MONASH University

Bayesian Shrinkage Methods for Linear Regression

Shu Yu Tew

Doctor of Philosophy

A thesis submitted for the degree of Doctor of Philosophy at
Monash University in 2023
Clayton School of Information Technology

Copyright notice

I certify that I have made all reasonable efforts to secure copyright permissions for third-party content included in this thesis and have not knowingly added copyright content to my work without the owner's permission.

Abstract

Shrinkage methods that shrink statistical parameter estimates towards zero have gained popularity in the domain of high dimensional data analysis due to their ability to improve parameter estimation and potentially induce sparsity in the estimated model. This thesis explores the application of Bayesian shrinkage methods for linear regression, aiming to address two primary limitations: (1) the lack of an efficient general procedure to estimate the posterior mode for sparsity-inducing priors such as the horseshoe; and (2) the problem of selecting an appropriate prior for predictors with natural grouping structures in the absence of prior knowledge on the sparsity level of the problem.

The first main contribution of this thesis addresses limitation (1) with the introduction of a novel expectation-maximization (EM) procedure that solves for the *exact* posterior mode of Gaussian linear regression models that is applicable to a wide range of priors, including those with no closed-from density function (e.g., the horseshoe prior). This EM procedure presents for the first time, the posterior mode estimates of the horseshoe prior. We further demonstrated the application of this procedure to Bayesian ridge regression and Bayesian lasso regression, where empirical experiments and theoretical analyses suggest that it yields results that are on par with or, in some scenarios, superior to their state-of-the-art counterparts. In the specific case of applying the EM algorithm to Bayesian ridge regression, we improve the computational efficiency of the EM algorithm substantially by utilizing the singular value decomposition (SVD) representation of the predictor matrix.

The second contribution addresses limitation (2) with the introduction of an MCMC sampler to estimate the hyperparameters of the adaptive normal-beta prime prior and its application to grouped regression problems. Specifically, we extend the grouped half-Cauchy hierarchy to allow for varying sparsity levels within and across the groups of predictors. In experiments performed on simulated data, we show the strong performance of this estimator across regression problems with varying sparsity levels and signal-to-noise ratio strengths, making it a reasonable default estimator when there is no prior knowledge about the sparsity level of the problem.

Publications during enrolment

Publications included in this thesis:

1. Shu Yu Tew, Daniel F Schmidt, and Enes Makalic. Sparse horseshoe estimation via expectation-maximisation. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, September 19–23, 2022, Proceedings, Part V*, pages 123–139. Springer, 2023

2. Shu Yu Tew, Mario Boley, and Daniel F Schmidt. Bayes beats Cross Validation: Efficient and Accurate Ridge Regression via Expectation Maximization. *Advances in Neural Information Processing Systems (to appear)*, 2024

Thesis including published works declaration

I hereby declare that this thesis contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

This thesis includes two original papers published (or accepted) in top machine learning conference. The core theme of the thesis is shrinkage methods via Bayesian inference. The ideas, development and writing up of all the papers in the thesis were the principal responsibility of myself, the student, working within the Clayton School of Information Technology, Department of Data Science and AI under the supervision of A/Prof Daniel Schmidt and Dr Mario Boley.

The inclusion of co-authors reflects the fact that the work came from active collaboration between researchers and acknowledges input into team-based research.

In the case of *Chapters 3, 4 and 5,* my contribution to the work involved the following:

Thesis Chapter	**Publication Title**	**Status** *(published, in press, accepted or returned for revision, submitted)*	**Nature and % of student contribution**	**Co-author name(s) Nature and % of Co-author's contribution***	**Co-author(s), Monash student Y/N***
3 & 5	Sparse horseshoe estimation via expectation-maximisation	*Published*	85%. Concept, conducting experiments and writing the manuscript	1. Daniel Schmidt, Supervised study, concept and writing the manuscript. 10% 2. Enes Makalic, Concept, and input into manuscript. 5%	No No
4	Bayes beats Cross Validation: Efficient and Accurate Ridge Regression via Expectation Maximization.	*Accepted*	80%. Concept, conducting experiments and writing the manuscript	1. Mario Boley, Concept, conducting experiments and writing the manuscript. 10% 2. Daniel Schmidt, Supervised study, concept and writing manuscript. 10%	No No

I have renumbered sections of submitted or published papers in order to generate a consistent presentation within the thesis.

Student name:

Student signature: **Date:**

I hereby certify that the above declaration correctly reflects the nature and extent of the student's and co-authors' contributions to this work. In instances where I am not the responsible author I have consulted with the responsible author to agree on the respective contributions of the authors.

Main Supervisor name:

Main Supervisor signature: **Date:**

Acknowledgements

I would like to first extend my heartfelt gratitude to my supervisors A/Prof Daniel Schmidt and Dr Mario Boley for their invaluable contribution in making this thesis possible. I am extremely grateful for their unwavering support and expert guidance throughout my PhD journey. I appreciate the time they have dedicated and the weekends they selflessly sacrificed to improve the work in this thesis. Their commitment to helping me become a better researcher is truly commendable, and I sincerely thank them for consistently giving their best to assist me.

I thank Daniel for his mentorship, guiding me through my early days as a confused undergraduate student who had no idea what she was doing, to now a less confused PhD student developing state-of-the-art machine learning algorithms. Thank you for your tireless efforts and patience in teaching and making all our PhD meetings an interesting one.

I thank Mario for his persistent motivation and belief in my potential, which has greatly influenced my personal and academic growth. His commitment to guiding me toward being a better researcher has been a great source of inspiration, encouraging me to exceed my limits and recognize my abilities. I have learned so much under his guidance and there is still so much more to learn from him.

Furthermore, I want to thank my panel members, Professor Graham Farr, Professor Geoff Webb, and Dr Lan Du, for their insightful feedback and suggestions provided during my milestone reviews. My appreciation also goes to my co-author, Dr Enes Makalic, and the peer reviewers, whose constructive comments and valuable feedback significantly improved the quality of the publications in this thesis.

I would also like to express my appreciation to the FIT Graduate Research team for their prompt assistance with all application and administrative matters. Special thanks to Mrs Rachael Unwin, Ms May Chew, and Mrs Tracy Ozcagli for their exceptional support.

Most importantly, I am grateful for the financial support extended by my supervisors, the university, and the faculty. I would like to acknowledge the support I received from the Faculty Graduate Research Completion Award and NeurIPS 2023 Scholar Award. This research was also supported by the Australian Research Council (DP210100045).

Finally, to my family, friends, and loved ones, I am extremely grateful for your unconditional love and emotional support throughout my PhD journey. Thank you everyone for everything.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Shrinkage methods, also known as regularization methods, are statistical techniques that shrink parameter estimates toward zero. Owing to the abundance of high-dimensional data analysis problems, shrinkage methods have become increasingly popular as they can improve estimators' predictive performance, especially for large numbers of covariates, and potentially induce sparsity in the estimated model. This thesis examines the application of shrinkage methods to the estimation of linear models of the form:

$$\mathbf{y} = \beta_0 \mathbf{1}_n + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n), \tag{1.1}$$

where $\mathbf{y} = (y_1, \cdots, y_n)^T \in \mathbb{R}^n$ is the target variable, $\mathbf{X} \in \mathbb{R}^{n \times p}$ is the matrix of predictor variables, $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p) \in \mathbb{R}^p$ corresponds to the regression coefficients, $\beta_0 \in \mathbb{R}$ is the intercept, and $\boldsymbol{\epsilon}$ is a vector of i.i.d. normally distributed random error with mean zero and unknown variance σ^2. In this context, the parameter estimates to be shrunk towards zero are the regression coefficients. This is typically achieved by introducing a penalty term into the least-squares objective function of the form

$$\underset{\boldsymbol{\beta}}{\text{minimize}} \Big\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma f(\boldsymbol{\beta}) \Big\} \tag{1.2}$$

where γ is the regularisation parameter and $f(\boldsymbol{\beta})$ is the penalty term. Given a penalty choice and a set of data $\mathbf{y}$ and $\mathbf{X}$, the goal is to accurately estimate and (potentially) identify the non-zero components of the unknown regression coefficients, which, consequently, requires optimizing the value of γ that controls the amount of shrinkage applied to the coefficients. In particular, this thesis presents new estimators that perform simultaneous model parameter estimation and hyperparameter tuning within a Bayesian framework, including settings where the regression coefficient exhibits sparsity, or the predictors have a natural grouping structure.

Penalized regression (1.2) can utilize penalties of varying forms, with the most common penalty function being the l_q-norm with $f(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_q = (\sum_{j=1}^{p} |\beta_j|^q)^{1/q}$. This penalty can be further categorized into two classes: convex ($q \geq 1$) and non-convex ($0 < q < 1$) penalties. Well-known estimators with convex penalties include the lasso [165] ($q = 1$) and ridge estimators [86] ($q = 2$), while non-convex penalties include methods like the Smoothly Clipped Absolute Deviation (SCAD) [61] and the Minimax Concave Penalty (MCP) [190]. In this thesis, we explore penalized regression using Bayesian inference, a natural probabilistic framework for quantifying uncertainty and estimating unknown parameters. Within the Bayesian framework, most penalized regression methods can be effectively solved by interpreting the regression coefficients $\boldsymbol{\beta}$ as posterior point estimates based on carefully chosen shrinkage priors. For example, the lasso estimate [165] corresponds to the posterior mode estimates of $\boldsymbol{\beta}$ when they follow independent and identical double exponential (Laplace) priors. Several popular examples of Bayesian penalized regression methods include the Bayesian Lasso [78, 113, 134], the horseshoe estimator [42] and the normal-gamma estimator [39], each possessing unique properties, advantages and disadvantages.

