.163	0.233	0.292	0.496	0.497	0.501	0.501
	2	LASSO	1.519	1.467	1.387	1.268	0.009	0.034	0.077	0.132	0.481	0.467	0.447	0.423
		SCAD	1.519	1.467	1.388	1.319	0.009	0.034	0.079	0.153	0.481	0.467	0.448	0.449
		MCP	1.514	1.453	1.389	1.347	0.013	0.046	0.106	0.216	0.480	0.466	0.459	0.482
		ANP	1.485	1.469	1.405	1.329	0.117	0.145	0.250	0.339	0.502	0.505	0.517	0.534
Unif(2,3)	1	LASSO	2.384	2.307	2.185	2.005	0.012	0.052	0.109	0.178	0.754	0.733	0.702	0.659
		SCAD	2.384	2.307	2.186	2.061	0.012	0.052	0.110	0.201	0.754	0.733	0.702	0.687
		MCP	2.378	2.290	2.180	2.122	0.017	0.073	0.157	0.300	0.753	0.732	0.715	0.743
		ANP	2.315	2.290	2.199	2.054	0.190	0.256	0.364	0.445	0.778	0.788	0.796	0.784
	2	LASSO	2.382	2.302	2.181	1.993	0.011	0.052	0.111	0.184	0.754	0.731	0.700	0.657
		SCAD	2.382	2.302	2.181	2.085	0.011	0.052	0.111	0.223	0.754	0.731	0.701	0.704
		MCP	2.374	2.282	2.176	2.137	0.015	0.073	0.153	0.317	0.751	0.729	0.713	0.755
		ANP	2.315	2.267	2.172	2.043	0.217	0.252	0.341	0.456	0.783	0.781	0.778	0.786

Table 4.3: MSE of simulation results with $J = 100$ and $n = 60$, varying β and σ_s

			$\beta \in \mathcal{A}$				$\beta \notin \mathcal{A}$				$\forall \beta$			
			λ				λ				λ			
$\beta \sim U$	σ_s	Penalty	80%	60%	40%	20%	80%	60%	40%	20%	80%	60%	40%	20%
U(1,2)	1	LASSO	1.516	1.433	1.286	1.010	0.002	0.011	0.029	0.048	0.479	0.454	0.408	0.323
		SCAD	1.516	1.433	1.286	0.997	0.002	0.011	0.029	0.039	0.479	0.454	0.408	0.319
		MCP	1.508	1.401	1.228	0.848	0.002	0.014	0.032	0.031	0.477	0.444	0.392	0.273
		ANP	1.391	1.296	1.082	0.511	0.034	0.050	0.075	0.067	0.446	0.420	0.358	0.178
	2	LASSO	1.513	1.429	1.285	1.046	0.005	0.022	0.050	0.084	0.478	0.452	0.409	0.338
		SCAD	1.513	1.429	1.285	1.041	0.005	0.022	0.050	0.076	0.478	0.452	0.409	0.336
		MCP	1.504	1.399	1.236	0.971	0.008	0.027	0.053	0.070	0.476	0.444	0.396	0.320
		ANP	1.417	1.336	1.144	0.823	0.080	0.103	0.161	0.155	0.458	0.440	0.391	0.306
U(2,3)	1	LASSO	2.369	2.238	2.003	1.546	0.003	0.015	0.039	0.067	0.749	0.708	0.636	0.495
		SCAD	2.369	2.238	2.003	1.525	0.003	0.015	0.039	0.060	0.749	0.708	0.636	0.487
		MCP	2.356	2.193	1.919	1.286	0.004	0.019	0.042	0.072	0.745	0.695	0.612	0.416
		ANP	2.207	2.090	1.692	0.793	0.048	0.076	0.121	0.155	0.708	0.676	0.564	0.281
	2	LASSO	2.372	2.258	2.043	1.620	0.005	0.022	0.050	0.084	0.751	0.715	0.649	0.521
		SCAD	2.372	2.258	2.043	1.619	0.005	0.022	0.050	0.076	0.750	0.715	0.649	0.520
		MCP	2.361	2.219	1.969	1.406	0.008	0.027	0.053	0.070	0.747	0.704	0.629	0.457
		ANP	2.235	2.131	1.783	0.971	0.080	0.103	0.161	0.155	0.723	0.695	0.601	0.347

Table 4.4: MSE of simulation results with $J = 100$ and $n = 90$, varying β and σ_s

			$\beta \in \mathcal{A}$				$\beta \notin \mathcal{A}$				$\forall \beta$			
			λ				λ				λ			
$\beta \sim U$	σ_s	Penalty	80%	60%	40%	20%	80%	60%	40%	20%	80%	60%	40%	20%
Unif(1,2)	1	LASSO	1.508	1.409	1.215	0.863	0.001	0.005	0.013	0.020	0.477	0.446	0.385	0.274
		SCAD	1.508	1.409	1.215	0.819	0.001	0.005	0.013	0.011	0.477	0.446	0.385	0.260
		MCP	1.497	1.369	1.121	0.606	0.001	0.005	0.007	0.003	0.473	0.433	0.355	0.192
		ANP	1.364	1.226	0.849	0.167	0.020	0.020	0.025	0.011	0.435	0.391	0.273	0.056
	2	LASSO	1.512	1.411	1.228	0.911	0.001	0.007	0.021	0.038	0.478	0.447	0.389	0.291
		SCAD	1.512	1.411	1.228	0.880	0.001	0.007	0.021	0.030	0.478	0.447	0.389	0.280
		MCP	1.503	1.371	1.144	0.730	0.001	0.006	0.016	0.020	0.475	0.434	0.363	0.233
		ANP	1.365	1.265	0.918	0.442	0.027	0.031	0.049	0.059	0.436	0.406	0.299	0.156
Unif(2,3)	1	LASSO	2.367	2.217	1.916	1.316	0.002	0.012	0.030	0.034	0.749	0.701	0.608	0.418
		SCAD	2.367	2.217	1.916	1.245	0.002	0.012	0.030	0.020	0.749	0.701	0.607	0.395
		MCP	2.353	2.159	1.779	0.817	0.003	0.014	0.024	0.001	0.744	0.684	0.565	0.258
		ANP	2.154	1.958	1.145	0.114	0.035	0.050	0.050	0.004	0.687	0.628	0.371	0.037
	2	LASSO	2.372	2.226	1.934	1.369	0.002	0.010	0.026	0.038	0.751	0.704	0.613	0.436
		SCAD	2.372	2.226	1.934	1.299	0.002	0.010	0.026	0.026	0.750	0.704	0.613	0.413
		MCP	2.360	2.170	1.792	0.917	0.003	0.012	0.022	0.009	0.746	0.687	0.569	0.291
		ANP	2.166	1.944	1.331	0.359	0.038	0.049	0.070	0.030	0.691	0.624	0.435	0.121

Table 4.2 presents the MSE values for $n = 30$, revealing insights into the performance of different penalization methods across scenarios. Despite observing higher errors in our penalty for the vector of non-zero coefficients, the scenario slightly shifts positively when focusing on the active set. Our method, particularly at higher tuning parameter values, often surpasses LASSO in effectiveness, which seems the most balanced penalty. However, when considering the entire coefficient vector, the analysis suggests our method's sensitivity to error for non-zero coefficients, positioning it as less favourable in this broad assessment.

When the sample size is $n = 60$ (Table 4.3), our proposal continues to provide the largest errors for the vector of null coefficients, but it is observed that the distance to the competitors gets thinner; again, LASSO seems to be the best method. Regarding the vector of active coefficients, our method outperforms the competitors in terms of MSE: while the differences are imperceptible for high values of λ, corresponding to very sparse estimated models, the differences become more pronounced as variables enter the model, particularly with smaller values of λ, sometimes reaching half the errors of the competitors. Decreasing distance from competitors for null coefficients and better performance for nonzero coefficients makes our method the best method for errors over the entire vector of coefficients, always achieving the lowest MSE values.

When the sample size increases (Table 4.4), the problem with the error for null coefficients remains. However, focusing on the vectors of non-null parameters, we observe that our errors continue to be the lowest across every value of the tuning parameter and in every setting, demonstrating a significantly greater distance than the previous table's case. Particularly, when coefficients are generated with higher intensity, for very small values of λ, our error can be up to a tenth of that obtained with LASSO and SCAD. Similar assessments can be extended to the global error: The gap from competitors is narrowed, but our proposal remains the best in every scenario and for every degree of sparsity.

However, some assessments must be made. Competitors exhibit

superior performance when considering the vector of null coefficients (i.e., $\beta \notin \mathcal{A}$). The result of the MSE for the coefficients not belonging to the active set seems to contradict that obtained in the AUC (the result may erroneously suggest that our model estimates too many non-zero variables). This is not true: our penalty allows each newly active coefficient to have a high magnitude instantly, given that the maximum non-convexity of the penalty function guaranteeing a single solution allows the active coefficients to be almost non-biased; conversely, the other three penalty functions attenuate the magnitude of the newly active coefficients that should be null, thus making the total error committed lower. The error obtained with the LASSO is the lowest of all scenarios since the penalty imposes the highest shrinkage on the coefficients. This observation prompts a strategic consideration for adjusting the tuning parameter ν, aiming to refine our penalty's performance closer to LASSO's efficiency. Further exploration of this adjustment strategy offers promising avenues for enhancing our method, as detailed in Section 3.6.

Furthermore, in the high-dimensional case ($J = 1000$), we present AUC and MSE values across four distinct values of λ.

Table 4.5: Median AUC of simulation results with $J = 1000$ and $n = 100$, varying β and σ_s

		$\beta \sim U$	
σ_s	Penalty	U(1,2)	U(2,3)
1	LASSO	0.999	0.999
	SCAD	0.999	0.999
	MCP	1.000	1.000
	ANP	1.000	1.000
5	LASSO	0.784	0.936
	SCAD	0.639	0.836
	MCP	0.641	0.797
	ANP	0.667	0.833

Table 4.5 provides simulation results for the median Area Under the Curve under varying conditions, including $J = 1000$ and $n =$

100, exploring different combinations of penalty methods, β distributions, and σ values. When σ remains relatively low, the performance of the penalty functions is virtually indistinguishable, with negligible differences. However, as σ increases, the non-convex penalties exhibit slightly inferior performance compared to the LASSO.

Table 4.6: MSE of simulation results with $J = 1000$ and $n = 100$, varying β and σ_s

			$\beta \in \mathcal{A}$				$\beta \notin \mathcal{A}$				$\forall \beta$			
			λ				λ				λ			
$\beta \sim U$	σ_s	Penalty	80%	60%	40%	20%	80%	60%	40%	20%	80%	60%	40%	20%
(1,2)	1	LASSO	1.537	1.437	1.239	0.877	0.013	0.013	0.013	0.013	0.154	0.144	0.124	0.088
		SCAD	1.537	1.437	1.239	0.787	0.013	0.013	0.013	0.009	0.154	0.144	0.124	0.079
		MCP	1.528	1.394	1.114	0.513	0.013	0.013	0.011	0.005	0.153	0.140	0.112	0.051
		ANP	1.376	1.206	0.709	0.151	0.016	0.016	0.014	0.007	0.139	0.122	0.073	0.017
	5	LASSO	1.542	1.473	1.373	1.274	0.017	0.024	0.041	0.068	0.154	0.148	0.142	0.143
		SCAD	1.542	1.473	1.373	1.307	0.017	0.024	0.041	0.081	0.154	0.148	0.142	0.154
		MCP	1.535	1.447	1.344	1.346	0.017	0.028	0.058	0.132	0.154	0.147	0.146	0.189
		ANP	1.468	1.419	1.345	1.309	0.056	0.096	0.141	0.179	0.159	0.173	0.194	0.220
(2,3)	1	LASSO	2.381	2.237	1.934	1.342	0.019	0.019	0.019	0.017	0.238	0.224	0.194	0.135
		SCAD	2.381	2.237	1.934	1.246	0.019	0.019	0.019	0.013	0.238	0.224	0.194	0.125
		MCP	2.369	2.183	1.780	0.765	0.019	0.019	0.017	0.007	0.237	0.218	0.178	0.076
		ANP	2.175	1.945	1.053	0.151	0.026	0.025	0.021	0.007	0.219	0.197	0.108	0.017
	5	LASSO	2.380	2.268	2.059	1.765	0.020	0.025	0.039	0.063	0.238	0.227	0.209	0.186
		SCAD	2.380	2.268	2.060	1.751	0.020	0.025	0.039	0.063	0.238	0.227	0.209	0.185
		MCP	2.369	2.223	1.957	1.620	0.021	0.028	0.046	0.087	0.237	0.223	0.201	0.184
		ANP	2.239	2.091	1.830	1.617	0.048	0.073	0.115	0.168	0.231	0.224	0.218	0.232

Table 4.6 presents the Mean Squared Errors for the experiments conducted in the previous subsection. Analyzing the entire vector of coefficients, our proposed method performs better than competitors when the noise levels are moderate, across different values of λ and for each type of distribution considered for the coefficients. When narrowing our focus to the subset of truly non-zero coefficients ($\beta \in \mathcal{A}$), our approach with the specified ν values consistently achieves superior results. Our model performs similarly with respect to other penalties when noise levels are low in scenarios where we are specifically interested in the subset of truly null coefficients ($\beta \notin \mathcal{A}$). However, as noise levels increase, employing a higher ν value appears to be a favourable strategy, leading to improved performance compared to competitors.

4.3 Grouped variables framework

The performance of the proposed estimator is compared with the well-known group-LASSO, group-SCAD, and group-MCP estimators. To assess the quality of the estimator, we consider three different data generation models, similar to those proposed by Yuan and Lin (2006):

- In the first scenario, we simulated J=150 latent variables, $X \sim N_p(0, \Sigma)$, with a Toeplitz correlation matrix $\Sigma_{jk} = 0.5^{|j-k|}$. We then categorized each X_j into three groups (0, 1, or 2), depending on whether it was less than $\Phi^{-1}(1/3)$, greater than $\Phi^{-1}(2/3)$, or in between. Consequently, the design matrix Z will have 300 columns. We randomly selected eight different groups of variables with non-zero coefficients, and for each group we drew two coefficients randomly from a uniform distribution ($U(1, 8)$);

- In the second model, we consider combining the two models. We generated 75 variables according to the first model and then generated 75 other independent vectors from a standard

normal distribution ($N(0,1)$). For each of the 75 vectors, we considered its third-degree polynomial. This resulted in a design matrix Z with 375 columns. We randomly chose eight group variables different from 0, and their coefficients were drawn from a uniform distribution ($U(1,\,2)$).