This thesis explores continuous shrinkage priors that fall under the class of global-local shrinkage priors [140]. Within this class, each β_j is assigned a continuous shrinkage prior centered at $\beta_j = 0$ which can be elegantly represented using the scale mixture of normals (SMN) formulation:

$$\begin{aligned} \beta_j | \tau^2, \lambda_j^2, \sigma^2 &\sim N\left(0,\ \tau^2 \lambda_j^2 \sigma^2\right) \\ \lambda_j^2 &\sim \pi(\lambda_j^2) d\lambda^2 \\ \tau^2 &\sim \pi(\tau^2) d\tau^2 \end{aligned} \tag{1.3}$$

where τ is the *global* shrinkage parameter that controls the overall degree of shrinkage, λ_j is the *local* shrinkage parameter associated with the j-th predictor that controls the shrinkage for individual coefficients and $\pi(\cdot)$ is an appropriate prior distribution of choice assigned to the shrinkage parameters. Bayesian methods prove to be advantageous in this context, as they offer an automated approach to estimate the shrinkage parameters through probabilistic treatment of the model parameters. This enables the data to influence the estimation of τ^2 and λ_j^2 directly. Consequently, there is no need for manual tuning or separate calibration of these shrinkage parameters, making the model-fitting process more efficient.

1.1 Motivation

Within the class of global-local shrinkage priors, two priors stand out—the horseshoe prior [42] and the normal-beta prime (NBP) prior [19]. The horseshoe prior is generally considered to be an excellent default choice of sparsity-promoting prior that does not require additional hyperparameters. The NBP prior is an adaptive prior with tunable hyperparameters that control the origin and tail behavior of its density function, allowing it to model varying sparsity levels (i.e. dense or sparse). However, there is limited research addressing the application of this prior to grouped predictors.

The normal beta-prime prior is also a generalization of the horseshoe prior; that is, with the appropriate hyperparameter settings, the NBP reduces to the horseshoe prior. The horseshoe prior has been recognized as a good default prior choice for Bayesian sparse estimation [25, 42]. It exhibits a pole at $\beta_j = 0$, and heavy, Cauchy-like tails, both of which are desirable properties to have in sparse estimation because they allow small coefficients to be heavily shrunk towards zero while ensuring large coefficients are not over-shrunk. Despite these favorable properties, achieving sparse estimates using the horseshoe prior remains an open problem. While there exist samplers exploring the posterior means or medians of the horseshoe estimator, these estimates themselves are not sparse. Sparse estimates can be obtained by instead considering the posterior mode, i.e., maximum a posteriori (MAP) estimation [13, 28, 62]. However, obtaining analytical solutions for this problem remains challenging because the horseshoe prior lacks an analytical form. Alternatively, simple thresholding ("sparsification") rules for the posterior mean or the posterior median of the coefficients can be used to produce sparse estimates, but these methods tend to lack theoretical justification, and inference can be highly sensitive to the choice of the threshold [42, 104].

To summarise, this thesis addresses the following research problems:

1. **Exact posterior mode estimation**: Finding the exact posterior mode estimates efficiently, particularly in cases where priors lack closed-form density functions can be challenging. This thesis introduces a novel expectation-maximization (EM) algorithm to address this problem, making it possible, for the first time, to explore exact posterior mode estimates for priors without analytic forms (e.g. the horseshoe prior).

2. **Sparsity adaptation and grouping**: Choosing an appropriate prior can be challenging when there is no prior knowledge about the sparsity level of the problem, especially in situations involving grouped predictors. This thesis extends the adaptive normal-beta prime (NBP) prior to grouped regression problems. Specifically, we extend the grouped half-Cauchy hierarchy of Xu et al. [185] by assigning an NBP prior to both the local and group shrinkage hyperparameters, allowing for varying sparsity levels within and across group predictors. An efficient Metropolis-Hasting based sampler is introduced to estimate the NBP hyperparameters.

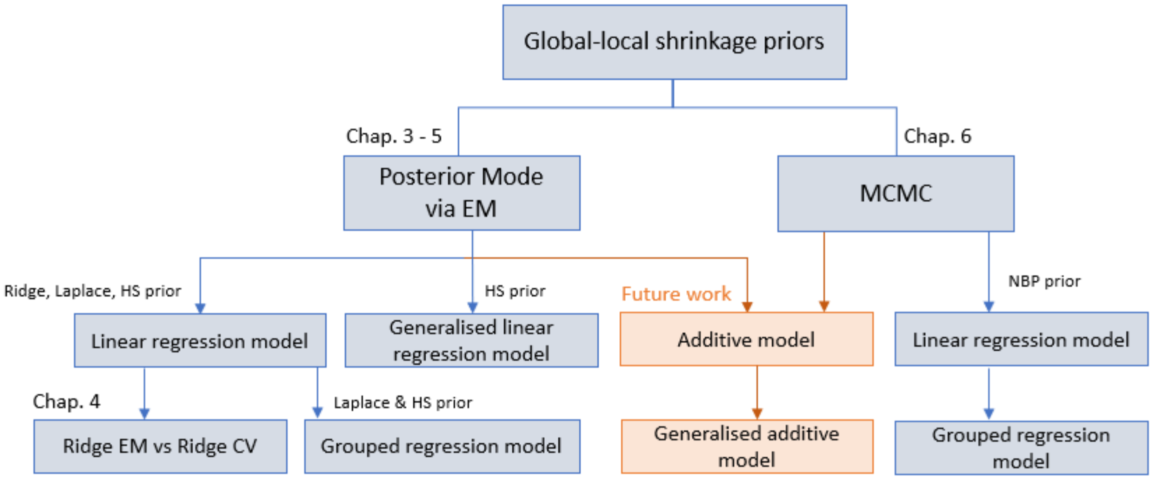


FIGURE 1.1: High-level overview of the research outcome.

1.2 Contributions

This Ph.D. research contributes to the understanding, applicability, and robustness of Bayesian penalized regression methods. More formally, the key contributions are as follows:

1. Proposed a novel expectation-maximization (EM) procedure to solve for the *exact* posterior mode of Gaussian linear regression models that is applicable to a wide range of priors including those with no closed-from density. We also extended this procedure to generalized linear models.

2. Applied the proposed EM procedure to the horseshoe and Laplace prior, resulting in a novel class of sparse estimators based on the Bayesian horseshoe and Bayesian lasso, with extensions to grouped regression.

3. Applied the proposed EM procedure to Bayesian ridge regression, and improved the computational efficiency using the singular value decomposition (SVD) representation of the predictor matrix **X**. This results in a novel method for tuning the regularization parameter of ridge regression that is faster than leave-one-out cross-validation (LOOCV). As a supplementary outcome, we also presented a faster implementation of the LOOCV ridge regression risk using the same SVD technique, outperforming existing implementations, including the popular `scikit-learn` python library, by approximately a factor of two.

4. Provided a finite sample bound to guarantee the unimodality of the posterior distribution within the Bayesian ridge regression model. This guarantees the convergence of iterative posterior optimization procedures like the proposed EM algorithm to a unique optimal solution, for a large enough sample size, under relatively mild conditions.

5. Derived an efficient MCMC sampler for the hyperparameters in a normal beta-prime (NBP) prior. This sampler estimates the hyperparameters for automatic adaptivity to the sparsity of the problem.

6. Extended the grouped half-Cauchy model of Xu et al. [185] to an adaptive grouped NBP model. Additionally, provided an alternative and concise interpretation of the interaction between the local and grouped shrinkage parameters using the log-scale representation of the shrinkage hyperparameters.

From a high-level perspective, this thesis presents two main research outputs (as illustrated in Figure 1.1): an EM procedure for posterior mode estimation and an MCMC sampler for estimating hyperparameters of the normal-beta prime prior. These methods are widely applicable to a wide range of global-local shrinkage priors with minimal modifications. In this thesis, we primarily focus on applying these methods to popular global-local shrinkage priors, including the horseshoe, Laplace, and ridge prior for the EM procedure and the normal-beta prime prior for the MCMC sampler. These methods have been implemented within the context of linear regression models, with extensions to include grouped regression and generalized linear regression models.

1.3 Structure of the Thesis

This thesis is structured as follows:

Chapter 2 reviews the relevant literature on penalized linear regression, its Bayesian counterparts, and other Bayesian shrinkage methods. We also explore a range of existing approaches for approximating posterior distributions, such as Monte Carlo sampling methods and non-sampling distribution approximation techniques.

Chapter 3 introduces the novel expectation-maximization (EM) procedure to solve for the *exact* posterior mode of the Gaussian linear regression model. This method remains applicable even when dealing with priors lacking an analytical form. This chapter also outlines two alternative extensions of the procedure to generalized linear models.