For each model, we generated $y = Z\beta + \epsilon$, where $\epsilon \sim N(0, 1.5)$ and $n = 50$. In each scenario, we conducted 500 simulations. To evaluate the performance of the estimators, we computed the MSE and the AUC. Regarding the MSE, we considered, for each replicate and each estimator, the minimum value obtained in the λ-path. This way, we compared the best value that the estimator could achieve. In terms of additional parameters, we set γ to 4 for SCAD and 3 for MCP (as done by default in the `R` package "`grpreg`" (Breheny and Huang, 2015), used for simulations). For our proposal, we used the minimum value of ν, defined as (3.26). Table 4.7 shows the results of the numerical study.

Table 4.7: Values averaged over the 500 replicates (standard deviation in brackets)

		Group ANP	Group LASSO	Group SCAD	Group MCP
Model I	AUC	0.956	0.882	0.946	0.959
		(0.04)	(0.04)	(0.04)	(0.04)
	MSE	2.175	8.664	2.207	2.161
		(1.20)	(2.29)	(1.11)	(1.22)
Model II	AUC	0.866	0.891	0.886	0.883
		(0.08)	(0.06)	(0.07)	(0.07)
	MSE	1.554	1.807	1.486	1.597
		(0.47)	(0.39)	(0.42)	(0.47)

Examining the results for the first model in the provided table, the AUC values are high across all groups, indicating good model performance in distinguishing between the classes. Group MCP has the highest AUC (0.959), closely followed by Group ANP (0.956) and Group SCAD (0.946). Group LASSO has the lowest AUC (0.882), suggesting it is less effective in this model than the others. The

standard deviation is 0.04 for all groups, indicating a similar level of variability in the AUC metric across the methods. The MSE values vary significantly across the groups. Group MCP has the lowest MSE (2.161), slightly better than Group ANP (2.175) and Group SCAD (2.207), indicating these methods are more accurate in their predictions. Group LASSO has a considerably higher MSE (8.664), with a higher standard deviation (2.29), suggesting less accuracy and more variability in its predictions.

In the framework of the second model, the AUC values are generally lower, suggesting Model II is a more challenging scenario. Group LASSO leads in this model with an AUC of 0.891, an interesting contrast to its performance in Model I. The other groups have AUC values close to each other, ranging from 0.866 (Group ANP) to 0.886 (Group SCAD), with Group MCP slightly lower at 0.883. The standard deviations range from 0.06 to 0.08, indicating slightly more variability in AUC scores in Model II than in Model I. The MSE values are lower across all groups in Model II compared to Model I, indicating all methods perform better in accuracy. Group SCAD has the lowest MSE (1.486), suggesting it is the most accurate method in Model II. The MSE values for the other groups are relatively close, with Group LASSO having a slightly higher MSE (1.807) than the others.

4.4 GGM framework

In this section, we present the results of our simulation studies comparing the performance of our penalization approach in the context of Graphical Gaussian Models against LASSO and SCAD. The primary objective is to assess the effectiveness of our proposal in accurately recovering the underlying graphical structure of the data while examining the errors incurred during coefficient estimation.

We consider a dataset with a dimensionality of $J = 100$, resulting in a Θ matrix of size 100×100 that encodes the graph structure. To simulate the sparse nature of this graph, we randomly set $\theta_{i,j} =$

$\theta_{j,i} = 1$ with $\theta_{i,j} \sim Ber(0.1)$. We investigate the performance across four distinct sample sizes, namely, $n = 30$, 60, 90, and 120.

Table 4.8 presents the Mean Squared Errors calculated using the three different methods: our proposed method, gLASSO, and gSCAD. These results are evaluated under various tuning parameter values (λ) for each sample size, providing insights into the accuracy and robustness of these methods in the context of GGMs.

Table 4.8: MSE of simulation results with $J = 100$, varying n

		$\theta \in \mathcal{A}$				$\theta \notin \mathcal{A}$				$\forall\theta$			
		λ				λ				λ			
n	Penalty	80%	60%	40%	20%	80%	60%	40%	20%	80%	60%	40%	20%
30	ANP	11.098	10.973	10.230	8.606	0.076	0.463	1.764	4.270	11.098	10.984	10.391	9.617
	LASSO	11.098	10.977	10.256	8.530	0.076	0.453	1.713	4.174	11.098	10.988	10.407	9.504
	SCAD	11.098	10.977	10.270	9.614	0.076	0.453	1.748	5.137	11.098	10.988	10.431	10.918
60	ANP	11.080	10.926	10.106	7.980	0.044	0.326	1.253	3.310	11.081	10.931	10.188	8.649
	LASSO	11.080	10.930	10.128	8.000	0.044	0.321	1.233	3.271	11.081	10.935	10.207	8.653
	SCAD	11.080	10.930	10.127	8.419	0.044	0.321	1.234	3.835	11.081	10.935	10.206	9.273
90	ANP	11.072	10.896	10.022	7.721	0.021	0.249	0.980	2.645	11.072	10.900	10.072	8.168
	LASSO	11.072	10.900	10.042	7.757	0.021	0.247	0.971	2.624	11.072	10.903	10.091	8.195
	SCAD	11.072	10.900	10.040	7.768	0.021	0.247	0.970	2.923	11.072	10.903	10.088	8.313
120	ANP	11.081	10.872	9.915	7.549	0.031	0.258	0.887	2.250	11.081	10.875	9.956	7.883
	LASSO	11.081	10.876	9.935	7.586	0.031	0.255	0.881	2.239	11.081	10.879	9.976	7.915
	SCAD	11.081	10.876	9.930	7.329	0.031	0.255	0.882	2.535	11.081	10.879	9.971	7.772

The table's results highlight differences in the performance of the ANP, LASSO, and SCAD penalty methods across various simulation scenarios characterized by changing sample sizes and regularization strengths. As regularization strength decreases from 80% to 20%, we observe a general trend of decreasing MSE for coefficients within the active set ($\theta \in \mathcal{A}$) across all penalty methods and sample sizes. This indicates that a less strict penalty, or lower λ, tends to improve the accuracy of estimating non-zero coefficients (as expected). Conversely, for coefficients not within the active set ($\theta \notin \mathcal{A}$), an increase in MSE with decreasing λ is observed, suggesting that a stronger penalty aids in effectively reducing the estimates of truly zero coefficients towards zero, thus diminishing their contribution to the overall MSE.

The effect of sample size on MSE is also evident, with larger sample sizes consistently leading to reduced MSE for both sets of coefficients. This improvement is more pronounced for coefficients within $\mathcal{A}$, likely due to the increased data providing a more solid basis for accurately estimating non-zero coefficients. When comparing the three penalty methods, the differences in MSE are relatively minor, suggesting that ANP, LASSO, and SCAD perform comparably in estimation accuracy. However, slight variations can be seen, particularly at lower levels of λ for coefficients not in $\mathcal{A}$, where SCAD occasionally exhibits slightly higher MSE values compared to ANP and LASSO.

In evaluating the performance across all coefficients ($\forall\theta$), the MSE trends closely mirror those observed for coefficients in $\mathcal{A}$, though with expected slight increases in MSE since the evaluation now also encompasses the accuracy of estimating zero coefficients. This comprehensive evaluation underscores the importance of selecting appropriate λ and penalty methods based on the modelling objectives, including the need to precisely identify significant predictors while minimizing the impact of non-significant ones.

Table 4.9 presents the median Area Under the Curve values. As the sample size increases from 30 to 120, a clear trend of increasing median AUC values is observed for all penalty methods

Table 4.9: Median AUC of simulation results with $J = 100$, varying n.

	Penalty		
n	ANP	LASSO	SCAD
30	0.608	0.605	0.610
60	0.711	0.698	0.713
90	0.781	0.757	0.782
120	0.841	0.811	0.841

(as expected). The ANP and SCAD methods exhibit very similar performance across all sample sizes.

While showing a consistent improvement in median AUC values with increasing sample sizes, LASSO generally performs slightly less effectively than ANP and SCAD. Although respectable at $n = 120$, LASSO's median AUC of 0.811 is lower than the 0.841 achieved by both ANP and SCAD.

4.5 The effect of ν

As sketched above, the additional parameter ν influences the results, specifically the convergence speed of non-null coefficients to the maximum likelihood estimates. The smaller the value, the closer the estimates. However, the smallest allowable value of ν (i.e. ν_{min}) is not always the best. The maximum likelihood theory requires that n is "sufficiently larger than J" (see, for example, Sur and Candès (2019)). If $n < J$, ML theory does not apply: estimates far from the ML ones are expected to perform better in these contexts. If ν_{min} does not guarantee to have the best solution when $n < J$, the natural question is about the optimal ν, i.e. ν_{opt}.

To try to provide some insights about ν_{opt}, we investigate the evolution of the estimator's performance as ν and the n/J ratio change, a further simulation study was undertaken. We simulated

data from

$$y_i = \beta_0 + \sum_{j=1}^{J} x_{ij}\beta_h + \epsilon_i \quad \epsilon_i \sim \mathcal{N}(0,1).$$

The number of coefficients is fixed to $J = 100$ (only $p_{\mathcal{A}} = 10$ are non-null); to study the influence of ν at different sample sizes, we consider different n. Some preliminary studies suggests that the ν_{opt} depends on $n/p_{\mathcal{A}}$. In other words, the cardinality of the active set. Then, we will consider 21 different sample-size n: the first 20 are fixed to have $n/p_{\mathcal{A}}$ ratios in [0.2, 3] and equally-spaced, the last is fixed to have $n/p_{\mathcal{A}} = 4$. The covariates are defined as $x_i \sim \mathcal{N}(0, \Sigma)$ with the Toeplitz correlation matrix, i.e. $\Sigma_{jk} = 0.5^{|j-k|}$. The locations of non-null coefficients are randomly chosen, and their values are drawn randomly from a $U(1, 2)$. We ran 100 replicates for each scenario, and for each replicate, we fitted 40 different penalized models using 40 different ν-values, i.e.

$$\nu_k = k \times \nu_{min}.$$

The 40 different values of ν proved are composed of an "expansion factor" $k \geq 1$: in this way, different values of increasing ν are used, guaranteeing the solution's uniqueness. The different values of k are 40 equispaced values in [1, 100]: the first value we use, therefore, corresponds to ν_{min}; on the other hand, as the "expansion factor" increases, the estimated model becomes more and more similar to LASSO. In each simulation, the parameter λ was set to the value that minimises the BIC.

Figure 4.1 shows the values of the MSE varying the coefficient of expansion (i.e. ν) and the ratio $n/p_{\mathcal{A}}$. The MSEs have been standardised in [0,1] by column, to make it easier to read the results when the ratio $n/p_{\mathcal{A}}$ varies: a standardisation carried out on all values would not have made the interpretation of the result easier, since the variations in the results due to the effect of k would have been covered by the influence of the ratio $n/p_{\mathcal{A}}$. Values close to 0

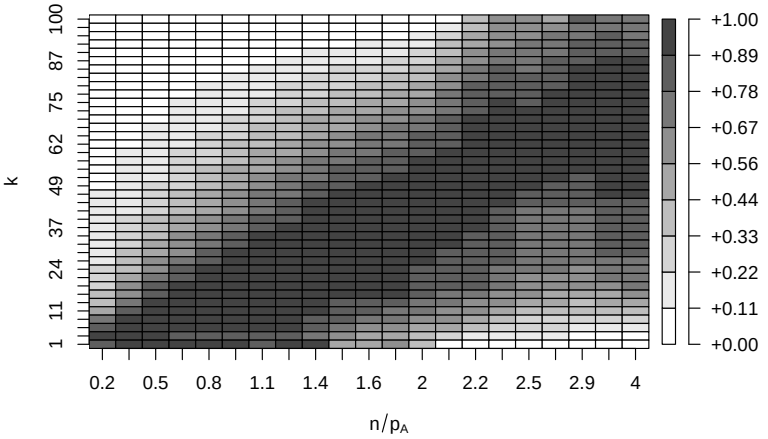


Figure 4.1: Scaled MSE (by $n/p_{\mathcal{A}}$), varying k and $n/p_{\mathcal{A}}$. Lighter squares correspond to lower MSE (better).

indicate better performance (as they correspond to the best values for a given $n/p_{\mathcal{A}}$), while values close to 1 indicate poor performance (as they are close to the maximum MSE calculated for a given $n/p_{\mathcal{A}}$).

It can be seen from the graph that in the presence of a low sample size with respect to a high number of non-null parameters, the best performance is obtained using a high value of $n/p_{\mathcal{A}}$. The interpretation of this result is quite simple: by using small values of ν, the estimates of the model quickly get closer to the maximum likelihood estimates. Since the maximum likelihood estimator requires the sample size to be larger than the number of parameters to be estimated, the penalised estimator with small ν therefore "inherits" the same difficulties as the maximum likelihood estimator. In this context, using a LASSO-like form of the penalty (which introduces bias into the estimated parameters at the cost of reduced variance) gives better results. Conversely, as the ratio $n/p_{\mathcal{A}}$ increases, the performance obtained with a reduced value of ν tends to improve;

for values of the ratio $n/p_{\mathcal{A}}$ greater than about 2, the MSEs calculated with the smallest value of ν are always the better. This is because the maximum likelihood estimator performs better with more information available, and so does the penalised model.

In the simulation study, we tried the same setting with much higher values of $n/p_{\mathcal{A}}$ (up to 20), and the pattern remained the same. However, for readability reasons, we reported values up to $n/p_{\mathcal{A}}$ equal to 4.