Chapter 4 presents the EM implementation of the Bayesian ridge regression with a comparative assessment against the frequentist version of ridge regression using LOOCV for hyperparameter tuning. This evaluation includes parameter estimation accuracy and computational complexity, considering both theoretical and empirical aspects.

Chapter 5 outlines the implementation of the proposed EM procedure with the horseshoe and Laplace prior. The resulting estimators are evaluated for their effectiveness in parameter estimation and variable selection, comparing them against similar estimators of existing work using both synthetic and real datasets. This chapter also presents the extension of the proposed estimators to sparse grouped regression and explores different parameterizations of the shrinkage parameter.

Chapter 6 presents a gradient-assisted Metropolis-Hastings algorithm to estimate the hyperparameters in the NBP model. This procedure is integrated into a Gibbs sampler for fitting adaptive Bayesian regression models with an extension to grouped regression. This chapter includes empirical experiments that consistently demonstrate the strong performance of the proposed adaptive regression method across varying levels of sparsity and signal-to-noise ratio.

Chapter 7 summarizes the work accomplished, discusses the limitations inherent in this research, and outlines potential avenues for future research.

Chapter 2

Literature Review

In the domain of shrinkage methods, their inherent link to sparsity becomes apparent, especially in the context of high-dimensional regression problems. Despite the widespread prevalence of the terms "sparsity" and "shrinkage", it is important to define these terms before advancing into the subsequent chapters. Sparsity refers to a scenario in which a proportion of parameter estimates assume zero values, indicating the absence of certain effects or relationships. Notably, this term often aligns with "variable selection" for which it only involves the identification of variables without providing estimates, while sparsity encompasses both variable selection and parameter estimation. Conversely, shrinkage estimation is an estimation strategy where the parameters are constrained toward zero, but they may not necessarily reach an absolute zero value.

The scope of this research revolves around the application of shrinkage methods to the linear regression problem

$$\mathbf{y} = \beta_0 \mathbf{1}_n + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n) \tag{2.1}$$

where $\mathbf{y} = (y_1, \cdots, y_n)^T \in \mathbb{R}^n$ is a vector of outcome variable, $\mathbf{X} \in \mathbb{R}^{n \times p}$ is a matrix of predictor variables, $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p) \in \mathbb{R}^p$ corresponds to the regression coefficients, $\beta_0 \in \mathbb{R}$ is the intercept, and $\boldsymbol{\epsilon}$ is a vector of i.i.d. normally distributed random error with mean zero and unknown variance σ^2. As is standard in linear regression, and without any loss of generality, the predictors are assumed to be standardized with a mean of zero and a standard deviation of one, and the target has a mean of zero, i.e., the estimate of the intercept is simply $\hat{\beta}_0 = (1/n) \sum y_i$. This allows the exclusion of the intercept term when estimating the remaining coefficients $\boldsymbol{\beta}$. Given $\mathbf{y}$ and $\mathbf{X}$, the goal is to accurately identify and estimate the non-zero components of the unknown regression coefficients $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p) \in \mathbb{R}^p$.

This problem statement lays the groundwork for a systematic exploration of existing penalized regression methods. Section 2.1 presents a comprehensive overview of several well-known penalized regression methods from the perspective of frequentist statistics. Correspondingly, in Section 2.2, we discuss equivalent penalized regression techniques within the Bayesian framework. In the latter part of this section, the concept of hierarchical priors is introduced, outlining the fundamental characteristics of a wide range of hierarchical representations for Bayesian sparse and shrinkage estimators. Section 2.3 explores methods to compute the posterior distributions effectively with the specific goal of facilitating posterior inference.

2.1 Penalized Regression

The main issue with regular regression techniques such as the ordinary least squares (OLS) is that they quickly lead to overfitting as the ratio of p predictor variables to n observations increases [173]. In OLS, the regression coefficients are estimated by minimizing the squared error between the observed data and the predicted outcome:

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = \underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \Big\} \tag{2.2}$$

where $\mathbf{y} = (y_1, \cdots, y_n)$ is the n-dimensional vector representing the values of the outcome variable, $\boldsymbol{X}$ is the $(n \times p)$ matrix of our observations with n data points and p predictor variables, and $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p)$ is the $(p \times 1)$ vector of regression coefficient of interest. When $\mathbf{X}$ is a full rank matrix, the OLS solution of (2.2) can be analytically calculated as:

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}. \tag{2.3}$$

However, if $\mathbf{X}$ is not full rank, meaning that $p > n$ or there are highly correlated variables in the data matrix, $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ is not positive definite, and the inverse does not exist. Consequently, the OLS solution cannot be uniquely determined. As such, a constraint or regularisation of the estimation process is required, which leads us to the introduction of penalized regression. The penalized regression technique adds a penalty term to our regression model with the intention to shrink small coefficient towards zero while leaving large coefficient relatively large, i.e.,

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma f(\boldsymbol{\beta}) \Big\} \tag{2.4}$$

where $\|\cdot\|_2^2$ is the squared l_2 norm with $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$, γ is the regularisation parameter, and the choice of the function $f(\boldsymbol{\beta})$ determines the type of penalty applied to the model. The regularisation parameter γ determines the strength of the shrinkage/penalization. As γ increases, the shrinkage effect on the estimated coefficients will be stronger and all the coefficient estimates, $\boldsymbol{\beta}$ will converge to 0 as γ approaches ∞; when $\gamma = 0$, it recovers to the ordinary least squares solution.

Table 2.1: Overview of the key features for the corresponding penalized regression approaches.

Method	Unbiasedness	Sparsity	Continuity	Convex	Closed-form
Lasso	✗	✓	✗	✓	✗
Ridge	✗	✗	✗	✓	✓
Bridge ($q > 0$)	✓, $q < 1$ ✗, $q \geq 1$	✓, $q \leq 1$ ✗, $q > 1$	✗	✓, $q \geq 1$ ✗, $q < 1$	✗
Elastic Net	✗	✓	✗	✓	✗
Group Lasso	✗	✓	✗	✓	✗
Sparse Group Lasso	✗	✓	✗	✓	✗
Adaptive Lasso	✗	✓	✗	✓	✗
SCAD	✓	✓	✓	✗	✗
MCP	✓	✓	✓	✗	✗

According to Fan and Li [61], an ideal penalty function should lead to an estimator with the following three key properties:

1. Unbiasedness: The resulting estimator should exhibit near-unbiasedness when the true unknown parameter is large, avoiding unnecessary modeling bias.
2. Sparsity: The resulting estimator should act as a thresholding rule, automatically setting small estimated coefficients to zero, reducing model complexity through sparsity.
3. Continuity: The resulting estimator should be continuous, indicating that the solution changes continuously with variations in the penalty parameter γ, ensuring stability in model prediction.

The most common penalty function is the l_q norm (also known as l_q penalty). However, none of the l_q penalties satisfies all three conditions simultaneously. l_q regularized regression only yields sparse solutions for $q \leq 1$. Although penalties with $q > 1$ do not produce sparse solutions, they have the advantage of convexity, for which the optimization problem is much simpler as compared to when dealing with non-convex penalties ($q < 1$). Consequently, convex l_q penalties with $q > 1$ fail to satisfy the sparsity condition and all convex penalties are unable to achieve unbiased solutions. To overcome these limitations, alternative nonconvex penalties such as the smoothly clipped absolute deviation (SCAD) and minimax concave penalty (MCP) were introduced. We will discuss these non-convex methods briefly in Section 2.1.2. Table 2.2 summarizes the penalty functions for each of the discussed penalties, while Table 2.1 provides an overview of their key features. Both tables draw inspiration from Emmert-Streib and Dehmer [60, Tab.1 & Tab.2]. In the next few subsections, we will explore popular examples of both convex and non-convex penalties.

Table 2.2: Penalty terms in penalized regression methods, with reference to their corresponding Bayesian counterparts.

Method	Penalty function $f(\boldsymbol{\beta})$	Bayesian
Lasso	l_1 penalty: $\|\|\boldsymbol{\beta}\|\|_1$	[78, 134]
Ridge	l_2 penalty: $\|\|\boldsymbol{\beta}\|\|_2^2$	[89]
Bridge	l_q penalty: $\sum_{j=1}^{p} \|\beta_j\|^q$, $q > 0$	[115, 143]
Elastic Net	$l_1 + l_2$ penalty: $(1-\alpha)\|\|\boldsymbol{\beta}\|\|_1 + \alpha\|\|\boldsymbol{\beta}\|\|_2^2$	[6, 108]
Group Lasso	$\sum_{g=1}^{G} \sqrt{p_g}\|\|\boldsymbol{\beta}_g\|\|_2$	[43, 145, 184]
Sparse Group Lasso	$\alpha\|\|\boldsymbol{\beta}\|\|_1 + (1-\alpha)\sum_{g=1}^{G} \sqrt{p_g}\|\|\boldsymbol{\beta}_g\|\|_2$	[48, 184]
Adaptive Lasso	weighted l_1 penalty: $\sum_{j=1}^{p} \omega_j\|\beta_j\|$	[5, 103]
SCAD	(2.14)	[1, 105]
MCP	(2.15)	

2.1.1 Convex Penalties

An advantage of convex penalties is that they have a unique optimal solution. Convex functions cannot possess local minima that are not also global minima [81]. This suggests that given an appropriate optimization algorithm and a convex penalty, it is possible to attain a global optimal solution, making the optimization process simpler compared to dealing with a non-convex penalty. In this subsection, we take a closer look at two of the more prominent class of convex penalty namely the Lasso and the Ridge.

2.1.1.1 Lasso Penalty

The lasso is a well-known convex penalty regression that provides sparse estimates through the l_1-constrained least squares, when $q = 1$:

$$\hat{\boldsymbol{\beta}}^{\text{lasso}} = \underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma\|\boldsymbol{\beta}\|_1\Big\}. \tag{2.5}$$

where $\|\cdot\|_1$ is the l_1-norm and $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p}|\beta_j|$. This l_1-penalty is unique among the convex penalties because it is the only one capable of producing sparse solutions, resulting in estimates with a few non-zero entries in β. This property makes it a popular technique for simultaneous estimation and variable selection. Such a property cannot be observed in any l_q penalty with $q > 1$ and for $q < 1$, while the solutions are sparse, the penalties become non-convex functions, posing challenges in dealing with them. The characteristics of these non-convex penalties will be further explained in the next subsection 2.1.2. However, there are a few disadvantages associated with the Lasso penalty:

- The Lasso lacks a closed-form solution because its objective function (2.5) is non-differentiable, making both computational and theoretical aspects relatively more challenging. However, closed-form solutions are possible for the special case of an orthonormal design matrix $\mathbf{X}$:

$$\hat{\beta}_j^{\text{lasso}} = \text{sgn}\left(\hat{\beta}_j^{\text{OLS}}\right)\left(\left|\hat{\beta}_j^{\text{OLS}}\right| - \gamma\right)_+ \tag{2.6}$$

where $(x)_+ = \max\{x, 0\}$.

- The Lasso penalty is not guaranteed to be consistent in terms of variable selection. There are proofs that suggest the Lasso produces inconsistent variable selection results under certain circumstances [121, 192].
- The estimates produced by the Lasso after shrinkage are biased towards zero. This bias is more significant for large regression coefficients because Lasso penalizes all coefficients with the same weight towards zero and large coefficients tend to get penalized heavily. As a consequence, unimportant variables may be penalized to zero, but the coefficients of important variables are also penalized with the same weight.
- When a group of predictor variables are highly correlated, the Lasso often selects only one predictor from that group and this selection is at random, resulting in a higher prediction error when compared with the Ridge regression.

These disadvantages (especially the first two mentioned) motivate the proposal of non-convex penalty functions. Non-convex penalties can reduce the shrinkage of large coefficients and produce less biased parameter estimates. Many variants of the Lasso penalty [192, 193] have been developed to deal with the drawbacks of the Lasso. These variants inherit the two essential features of the standard lasso which is to perform simultaneous estimation and variable selection, while also offering additional properties that the standard Lasso penalty cannot achieve. Below are some examples of the generalizations of the Lasso penalty:

- **Group Lasso**

  The group lasso [189] is designed to handle regression problems with covariates having a natural group structure, aiming to simultaneously set all coefficients within a group to either zero or non-zero [168]. This approach addresses scenarios where predictor variables are divided into G distinct groups such as gene pathways in gene expression data or factor level indicators in categorical data. Instead of focusing solely on individual variables, the objective extends to selecting non-zero coefficients at the group level. The group lasso achieves this by solving the following convex optimization problem:

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\left\{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma \sum_{g=1}^{G} \sqrt{p_g}\|\boldsymbol{\beta}_g\|_2\right\}. \tag{2.7}$$

  where p_g is the number of variables in group g and $\|\cdot\|_2$ is the l_2 norm (not squared). When the size of each group is set to 1 ($G = p$), this procedure effectively recovers the regular lasso solution. It can be seen as applying the lasso at the group level, where instead of selecting individual variables to be zero, entire groups of variables are chosen or excluded together. This implies that if any variable within a group is included in the model, then all coefficients within that group will be non-zero. However, the group lasso procedure is limited to inducing sparsity at the group level and cannot achieve sparsity within groups. In response to this constraint, Simon et al. [158] introduced the sparse group lasso, which is a convex combination of the lasso and group lasso penalties to this group regression problem:

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\gamma\Big(\alpha\|\boldsymbol{\beta}\|_1+(1-\alpha)\sum_{g=1}^{G}\sqrt{p_g}\|\boldsymbol{\beta}_g\|_2\Big)\Big\}. \tag{2.8}$$

where $\alpha \in [0,1]$. When $\alpha = 0$, the group lasso is achieved and when $\alpha = 1$, we recover the regular lasso.

- **Adaptive Lasso**

  The adaptive lasso [192] is a modification of the standard lasso model, introduced to achieve oracle properties i.e. identify the true non-zero coefficients (variable selection consistency) as well as achieve the optimal estimation rate (estimation consistency). The key idea behind the adaptive lasso is to introduce weights ω_j for the penalty on each regression coefficient β_j such that:

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\gamma\sum_{j=1}^{p}\omega_j|\beta_j|\Big\}. \tag{2.9}$$

  The OLS estimates of the regression coefficients, denoted as $\hat{\boldsymbol{\beta}}^{\text{OLS}}$, can be used as the adaptive weights, with each weight given by $\omega_j = 1/|\hat{\beta}_j^{\text{OLS}}|$. However, it is also worth noting that users have the flexibility to supply their own weights. For instance, in scenarios with high-dimensional regression problems, where the number of predictors p is greater than the number of observations n, one might opt to use the ridge regression estimates as the adaptive weights instead of the OLS coefficients. This adaptability allows for a more tailored and robust approach in different situations.

It's important to acknowledge that there are numerous other Lasso variants beyond the ones covered here. While they are interesting and have made significant contributions to the literature, I have chosen not to discuss them extensively in this section due to page limitations and their limited relevance to our specific study. The vast array of these variants, including fused lasso [167]. group fused lasso [4] and relaxed lasso [120], continues to expand, and it is likely that new ones will emerge in the future.

2.1.1.2 Ridge Penalty

In the case where $q = 2$, the solution to the penalized minimization problem for equation (2.4) leads to ridge penalized regression, which involves the application of l_2 regularization, as expressed by the following equations:

$$\begin{aligned}\hat{\boldsymbol{\beta}}^{\text{ridge}} &= \underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma\sum_{j=1}^{p}(\beta_j)^2\Big\} \\ &= (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \gamma\mathbf{I_p})^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}, \qquad (2.10)\end{aligned}$$

whereby (2.10) is the closed-form solution for the ridge regression. Notably, ridge penalized regression does not yield a sparse solution; instead, all ridge estimates are non-zero and subject to the same magnitude penalty, which is determined by the regularization parameter γ. The ridge penalty, also characterized as the squared l_2 norm, measures the squared distance between the coefficient vector $\boldsymbol{\beta}$ and zero, defined as $d(\boldsymbol{\beta}, 0) = \sqrt{\sum_{j=1}^{p}(\beta_j)^2}$. Minimizing this term encourages solutions where the coefficients are brought closer to zero, thereby effectively shrinking the coefficients to mitigate overfitting. Ridge regression is driven by the goal of handling multicollinearity in the data, particularly when $p > n$, and its advantageous outcome of generating more accurate prediction estimates with lower mean squared error compared to a regular multiple linear regression model within this context [86]. Although ridge regression cannot perform variable selection like its counterpart, Lasso regression, it remains advantageous in various practical applications. Notably, in situations where $n > p$ and predictors are highly correlated, ridge regression often outperforms Lasso regression [193]. Ridge regression also holds a particular advantage in cases where all the underlying true coefficients exhibit approximately the same order of magnitude, as this assumption aligns with the fundamental basis of its predictions.

2.1.1.3 Elastic Net

The Elastic Net [193] combines the Lasso (l_1 penalty) and the Ridge penalty (l_2 penalty) to create a hybrid regularization approach. It aims to minimize the objective function:

$$\hat{\boldsymbol{\beta}}^{\text{EN}} = \underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma\Big((1-\alpha)\|\boldsymbol{\beta}\|_1 + \alpha\|\boldsymbol{\beta}\|_2^2\Big)\Big\}. \qquad (2.11)$$

where α is a mixing parameter that controls the balance between the l_1 and l_2 penalties. For $\alpha = 1$, the Lasso penalty is obtained, while for $\alpha = 0$, we recover the Ridge regression penalty. The Elastic Net benefits from the complementary advantages of both penalties: variable selection (from the Lasso) and group selection (from the Ridge). In contrast to the Lasso, which may arbitrarily select one variable from a group of highly correlated predictors, the Elastic Net can identify correlations among predictors and potentially include all correlated variables in the model.

Tibshirani et al. [168] demonstrated that the coefficients of highly correlated variables produced by the Elastic Net are approximately equal and are selected together as a group.