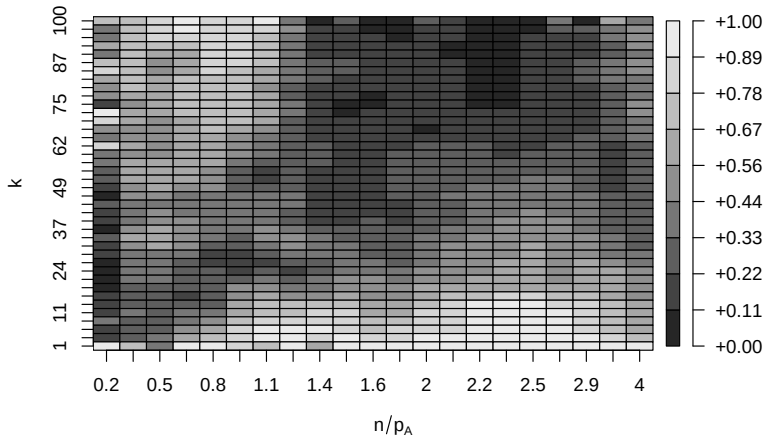


Figure 4.2: Scaled AUC (by $n/p_{\mathcal{A}}$), varying k and $n/p_{\mathcal{A}}$. Lighter squares correspond to higher AUC (better).

Figure 4.2 shows the results based on the AUC. The values have been standardised with respect to the different $n/p_{\mathcal{A}}$ values for the same reasons as above. In this context, however, the best results are obtained for the $n/p_{\mathcal{A}}$ values that have a value of 1, since they correspond to the combination that obtained the highest AUC value (and therefore the best identification of the correct subset of non-zero coefficients).

Again, it can be observed that for larger values of $n/p_{\mathcal{A}}$, the

use of large ν gives better results: as $n/p_{\mathcal{A}}$ increases, it is more convenient to use smaller ν. Although the rationale is the same as that observed in Figure 4.1, in the case of the AUC it is observed that the reversal of the trend occurs much earlier (already for $n/p_{\mathcal{A}}$ equal to 1.1, the use of the smaller ν is better).

Chapter 5

Real data analysis

In this section, we delve into analyzing two real-world datasets: the aim is to assess any differences between our penalized model and those known from the literature. This exploration is essential for confirming how well the Adaptive Non-Convex Penalty function works in practical situations. By comparing our approach with established methods, we highlight our model's advantages and potential limitations within the context of Generalized Linear Models and Gaussian Graphical Models.

The analysis begins by presenting the real-world dataset selected. Through the analysis, we aim to showcase the performance of the ANP function in handling real-world data and provide insights into comparison with traditional penalized methods.

5.1 GLM framework

The dataset utilized in this analysis originates from the "Childhood Asthma and Environmental Study," which was carried out between September 2011 and 2017. This extensive research was conducted at the Pediatric Pulmonology & Allergology outpatient clinic, part of

the CNR-IBIM research unit. The dataset has been obtained from the PhD thesis of Cilluffo (2018).

This dataset focuses on 529 asthmatic children aged 5–17 years and encompasses a wide array of variables collected through a modified version of the SIDRIA (Italian Studies on Respiratory Disorders in Children and the Environment) questionnaire. This comprehensive dataset includes socio-demographic characteristics, parental history of asthma, early and current outdoor and indoor environmental exposures, child's history of wheezing, co-morbidities, asthma severity level, and asthma control status.

Pulmonary function tests were conducted using a portable spirometer to measure the Forced Expiratory Volume in the first second (FEV1), with results appropriately transformed based on the Global Lungs Initiative guidelines. The goal is to assess the determinants of lung function, considering some patient variables.

We estimate four penalized regression models to achieve this: our novel proposal and the established LASSO, SCAD, and MCP methods.

The dataset under analysis comprises $n = 529$ units, with $J = 82$ explanatory variables at our disposal. Given the insights from the simulation results detailed in Section 4.5, we adopt the smallest permissible value for ν in our proposed model. Concurrently, we set the tuning parameters to 3.7 and 3 for SCAD and MCP, respectively. We estimate the competitors' models using the `ncvreg` R package.

Figure 5.1 provides a comparative depiction of the path coefficient for the four penalized regression models. The path coefficient graph for LASSO underscores the constant shrinkage, with parameters intensifying steadily as they approach the peak at λ near 0. A similar pattern is initially discernible for SCAD and MCP; both begin akin to LASSO due to their overlap in the early penalization phase. However, the graph reveals a pivotal moment where the coefficient intensities for SCAD and MCP increase significantly until they reach a stage where the growth is very small.

In contrast, our proposed model exhibits a distinctive pattern: upon activation, each parameter's estimate promptly explodes in

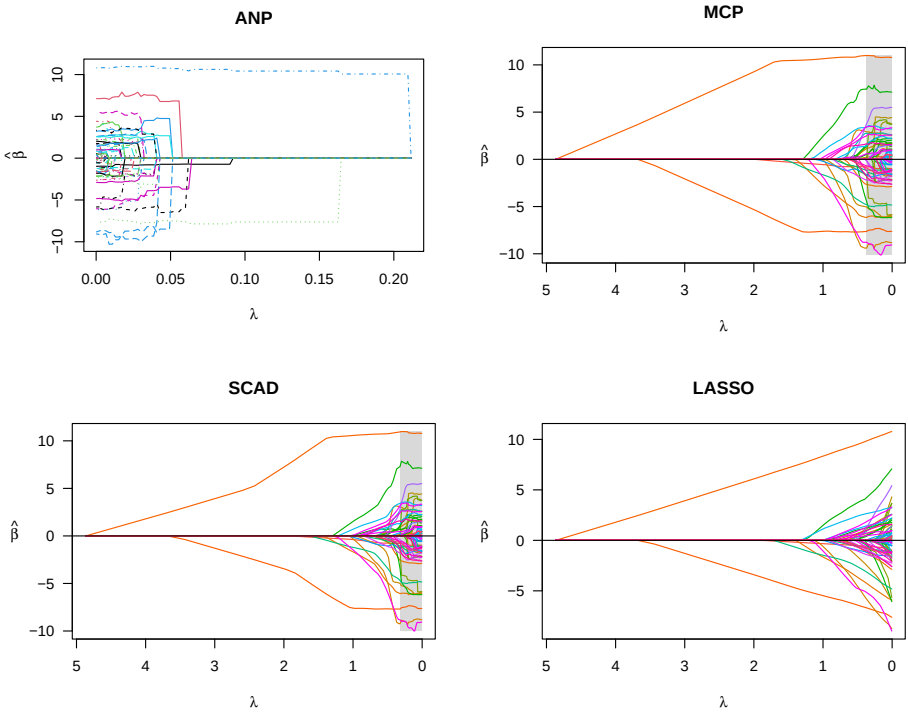


Figure 5.1: Path Coefficient for Penalized Regression Models: ANP, MCP, SCAD, and LASSO

intensity. This estimate remains the same until the activation of another parameter, delineating a swift 'convergence' to the values akin to those from maximum likelihood estimates. This behaviour reflects, even graphically, the maximum speed of convergence of the estimates to those of maximum likelihood. However, the behaviour observed for our proposal is typical of the choice of parameter ν, which has been set at the minimum allowable value. By increasing ν, the path will increasingly resemble that of LASSO.