2.1.1.4 Bridge Penalty

Bridge regression minimizes the residual sum of square (RSS) subject to a constraint determined by the parameter q:

$$\hat{\boldsymbol{\beta}}^{\text{Bridge}} = \underset{\boldsymbol{\beta}}{\text{minimize}} \Big\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \gamma \sum_{j=1}^{p} |\beta_j|^q \Big\}. \tag{2.12}$$

This regularisation term has the form of l_q-norm, similar to the one presented in equation (2.4), with $q > 0$. Notably, setting $q = 1$ leads to the Lasso penalty, and $q = 2$ results in ridge regression. An interesting fact pointed out by Emmert-Streib and Dehmer [60] is that Bridge regression [63] was introduced prior to the Lasso in 1993. However, it did not gain significant attention and recognition at that time because the model was not extensively studied. It was only later that Tibshirani [165] presented a comprehensive analysis of the Lasso in 1996, establishing it as a groundbreaking and influential method. This analysis contributed significantly to the Lasso's widespread adoption and its subsequent position as a dominant regularization technique in the field of statistics and machine learning.

2.1.2 Non-Convex Penalty

In this section, we will explore nonconvex penalties, specifically SCAD, MCP, and l_q penalties when $0 \leq q < 1$. These penalties are all singular at the origin and they encourage sparsity. This property is necessary when the penalty function is employed for variable selection and reducing model complexity for enhanced interpretability. However, if the focus is solely on regularization without variable selection, this singularity can give rise to theoretical and computational challenges. For instance, the l_0 penalized regression is a popular best subset selection method that involves fitting all 2^p possible models and selecting the best model based on a selection criterion (e.g., AIC, BIC, adjusted R-squared). This approach becomes computationally expensive and infeasible for high-dimensional data as the complexity increases exponentially with the number of predictors, p. The l_0 regularization problem is also known to be NP-hard [129] and computational methods based on exhaustive search rapidly become infeasible when p increases.

SCAD and MCP are designed to overcome some shortcomings of the lasso penalty, particularly the bias introduced when the true parameter values are large. Both SCAD and MCP minimize the RSS subject to a piecewise penalty function, which applies different penalty magnitudes for varying coefficient magnitude values:

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\Big\{\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\sum_{j=1}^{p}p_\gamma(|\beta_j|)\Big\}, \tag{2.13}$$

where $p_\gamma(|\beta_j|)$ is the penalty function. The SCAD penalty is defined as

$$p_\gamma^{\mathrm{SCAD}}(|\beta_j|)=\begin{cases}\gamma|\beta_j| & \text{if } 0\leq|\beta_j|<\gamma,\\ \frac{(a^2-1)\gamma^2-(|\beta_j|-a\gamma)^2}{2(a-1)} & \text{if } \gamma\leq|\beta_j|<a\gamma,\\ \frac{(a+1)\gamma^2}{2} & \text{if } |\beta_j|\geq a\gamma,\end{cases} \tag{2.14}$$

for $a>2$. The recommended setting is $a=3.7$ following the simulation results provided by Fan and Li [61], suggesting it to be approximately optimal. The SCAD penalty function is characterized by three regions: lasso penalty until $|\beta_j|=\gamma$, followed by a smooth transition to a quadratic penalty until $|\beta_j|=a\gamma$, and a constant penalty for $|\beta_j|>a\gamma$.

Similarly, the MCP penalty function also employs a piecewise approach:

$$p_\gamma^{\mathrm{MCP}}(|\beta_j|)=\begin{cases}\gamma|\beta_j|-\frac{\beta_j^2}{2a} & \text{if } 0\leq|\beta_j|\leq a\gamma,\\ \frac{a\gamma^2}{2} & \text{if } |\beta_j|>a\gamma,\end{cases} \tag{2.15}$$

for $a>0$, it initially applies a linear penalty similar to the lasso before smoothly decreasing the penalization rate as the absolute value of the coefficient increases, ultimately reaching zero for $|\beta_j|>a\gamma$.

Hence, both SCAD and MCP can be viewed as modifications of the Lasso penalty, offering less shrinkage for large coefficient estimates and addressing the problem of potential inconsistency of the Lasso penalty. They start with a similar rate of penalization as the lasso for small coefficients but smoothly reduce the penalty for larger coefficients, resulting in less sparsity when necessary. Despite having desirable characteristics that satisfy the oracle property defined by Fan and Li [61], penalties with non-convex functions have multiple optimal solutions, making the optimization process computationally challenging. This introduces complexities in finding the global minimum, making the convergence process slower and requiring specialized optimization algorithms. However, despite these computational challenges, SCAD and MCP have gained popularity due to their superior performance in promoting sparsity and handling large coefficient values in various statistical modeling scenarios.

2.1.3 Regularization Parameter Tuning Methods

Hyperparameter tuning is a fundamental aspect of building effective machine learning models. In the domain of penalized regression methods, finding the optimal regularization parameter γ is of paramount importance. The regularization parameter plays a crucial role in controlling the level of regularization imposed on the model, balancing the trade-off between model complexity and data fitting. Three common approaches for hyperparameter tuning include:

1. **Grid Search**
   Grid search requires the specification of a set of values for each hyperparameter and exhaustively evaluate all possible values given in the set. However, grid search suffers from the curse of dimensionality, because the number of every possible hyperparameter combination values grows exponentially with the number of hyperparameters and the grid size. Nonetheless, in low dimensional space (e.g. 1-d or 2-d), grid search remains a reliable and efficient choice [23]. Typically, the grid is defined with values evenly spaced in log scale between a specified upper and lower bounds of the hyperparameter.

2. **Random Search**
   In its simplest form, random search involves independent random sampling of the hyperparameters from a pre-specified (often uniform) distribution. Bergstra and Bengio [23] have demonstrated that random search retains the practical advantages of grid search (conceptual simplicity, ease of implementation and trivial parallelism) while providing improved efficiency, particularly in high-dimensional search spaces, albeit with a minor reduction in efficiency in low-dimensional spaces.

3. **Line Search**
   This approach involves the use of an iterative numerical optimization algorithm to minimize a specific loss or risk function for determining the optimal hyperparameter. Examples of such algorithms include gradient descent algorithms, simplex optimization [131], and constrained optimization by linear approximation [144, 180] among others.

In cases where the optimization function exhibits multiple bad local minima, grid-based search algorithms are advisable. Conversely, if attaining a local minimum suffices, or the optimization problem is straightforward and unimodal, line search approaches gradient descent algorithm, offer superior time efficiency. The application of grid search algorithms are often guided by a number of performance metrics, usually measured by cross-validation (CV) on the training data. In the following subsection, we discuss the application of CV in hyperparameter tuning and its relevance to ridge regression.

2.1.3.1 Cross-Validation (CV)

Cross-validation is arguably the most common method used along side with grid search for regularization tuning, with **k-fold cross-validation** being particularly prevalent. Here, the data is divided into k subsets, with $k \leq n$, and the model is trained and evaluated k times, each time using a different subset as the validation set while the remaining subsets form the training data. Subsequently, the performance of the model is averaged across all k folds, and the γ value that yields the best average performance is selected as the final hyperparameter.

A crucial consideration in k-fold cross-validation is the choice of the value of k. A small k may result in higher variance (inconsistent) estimates [38], as different runs can lead to varying optimal γ values being chosen. On the other hand, selecting a large k provides more consistent estimates but can become computationally inefficient, especially when dealing with large datasets, n and a large number of features, p. While the general suggestion of Kohavi et al. [100] to use 10-fold CV has been widely accepted, such a choice may suffer from high bias in high-dimensional problems. Alternatively, some researchers propose that the optimal k value should be problem-dependent, taking into account factors like sampler size, signal-to-noise ratio (SNR), and the specific framework of the problem (e.g., classification, regression, or density estimation). Another approach involves treating k as a hyperparameter and estimating it adaptively based on the data to achieve an appropriate choice of the cross-validation configuration [11]. Nonetheless, further research is needed to thoroughly investigate the performance and theoretical foundations of these methods to ascertain the superiority of these approaches for hyperparameter tuning.

An alternative variant is the **leave-one-out cross-validation (LOOCV)**, where each data point is used as a validation set once, effectively setting $k = n$. LOOCV mitigates bias issues (i.e. is nearly unbiased [177]) but is computationally expensive since it necessitates n model fits. Li [107] established that under specific conditions, LOOCV is consistent and asymptotically optimal, suggesting that LOOCV provides estimates that perform as well as if one knew the true values with probability that converges to 1, as $n \to \infty$. More recently, Xu et al. [182] proved the consistency of the LOOCV under moderately high dimensional asymptotic scenarios, where $n, p \to \infty$ with $n/p \to \delta > 1$. However, despite its consistency in estimation, Shao [157] showed that LOOCV is asymptotically inconsistent in model selection. In other words, as $n \to \infty$, the probability of LOOCV selecting the model with the best predictive ability does not converge to one as $n \to \infty$.

Although there have been recent advancements in the theoretical understanding of LOOCV and its various approximations in high-dimensional settings [45, 102, 119, 125, 133, 175, 178, 181, 182], there remains a need for further research on these methods, particularly regarding the statistical properties of the tuned estimators under general distributional assumptions, as highlighted by Patil et al. [135].