Coefficients	**ANP**	**MCP**	**SCAD**	**LASSO**
Intercept	94,32	95,91	95,24	95,98
Age	//	-0,24	-0,18	-0,15
Asthma (ref = No)	-7,63	-6,74	-5,63	-4,53
Gender (ref = F)	10,42	10,44	9,77	7,22
Pasthma (ref = No)	//	//	-0,27	-0,49

Table 5.1: Estimated non-null coefficients for penalized models at λ minimizing BIC

Table 5.1 shows the estimated non-zero coefficients for each model, selecting the regularization parameter that minimizes the BIC. Null coefficients, which the models deem non-significant at this λ value, are omitted to enhance the table's clarity. The "//" entries, representing null coefficients at the optimal BIC λ, reflect each model's decision-making on variable selection.

Notably, the ANP model did not find Age and Pasthma (i.e. indicating a paternal history of asthma) as significant predictors at this λ level, suggesting a possible lower impact of these variables. In contrast, MCP and SCAD attribute a slight negative influence to Age and Pasthma, indicating these factors slightly reduce the response variable when other covariates are held constant.

The consistent negative coefficient for Asthma across all models confirms its expected inverse relationship with the outcome. Gender shows a positive association in all models, with LASSO attributing a lesser magnitude, potentially indicating a more conservative esti-

mation.

5.2 GGM framework

In this context, we will use the stock prices of companies listed in the S&P500 index from 2003 to 2008, a market-capitalization-weighted index tracking the performance of 500 of the largest publicly traded companies in the United States. Widely used as a barometer for the overall health of the U.S. stock market, it serves as a benchmark for investors and is categorized into distinct sectors. The included spans many areas, such as Energy, Financials, Health Care and Utilities.

For the purposes of the analysis, we will focus, in particular, on companies that fall within the Utilities sector. The decision to focus on the "Utilities" category involves examining companies that provide essential services such as electricity, water and gas. The objective is to evaluate the potential existence of a conditional dependency structure among various actors within this sector. We will estimate penalized Gaussian graphical models to achieve this, elucidating company interdependencies. Additionally, we will evaluate whether the "optimal" outcomes derived from the diverse models exhibit divergences or remain consistent.

The dataset contains 32 companies operating in the Utilities sector, and for each of them, we have 1260 share prices for each of the opening days from 2003 to 2008.

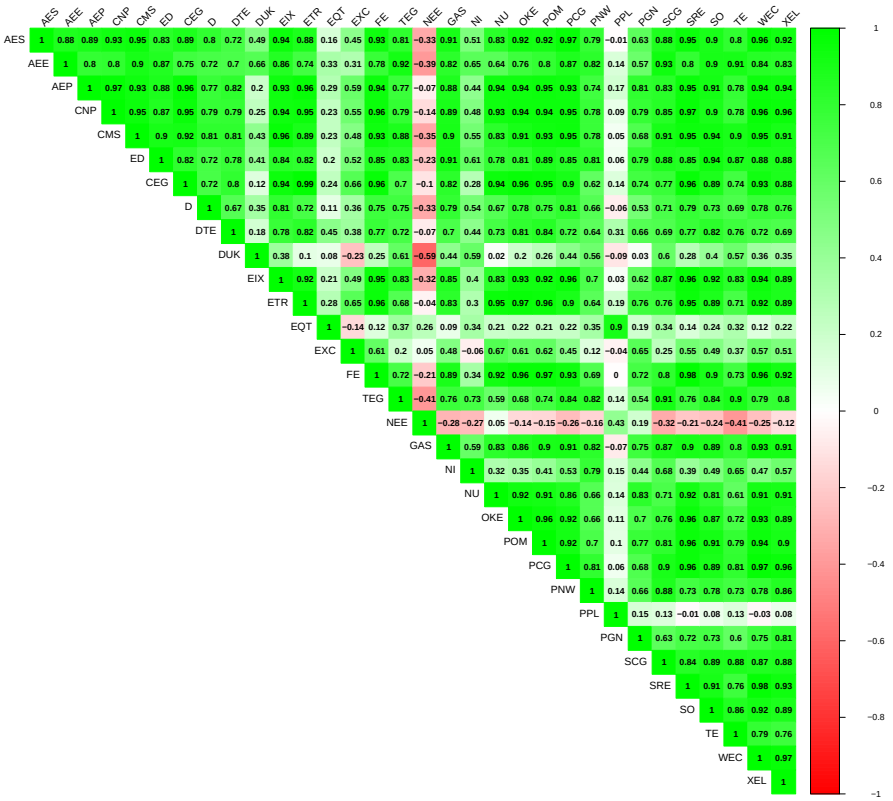


Figure 5.2: Heatmap of the correlation matrix of stock prices of companies

The correlation matrix heatmap in Figure 5.2 illustrates the pair-

wise correlations between variables. The colour intensity represents the strength and direction of the correlations. Only the upper triangular portion of the matrix is displayed, and coefficient values are shown in black. Examining the graph reveals a robust positive correlation in the price trends of all companies, albeit with varying degrees of intensity—indicating a tendency for them to rise or fall collectively. Notably, two companies stand out from this general pattern. Firstly, PPL Corporation (PPL) appears somewhat disconnected from the overall trend, except for its correlation with EQT Corporation (EQT). Secondly, NextEra Energy Resources (NEE) exhibits an inversely proportional correlation compared to the remaining companies.

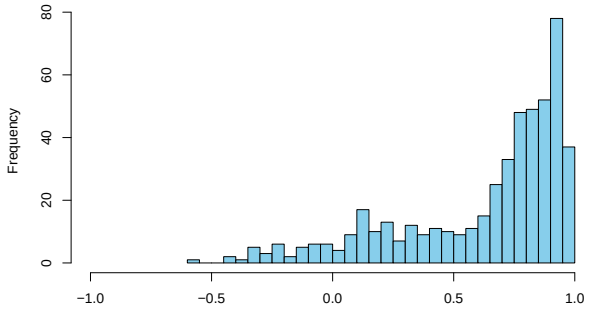


Figure 5.3: Histogram of the upper triangle values of the correlation matrix

The histogram in Figure 5.3 displays the distribution of values in the upper triangle of the correlation matrix. It confirms earlier observations, reaffirming a predominant positive and strongly correlated values pattern. The scarcity of values below zero further supports the overarching trend, indicating a collective inclination

Penalty	BIC	# Edges
ANP	685895.7	32
SCAD	772283.3	163
LASSO	823649.2	232

Table 5.2: BIC and number of edges by penalty function

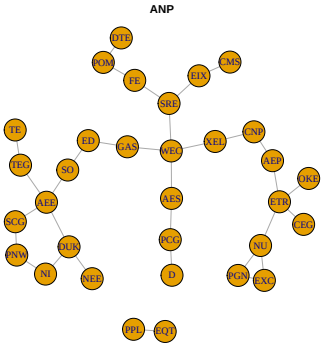
for the stock prices to rise or fall simultaneously.