Application to Ridge Regression For ridge regression, the LOOCV method can be efficiently computed due to the existence of a closed-form solution for the optimal γ under the linear regression setting. This implies that there is no need to fit the model n times, making LOOCV computationally efficient and a popular choice in practice. Moreover, unless $p/n \to 1$, the LOOCV ridge regression risk as a function of γ converges uniformly (almost surely) to the true risk function on $[0, \infty)$ and therefore optimizing it consistently estimates the optimal γ [82, 135]. However, for finite n, the LOOCV risk can be multimodal and, even worse, there can exist local minima that are almost as bad as the worst γ [161]. Therefore, iterative algorithms like gradient descent cannot be reliably used for the optimization, giving theoretical justification for the pre-dominant approach of optimizing over a finite grid of candidates $G = (\gamma_1, \dots, \gamma_l)$. In Chapter 4, we examine this fast LOOCV estimate for γ and compare its performance to our proposed hyperparameter tuning method for ridge regression. It is worth mentioning that although there are faster approximation methods like Generalized Cross-Validation (GCV) [73] for ridge regression, which provide a convenient approximation to LOOCV under squared-error loss, we focus on the exact LOOCV score in this thesis.

2.2 Bayesian Penalised Regression

Penalized regression introduces shrinkage and sparsity by including a penalty term. In Bayesian learning, the type (degree) of shrinkage applied to the estimated parameters is predominantly influenced by the choice of a shrinkage prior. Prior distributions and posterior distributions are the core of Bayesian inference. The prior distribution encapsulates the initial beliefs or knowledge about the parameter of interest before incorporating any observed data. This distribution shapes characteristics of the ensuing posterior distribution, which in turn influences the ultimate conclusions drawn from the inference process; the posterior distribution, on the other hand, is the updated distribution of the parameter of interest after incorporating the observed data. It is derived by combining the likelihood of the data given the parameter and the prior distribution of the parameter. The posterior distribution represents the new state of knowledge about the parameter after considering the data. Leveraging the posterior distribution, a plethora of analytical possibilities come into play. These include, but not limited to, parameter point estimation, the quantification of uncertainty via credible intervals, and model selection.

Consider the standard linear regression model, defined by Equation (2.1), which we repeat here for convenience:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{2.16}$$

whereby we assume a Gaussian noise term with zero mean and variance σ^2 with $\epsilon_i \sim N(0, \sigma^2)$. This corresponds to the following likelihood function:

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2\right), \tag{2.17}$$

which captures the probability of observing $\mathbf{y}$ given the data matrix $\mathbf{X}$, regression coefficient $\boldsymbol{\beta}$ and the noise variance σ^2. The linear regression model can then be concisely represented within the Bayesian framework as follows:

$$\mathbf{y}|\boldsymbol{X}, \boldsymbol{\beta}, \sigma^2 \ \sim \ N_n \left(\boldsymbol{X\beta}, \ \sigma^2 \boldsymbol{I}_n \right) \tag{2.18}$$

where $N_k(\cdot, \cdot)$ is the k-variate Gaussian distribution. By conditioning on the data $\mathbf{y}$ and $\mathbf{X}$ and applying Bayes's rule, the resulting posterior density takes the form of:

$$p(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) \times p(\boldsymbol{\beta}|\sigma^2) p(\sigma^2)}{p(\mathbf{y})}, \tag{2.19}$$

where $p(\boldsymbol{\beta}|\sigma^2)$ represents the chosen prior distribution, which hinges on the specific type of penalty inferred for the regression coefficient β and $p(\sigma^2)$ denotes the prior associated the noise variance. A widely adopted choice for $p(\sigma^2)$ is the standard scale-invariant prior $\sigma^2 \sim \sigma^{-2}\ d\sigma^2$ that is uninformative and expresses a priori ignorance regarding the scale of the data. However, for parameter estimation purposes, the normalizing constant $p(\mathbf{y})$ is frequently omitted since it does not rely on the parameters of interest $\boldsymbol{\beta}, \sigma^2$. Instead, a common practice involves working with the following proportional relationship:

$$p(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) \times p(\boldsymbol{\beta}|\sigma^2) p(\sigma^2). \tag{2.20}$$

In this section, we explore some well-established prior distributions that have been applied to the linear regression problem. Subsequently, we will explore widely used and effective techniques for computing the posterior distribution in Section 2.3.

As previously discussed, adopting a Bayesian perspective for penalized linear regression involves introducing priors on the parameters of interest, $\boldsymbol{\beta}$. If sparsity is the objective, opting for a sparsity-inducing shrinkage prior can be advantageous. The research on shrinkage priors is continuously expanding, with the two prevalent types being: (a) two-component discrete mixture priors and (b) continuous shrinkage priors. It is worth noting that many solutions to the penalized regression problem, as described in Equation (2.4), can be elegantly addressed in the Bayesian framework through the use of shrinkage priors combined with a posterior point estimate [173]. The choice of the posterior point estimate depends on the specific objective function of the penalized regression, resulting in nearly all regression penalties mentioned in the previous subsection having their corresponding Bayesian counterparts, as summarized in Table 2.2. This adaptability is achieved by varying the priors employed. In comparison to the classic penalization techniques mentioned in Section 2.1, Bayesian penalization techniques often demonstrate similar or superior performance, offering additional advantages such as readily available uncertainty estimates, automatic estimation of the penalty parameter, and greater flexibility in considering various penalty types [173].

Two-component discrete mixture priors are well-known for their effective performance in sparse learning within linear models. However, the discrete nature of this prior poses computational challenges in obtaining posterior inference, particularly with a large number of predictive variables. To address this, analytical approximations like variational inference (VI) or expectation propagation (EP) can be used. But this comes with the cost of increased analytical work required to derive separate equations for each model, making the implementation more complex [138]. For a comprehensive overview of inference approximation schemes and the differences between VI and EP, refer to [34]. On the other hand, continuous shrinkage priors offer comparable or even superior results and are more straightforward to implement. They can be conveniently computed using generic sampling tools that implement various Markov chain Monte Carlo (MCMC) algorithms. For instance, Stan [67] utilizes the Hamiltonian Monte Carlo algorithm and Just Another Gibbs Sampler (JAGS) [139], as the name suggests, implements Gibbs sampling.

2.2.1 Two-Component Discrete Mixture Prior

A widely used Bayesian approach to obtain sparse estimates of the regression coefficient $\boldsymbol{\beta}$ is through the Two-Component Discrete Mixture Prior. The prior, also known as spike-and-slab prior [57], consists of a mixture of two components: a "spike" component that concentrates its mass around zero, enabling shrinkage of small effects towards zero, and a "slab" component that spreads its mass over a wide range of plausible values for the regression coefficients. The spike and slab prior was initially introduced by Mitchell and Beauchamp [126] and is commonly represented as follows:

$$\begin{aligned} p(\boldsymbol{\beta}|\boldsymbol{\gamma},\sigma^2) &= \prod_{j=1}^{p}\left[(1-\gamma_j)\delta_0(\beta_j)+\gamma_j p(\beta_j|\sigma^2)\right], \\ p(\boldsymbol{\gamma}|\theta) &= \prod_{j=1}^{p}\theta^{\gamma_j}(1-\theta)^{1-\gamma_j}, \\ \theta &\sim \pi(\theta^2)d\theta, \\ \sigma^2 &\sim \sigma^{-2}d\sigma^2 \end{aligned} \tag{2.21}$$

where δ_0 is a point mass at zero (the "spike"), $p(\beta_j|\sigma)$ represents a uniform diffused density rescaled by the variance σ^2 (the "slab"), $\theta \in (0,1)$ is the mixing proportion and $\boldsymbol{\gamma}$ is a binary vector that indexes the 2^p possible models. There exist two different spikes that have been proposed in the literature: (a) Dirac delta function centered at zero and (b) an absolutely continuous distribution. The absolutely continuous spike can be specified using any unimodal continuous distribution with its mode at zero [117]. On the other hand, the Dirac delta function [20] is a specific mathematical function that takes the value of zero everywhere except at one point, where it can be interpreted as undefined or having an "infinite" value. Importantly, the integral of the Dirac delta function over the entire real number line is equal to one.

The spike-and-slab prior has long been regarded as the gold standard for Bayesian sparse inference, owing to its desirable properties that facilitate natural variable selection. However, it comes with a notable drawback – it is computational inefficiency, especially as the number of variables increases due to its combinatorial complexity. In the following subsection, we present an overview of existing continuous shrinkage priors. Among them, the horseshoe prior stands out as it has been shown to achieve comparable performance to the spike-and-slab prior while also enjoying the inherent computational advantage that comes with continuous shrinkage priors.