Building upon these visual explorations, our analysis estimates penalized Gaussian graphical models using our penalty function, LASSO, and SCAD to study the structured sparsity and conditional dependencies among stock prices.

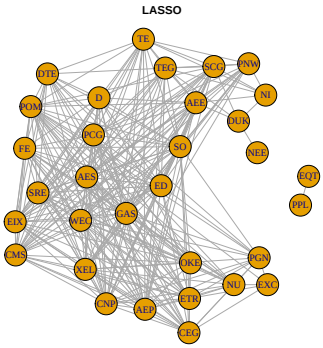
The resulting graphical models are assessed using Bayesian Information Criterion (BIC) values. Specifically, we calculate BIC values for each penalty method across a range of tuning parameter values, enabling us to identify the optimal model complexity that balances the goodness of fit with model parsimony. The additional parameter ν is settled to the minimum value allowable ν_{min}, while $\gamma_{SCAD} = 3.7$.

Looking at Figure 5.4, notable differences emerge among the precision matrices estimated by various penalty methods. The precision matrix obtained with LASSO, which minimizes the BIC, appears remarkably dense, comprising 232 links (46.77% of possible links) and resulting in a BIC value of 823649.2. An intermediate scenario is observed with SCAD, producing a graph with 163 links (32.86%) and a BIC equal to 772283.3. In contrast, our proposed penalty method generates a sparse graph with only 32 links (6.45%) and a lower BIC equal to 685895.7.

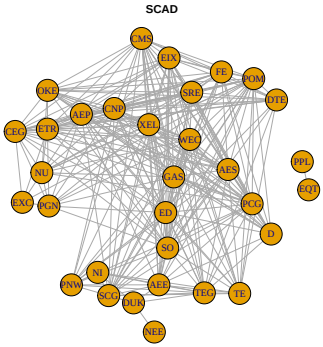
It is essential to emphasize that the BIC value for our proposed penalty is much lower than that calculated with the other models. Our method offers a more interpretable scenario, facilitating a clearer interpretation of the underlying reality. While further investigation is warranted, the observed superior performance may be attributed to the unique characteristics of the Adaptive Non-Convex



(a) ANP



(b) LASSO



(c) SCAD

Figure 5.4: Precision matrix graph for the corresponding method.

Penalty, such as its ability to balance sparsity, unbiasedness and the degree of non-convexity, as discussed in Section 3.6.

Chapter 6

Conclusions

This thesis introduces an innovative method, the Adaptive Non-Convex Penalty function, specifically tailored to tackle the complexities of high-dimensional data analysis through a penalty-based approach. Our methodology extends the application of Adaptive Non-Convex Penalty to encompass Generalized Linear Models, accommodating grouped variables and Gaussian Graphical Models. This extension positions our approach as a versatile and adaptable tool for diverse applications.

The fundamental innovation resides in our meticulously designed penalty function, strategically crafted to address the challenges inherent in high-dimensional settings. At the heart of the methodology lies the parameter ν, which is pivotal in determining solution uniqueness and governing the convergence rate towards maximum likelihood estimates. By harnessing the power of the Alternating Direction Method of Multipliers, we deciphered the non-convex nature of the penalty function, thereby identifying the ν_{min} value, crucial for ensuring gradient monotonicity and consequently ensuring the uniqueness of the solution. Thanks to this step, it is possible to define the degree of non-convexity of the penalty function so that it does not override the degree of convexity of the loss function. In

this way, we can guarantee the optimal convexity of the penalised objective function, thus recovering all benefits.

Our approach underwent extensive simulations benchmarked against existing penalization techniques across diverse frameworks: it often performs better than all competitors. Despite varying performance outcomes, our methodology exhibits superior qualities in many scenarios, notably in the Mean Squared Error and Area Under the Curve values.

Furthermore, our findings provide valuable insights into parameter selection. The impact of the parameter ν on estimation error and non-zero coefficient identification, particularly concerning the ratio of observations to variables ($n/p_{\mathcal{A}}$), offers practical guidance. Optimal ν selection varies based on the dataset's characteristics, emphasizing the need for tailored parameter tuning in different scenarios.

The adaptability and performance of the ANP estimator position it as a compelling alternative to established methods, particularly in applications necessitating variable selection, both in high dimensional context and not. By offering a robust solution to high-dimensional data challenges, our approach promises advancements in fields where precision and data-driven insights are paramount.

On the other hand, while ANP is designed for high-dimensional settings, its performance might not be superior in all high-dimensional scenarios. It's essential to evaluate, for future progress, the specific characteristics of the dataset, such as the noise level and the true sparsity of the model, before applying ANP and possibly to adapt the best value of ν.

There are still numerous potential advancements concerning the function. Drawing insights from the findings derived through the examination or simulation of the ν effect, it becomes evident that exploring the link between the optimal parameter value and the ratio $n/p_{\mathcal{A}}$ would be crucial. However, given that the active set's cardinality is unknown, a plausible approach could involve investigating the relationship between ν_{opt} and the model's degrees of freedom.

The idea of an optimal selection of the parameter ν can (and probably must) be understood as the search for more optimal values: the advantage of adjusting the non-convexity of the penalty function along the path of the coefficients may entail the characteristic of determining different speeds of growth of the magnitude of the coefficients. While it may be plausible to think that coefficients that are activated earlier are the potential truly non-zero coefficients (and that those activated towards $\lambda = 0$ constitute noise), a slower rate of parameter explosion proportional to the decrease of the λ parameter may seem convenient. This can be translated into using more values of the parameter ν, whose values will be inversely proportional to the value of λ.

In addition, a primitive version of the `penalizedcdf` package can be downloaded on CRAN, which only allows (at the moment) the estimation of a penalised linear model using the ANP function. Currently, the computational demand, especially for very large datasets, can be a limitation.

In conclusion, our thesis underscores the merits of our proposed ANP estimator and elucidates its potential implications across various frameworks. Our methodology's adaptability, performance, and practical insights position it as a robust tool for high-dimensional data analysis and pave the way for future advancements in data-driven research.

Bibliography

Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.

Antoniadis, A. (1997). Wavelets in statistics: a review. *Journal of the Italian Statistical Society*, 6(2):97–130.

Augugliaro, L., Mineo, A. M., and Wit, E. C. (2013). Differential geometric least angle regression: a differential geometric approach to sparse generalized linear models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 75(3):471–498.

Bakin, S. et al. (1999). Adaptive regression and model selection in data mining problems.

Banerjee, O., El Ghaoui, L., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516.

Belloni, A., Chernozhukov, V., and Wang, L. (2011). Square-root lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806.

Berk, K. N. (1978). Comparing subset regression procedures. *Technometrics*, 20(1):1–6.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122.

Breaux, H. J. (1967). On stepwise multiple linear regression. Technical report, Army Ballistic Research Lab Aberdeen Proving Ground MD.

Breheny, P. and Huang, J. (2009). Penalized methods for bi-level variable selection. *Statistics and its interface*, 2(3):369.

Breheny, P. and Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The annals of applied statistics*, 5(1):232.

Breheny, P. and Huang, J. (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 25:173–187.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384.