2.2.2 Global-Local Shrinkage Prior

Many continuous shrinkage priors can be represented in the class of global-local scale mixtures of normals [140]. In the class of global-local shrinkage priors, each β_j is assigned a continuous shrinkage prior centered at $\beta_j = 0$. The prior distribution for $\boldsymbol{\beta}$ can be represented in the class of global-local scale mixtures of normals [140]:

$$\begin{aligned} \beta_j|\lambda_j^2, \tau^2, \sigma^2 &\sim N(0,\ \lambda_j^2\tau^2\sigma^2) \\ \lambda_j^2 &\sim \pi(\lambda_j^2)d\lambda^2 \\ \tau^2 &\sim \pi(\tau^2)d\tau^2 \end{aligned} \tag{2.22}$$

where τ is the global shrinkage parameter that controls the overall degree of shrinkage, λ_j is the local shrinkage parameter associated with the jth predictor that loosens the amount of shrinkage on significantly large coefficients and $\pi(\cdot)$ are generic probability density functions that do not necessarily refer to the same function in different problems. To reduce noise and shrink all coefficients towards zero, τ^2 should be small as suggested in Polson and Scott [140]. And for large signals to override this effect, λ_j^2 has to be allowed to be large.

There exist many different global-local shrinkage priors in the literature, this includes many variants of the horseshoe prior[42] (eg horseshoe+ [26] and horseshoe-like [28]), the Dirichlet-Laplace[31], the generalized double Pareto[15], etc. Bhadra et al. [27] provides a comprehensive table of existing global-local shrinkage priors.

2.2.2.1 Ridge

Bayesian ridge regression [89] employs only the global shrinkage hyperparameter τ and does not incorporate any local shrinkage hyperparameters. Consequently, the ridge prior corresponds to a normal prior centered around 0 for the regression coefficients:

$$\beta_j|\tau, \sigma^2 \ \sim\ N(0, \tau^2\sigma^2). \tag{2.23}$$

This is equivalent to assuming a Dirichlet point-mass prior for $p(\lambda_j)$ or setting all λ_j in (2.22) to 1.

When $\boldsymbol{\beta}$ follows a multivariate normal prior distribution, the posterior distribution of $\boldsymbol{\beta}$ is [109]

$$\begin{aligned} \boldsymbol{\beta} \,|\, \tau, \sigma, \mathbf{y} &\sim N_p(\mathbf{A}^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \sigma^2\mathbf{A}^{-1}) \\ \mathbf{A} &= (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \tau^{-2}\mathbf{I}_p) \end{aligned} \tag{2.24}$$

From the above expression, it becomes evident that for any fixed value $\tau > 0$, the posterior mean (or mode) of $\boldsymbol{\beta}$ is equivalent to the ridge estimate (2.10) with penalty $\gamma = 1/\tau^2$.

2.2.2.2 Student-t

The Student-t distribution, often considered the second most essential continuous probability distribution after the normal distribution, holds significant importance in statistics and various scientific disciplines. It can be effectively employed as a prior distribution by expressing it using an inverse-gamma mixing density, leading to the following hierarchical specification:

$$\begin{aligned} \beta_j|\lambda_j, \tau, \sigma^2 &\sim N(0, \lambda_j^2\tau^2\sigma^2), \\ \lambda_j^2|\nu, \tau &\sim \mathrm{IG}\left(\frac{\nu}{2}, \frac{\nu}{2\tau}\right), \end{aligned} \tag{2.25}$$

for which by integrating out λ_j^2, the resulting conditional prior density for β follows a Student-t distribution:

$$\beta_j|\nu, \tau, \sigma^2 \sim t\left(\nu, 0, \frac{\sigma^2}{\tau}\right). \tag{2.26}$$

Here, $t(a, b, c)$ denotes a Student-t distribution with a degrees of freedom, location parameter b, and scale parameter c. The Student-t prior is characterized by heavier tails compared to the ridge prior in (2.23), and a smaller value for ν results in an even heavier-tailed distribution, with $\nu = 1$ corresponding to a Cauchy prior for β_j. For more comprehensive details on the Student-t distribution and its application as a shrinkage prior, readers can refer to [2, 75].

2.2.2.3 Lasso

Tibshirani [165] were among the first to highlight that the lasso estimate can be interpreted as a Bayes posterior mode estimate by placing a double-exponential (Laplace) prior on the regression coefficient $\boldsymbol{\beta}$:

$$\beta_j|\tau \sim \mathrm{La}\left(0,\ \tau\right). \tag{2.27}$$

Here, $\mathrm{La}(a, b)$ represents a Laplace distribution with a denoting the location parameter, b representing the rate parameter, and the probability density function is defined as:

$$p(\beta_j|\tau) \;=\; \frac{\tau}{2}\exp(-\tau|\beta_j|) \tag{2.28}$$

While this specific formulation of the Laplace prior does not lend itself to a tractable Bayesian analysis due to the lack of conjugacy with the conditional distribution in (2.18), Girolami [71] derived a variational EM algorithm using a convex variational representation of the Laplace prior to approximate the posterior mean. Subsequently, Figueiredo [62] leveraged the hierarchical nature of the Laplace prior and developed an EM algorithm to obtain the posterior mode estimate (MAP estimator). The Laplace prior distribution can be represented as a scale mixture of normal distributions with an exponential mixing density [10] for which the hierarchical model [62] employed was:

$$\begin{aligned} \beta_j|\lambda_j \;&\sim\; N(0, \lambda_j^2), \\ \lambda_j^2|\tau^2 \;&\sim\; \mathrm{Exp}\left(\frac{\tau^2}{2}\right), \end{aligned} \tag{2.29}$$

where $\mathrm{Exp}(r)$ denotes an exponential distribution with rate parameter r. In this two-level hierarchical Bayes model, each β_j is assumed to have a zero-mean Gaussian prior with its own variance λ_j, and each of the λ_j is given an exponential (hyper)prior with a rate parameter of $\frac{\tau^2}{2}$. Integrating out the latent variable $\boldsymbol{\lambda}$, we arrive at:

$$p(\beta_j|\tau) = \int_0^\infty p(\beta_j|\lambda_j)p(\lambda_j|\tau)d\lambda_j = \frac{\tau}{2}\mathrm{exp}\Big\{-\tau|\mathrm{\beta_j}|\Big\} \tag{2.30}$$

which recovers the Laplace density as in equation (2.27). This hierarchical representation enables a more tractable Bayesian analysis, facilitating the use of the EM algorithm to estimate the posterior mode.

Park and Casella [134], later made significant contributions to Bayesian lasso regression by proposing a comprehensive Bayesian approach for estimating posterior medians through Markov Chain Monte Carlo (MCMC) methods. They emphasized the importance of conditioning on σ^2 to achieve unimodality in the posterior distribution, as the lack of unimodality could slow down the convergence of the Gibbs sampler, which is an integral part of the MCMC methodology they employed. Furthermore, Park and Casella [134] extended their model to accommodate the uncertainty of noise variance and τ by assigning Gamma priors to the parameters. The hierarchical representation of the Bayesian lasso model [134] can be then formulated as:

$$\begin{aligned}
\beta_j|\lambda_j &\sim N(0, \lambda_j^2\sigma^2), \\
\lambda_j^2|\tau^2 &\sim \text{Exp}\left(\frac{\tau^2}{2}\right), \\
\sigma^2 &\sim \sigma^{-2}d\sigma^2, \\
\tau^2 &\sim \text{Ga}(r, \delta),
\end{aligned} \tag{2.31}$$

where $\text{Ga}(\text{a}, \text{b})$ represents the Gamma distribution with shape a and rate b. The conditional posteriors under this hierarchical representation are straightforward to derive, and readers can refer to [134] for more details.

We can strategically reposition the scale (inverse rate) parameter τ^2 associated with the exponential distribution on λ_j^2 within the hierarchy (2.31) to align it with the global-local SMN hierarchy defined in (2.22):

$$\begin{aligned}
\beta_j|\lambda_j, \tau^2, \sigma^2 &\sim N(0, \lambda_j^2\tau^2\sigma^2), \\
\lambda_j^2 &\sim \text{Exp}(2), \\
\sigma^2 &\sim \sigma^{-2}d\sigma^2, \\
\tau^2 &\sim \text{Ga}(r, \delta),
\end{aligned} \tag{2.32}$$

where τ represents the global shrinkage parameter and λ_j is the local shrinkage parameter. This hierarchical representation offers a more flexible and interpretable formulation for Bayesian analysis of the lasso regression model, making it easier to compare and contrast with other global-local shrinkage priors.

As previously mentioned in Section 2.1.1.1, there exists a diverse range of Lasso variants, each possessing its own distinctive characteristics. These variants include the elastic net, which combines the strengths of both the lasso (ℓ_1 penalty) and ridge (ℓ_2 penalty) regularization; the group lasso, which is just lasso applied to group regression; and the adaptive Lasso, a weighted lasso approach achieving oracle properties. Each of these variants has its own Bayesian counterpart that can be expressed in straightforward hierarchical forms. For a more comprehensive exploration and summary of posterior inferences involving these Lasso variants, interested readers can explore the discussions in [43, 101, 173].

2.2.2.4 Normal-Exponential-Gamma (NEG)

The NEG prior is a generalized version of the LASSO also known as Hyperlasso[76, 88], serving as a Bayesian alternative to the adaptive lasso. The motivation behind introducing the hyperlasso arises from the demanding nature of the weights included in the adaptive lasso. In particular, the weights used in the adaptive lasso can place significant demands on the data, resulting in diminished performance in prediction and variable selection, especially when dealing with small sample sizes [76].