Bühlmann, P. and Meier, L. (2008). Discussion: One-step sparse estimates in nonconcave penalized likelihood models. *The Annals of Statistics*, 36(4):1534–1541.

Candes, E. J., Wakin, M. B., and Boyd, S. P. (2008). Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14:877–905.

Cilluffo, G. (2018). *Induced Smoothing in LASSO Regression*. PhD thesis, Università degli Studi di Palermo, Palermo.

Dempster, A. P. (1972). Covariance selection. *Biometrics*, pages 157–175.

Derksen, S. and Keselman, H. J. (1992). Backward, forward and stepwise automated subset selection algorithms: Frequency of obtaining authentic and noise variables. *British Journal of Mathematical and Statistical Psychology*, 45(2):265–282.

Desboulets, L. D. D. (2018). A review on variable selection in regression analysis. *Econometrics*, 6(4):45.

Donoho, D. L. (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *Aide-Memoire of a Lecture at AMS Conference on Math Challenges of the 21st Century*.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of statistics*, 32(2):407–499.

Fan, J. and Fan, Y. (2008). High dimensional classification using features annealed independence rules. *Annals of statistics*, 36(6):2605–2637.

Fan, J., Feng, Y., and Wu, Y. (2009a). Network exploration via the adaptive lasso and scad penalties. *The annals of applied statistics*, 3(2):521.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.

Fan, J. and Li, R. (2006). Statistical challenges with high dimensionality: Feature selection in knowledge discovery. *arXiv preprint math/0602133*.

Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911.

Fan, J. and Lv, J. (2010). A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20(1):48.

Fan, J. and Lv, J. (2011). Nonconcave penalized likelihood with np-dimensionality. *IEEE Transactions on Information Theory*, 57(8):5467–5484.

Fan, J., Samworth, R., and Wu, Y. (2009b). Ultrahigh dimensional feature selection: beyond the linear model. *The Journal of Machine Learning Research*, 10:2013–2038.

Frank, L. E. and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135.

Friedman, J., Hastie, T., Hoefling, H., and Tibshirani, R. (2007). Pathwise coordinate optimization. *The Annals of Applied Statistics*, 2(1):302–332.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Friedman, J., Hastie, T., and Tibshirani, R. (2010a). A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*.

Friedman, J., Hastie, T., and Tibshirani, R. (2010b). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1.

Fu, W. and Knight, K. (2000). Asymptotics for lasso-type estimators. *The Annals of statistics*, 28(5):1356–1378.

Gabay, D. (1983). Chapter ix applications of the method of multipliers to variational inequalities. In *Studies in mathematics and its applications*, volume 15, pages 299–331. Elsevier.

Garside, M. (1965). The best subset in multiple regression model. *Applied statistics*, 14:196–200.

Gasso, G., Rakotomamonjy, A., and Canu, S. (2009). Recovering sparse signals with a certain family of nonconvex penalties and dc programming. *IEEE Transactions on Signal Processing*, 57(12):4686–4698.

Hall, P., Pittelkow, Y., and Ghosh, M. (2008). Theoretical measures of relative performance of classifiers for high dimensional data with small sample sizes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):159–173.

Hannan, E. J. and Quinn, B. G. (1979). The determination of the order of an autoregression. *Journal of the Royal Statistical Society: Series B (Methodological)*, 41(2):190–195.

Hendry, D. F. and Richard, J.-F. (1987). Recent developments in the theory of encompassing. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Hestenes, M. R. (1969). Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320.

Hocking, R. R. (1976). A biometrics invited paper. the analysis and selection of variables in linear regression. *Biometrics*, pages 1–49.

Højsgaard, S. and Lauritzen, S. L. (2008). Graphical gaussian models with edge and vertex symmetries. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):1005–1027.

Horst, R. and Thoai, N. V. (1999). Dc programming: overview. *Journal of Optimization Theory and Applications*, 103(1):1–43.

Huang, J., Breheny, P., and Ma, S. (2012). A selective review of group selection in high-dimensional models. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 27(4).

Huang, J., Ma, S., and Zhang, C.-H. (2008). Adaptive lasso for sparse high-dimensional regression models. *Statistica Sinica*, pages 1603–1618.

Huang, J. and Zhang, T. (2010). The benefit of group sparsity.

Hunter, D. R. and Li, R. (2005). Variable selection using mm algorithms. *Annals of statistics*, 33(4):1617.

Hurvich, C. M. and Tsai, C. (1990). The impact of model selection on inference in linear regression. *The American Statistician*, 44(3):214–217.

Jørgensen, B. (1987). Exponential dispersion models. *Journal of the Royal Statistical Society, Series B*, 49:127–162.

Ke, Z. T., Jin, J., and Fan, J. (2014). Covariate assisted screening and estimation. *The Annals of Statistics*, 42(6):2202–2242.

Kim, Y., Choi, H., and Oh, H.-S. (2008). Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association*, 103(484):1665–1673.

Lauritzen, S. L. (1996). Graphical models.

Lv, J. and Fan, Y. (2009). A unified approach to model selection and sparse recovery using regularized least squares. *The Annals of Statistics*, 37(6A):3498–3528.

Lv, J. and Liu, J. S. (2014). Model selection principles in misspecified models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):141–167.

Mallows, C. L. (1973). Some comments on cp. *Technometrics*, 15:661–75.

Mazumder, R., Friedman, J. H., and Hastie, T. (2011). Sparsenet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, 106(495):1125–1138.

McCullagh, P. and Nelder, J. (1989). Generalized linear models, second edition.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, 34(3):1436 – 1462.

Powell, M. J. (1969). A method for nonlinear constraints in minimization problems. *Optimization*, pages 283–298.

Pretis, F., Reade, J. J., and Sucarrat, G. (2018). Automated general-to-specific (GETS) regression modeling and indicator saturation for outliers and structural breaks. *Journal of Statistical Software*, 86(3):1–44.

Ranciati, S., Roverato, A., and Luati, A. (2021). Fused graphical lasso for brain networks with symmetries. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 70(5):1299–1322.

Sagiroglu, S. and Sinanc, D. (2013). Big data: A review. In *2013 international conference on collaboration technologies and systems (CTS)*, pages 42–47. IEEE.

Santos, C., Hendry, D. F., and Johansen, S. (2008). Automatic selection of indicators in a fully saturated regression. *Computational Statistics*, 23:317–335.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, pages 461–464.

Sur, P. and Candès, E. J. (2019). A modern maximum-likelihood theory for high-dimensional logistic regression. *Proceedings of the National Academy of Sciences*, 116(29):14516–14525.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Wang, H., Li, R., and Tsai, C.-L. (2007a). Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika*, 94(3):553–568.

Wang, L., Chen, G., and Li, H. (2007b). Group scad regression analysis for microarray time course gene expression data. *Bioinformatics*, 23(12):1486–1494.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35.

Zhang, C.-H. (2010a). Nearly unbiased variable selection under minimax concave penalty. *The Annals of statistics*, 38(2):894–942.

Zhang, T. (2010b). Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11(3).

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. *The Journal of Machine Learning Research*, 7:2541–2563.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.

Zou, H. and Li, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Annals of statistics*, 36(4):1509.