The NEG prior can be viewed as a generalization of the laplace prior (2.27). The concept emerged from recognizing the exponential distribution in (2.29) as a special case of the gamma distribution, wherein $\lambda_j \sim \text{Ga}(\nu, \tau^2/2)$, with the density:

$$\frac{\tau^{2\nu}}{2^{\nu}\Gamma(\nu)}\lambda_j^{\nu-1}\exp\left(-\frac{\tau^2\lambda_j}{2}\right) \tag{2.33}$$

This led to the development of the normal-gamma (NG) prior, which has been utilized in fully Bayesian analyses for regression problems [39]. The NEG distribution was then formulated by introducing an additional level of hyperparameters ϕ_j, making each λ_j adaptive. This adaptivity allows the parameter of the exponential mixing distribution to vary according to a gamma distribution, given as:

$$\begin{aligned} \beta_j|\lambda_j &\sim N(0, \lambda_j^2), \\ \lambda_j^2|\phi_j &\sim \text{Exp}(\phi_\text{j}), \\ \phi_j|\nu, \tau^2 &\sim \text{Ga}\left(\nu, \frac{1}{\tau^2}\right). \end{aligned} \tag{2.34}$$

An alternative to the NEG prior is the Generalized Double Pareto (GDP) shrinkage prior, proposed by [15]. Although similar to the NEG prior, the GDP prior possesses a marginal density with a simple analytic form, making it advantageous for the study of specific properties.

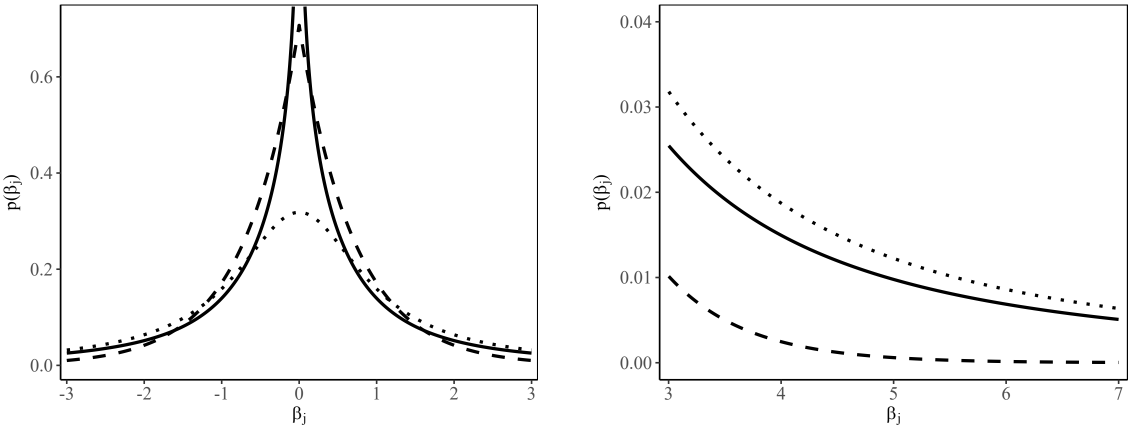
2.2.2.5 Horseshoe

The horseshoe prior [42] is widely regarded as one of the most popular global-local shrinkage priors in Bayesian inference. Its popularity can be attributed to several appealing properties it possesses: 1) an asymptote at zero, effectively heavily shrinking small coefficients towards zero; 2) heavy tails to prevent over-shrinking of large coefficients. This prior is represented as a scale mixture of normals with a half-Cauchy mixing density:

$$\begin{aligned} \beta_j|\lambda_j, \tau^2, \sigma^2 &\sim N(0, \lambda_j^2\tau^2\sigma^2), \\ \lambda_j &\sim \text{C}^+(0, 1), \end{aligned} \tag{2.35}$$

where $C^+(0,1)$ denotes a standard half-Cauchy distribution with a probability density function of

$$p(z) = \frac{2}{\pi(1+z^2)}, \; z > 0.$$



(A)

(B)

FIGURE 2.1: Comparison of the Lasso (dashed), Student-t$_{\gamma=1}$ (dotted), and horseshoe (solid) densities on the marginal prior over $\boldsymbol{\beta}$ around the origin (left) and the tail behavior (right). Figure adapted from [42].

Under this hierarchical specification, the marginal prior for each β_j is unbounded at the origin and the tails decay at a polynomial rate (Figure 2.1).

Regarding the choice of prior for the global shrinkage hyperparameter, in the absence of prior knowledge about the degree of shrinkage of the regression coefficients, [42, 141] recommend the half-Cauchy prior:

$$\tau \sim \mathrm{C}^{+}(0,1). \tag{2.36}$$

In cases where there is prior knowledge about the number of nonzero coefficients in the model, Piironen and Vehtari [137] suggest using a half-Cauchy prior as well, but with a scale parameter that depends on the prior guess for the number of relevant variables.

Implementing a standard Gibbs sampler directly under the representation (2.35-2.36) is challenging due to the non-standard form of the conditional posterior distributions for the hyperparameters $(\lambda_1, \cdots, \lambda_j)$ and τ. To address this issue, there exist a number of approaches to sampling the conditional posterior of the hyperparameters. These include slice sampling [143], blocked Metropolis-within-Gibbs sampler [94], an inverse-gamma inverse-gamma scale mixture representation [112], and a gamma-gamma scale mixture representation [14]. Among these methods, the approach presented by Makalic and Schmidt [112] stands out as the most straightforward, representing the half-Cauchy prior as a mixture of inverse-gamma distributions

$$\lambda_j^2|\nu_j \sim \mathrm{IG}(1/2, 1/\nu_j) \quad , \quad \nu_1, \cdots \nu_p \sim \mathrm{IG}(1/2, 1) \tag{2.37}$$

simplifying the computation of the conditional posterior distributions for the hyperparameters. The prior on τ can also be represented in this decomposition of the half-Cauchy prior such that:

$$\tau^2|\epsilon \sim \mathrm{IG}(1/2, 1/\epsilon) \quad , \quad \epsilon \sim \mathrm{IG}(1/2, 1). \tag{2.38}$$

Using this inverse-gamma representation of the half-Cauchy prior, the conditional posterior distribution of all parameters, except the regression coefficients $\boldsymbol{\beta}$, are also inverse-gamma, for which efficient sampling methods already exist [112, 146]. This makes the application of Gibbs sampling using this sampling scheme relatively straightforward. Further details on the full conditional posterior distribution can be found in Makalic and Schmidt [112].

There exist substantial theoretical findings that support the superiority of the horseshoe prior for sparse estimation. Most notably the horseshoe prior exhibits performance comparable to the spike and slab prior, and very often, it surpasses the Laplace prior in sparse estimation [41, 42, 140]. When the true parameter mean is zero, the horseshoe prior guarantees that the Bayes estimate for the sampling density converges to the true sampling density at a super-efficient rate in terms of the Kullback–Leibler (KL) divergence metric [42]. For a more comprehensive discussion on the horseshoe prior, readers can refer to [29, 50, 171].

2.2.2.6 Horseshoe +

The horseshoe+ estimator [26] is an extension of the horseshoe estimator tailored for ultra-sparse problems. It offers several advantages over the standard horseshoe estimator, including lower posterior mean squared error and faster posterior concentration rates, measured using the Kullback–Leibler distance between the true model and the estimator of the density function. The horseshoe+ prior extends the horseshoe such that an addition level of hyperparameter ϕ_j is introduced, where each ϕ_j corresponds to the prior variance associated with λ_j:

$$\begin{aligned} \lambda_j &\sim \mathrm{C}^+(0, \mathrm{\phi_j}), \\ \phi_j &\sim \mathrm{C}^+(0, 1). \end{aligned} \tag{2.39}$$

This new hierarchical structure results in a heavier tail as compared to the regular horseshoe prior, allowing for better separation of signals and improved handling of sparsity due to a larger mass near the origin [26]. The hierarchy (2.39) can also be expressed using the previously mentioned inverse-gamma inverse-gamma decomposition [112] as follows:

$$\begin{aligned} \lambda_j^2|\nu_j &\sim \mathrm{IG}(1/2, 1/\nu_j) \quad , \quad \nu_j|\phi_j^2 \sim \mathrm{IG}(1/2, 1/\phi_j^2), \\ \phi_j^2|\epsilon_j &\sim \mathrm{IG}(1/2, 1/\epsilon_j) \quad , \quad \epsilon_j \sim \mathrm{IG}(1/2, 1) \end{aligned} \tag{2.40}$$

which allows for an efficient Gibbs sampler for the horseshoe+ estimator, as the computation of the posterior conditional distributions for all hyperparameters is straightforward [111].

2.2.2.7 Normal Beta Prime

The normal beta prime (NBP) prior [18, 19] has been referred to by different names, such as the three-parameter beta (TPB) prior [14], the adaptive normal hypergeometric inverted beta prior [188], the inverse-gamma gamma prior [17], and the generalized horseshoe prior [155]. Despite the different names used, all of them describe the same underlying prior model, which involves placing a beta prime distribution over the local shrinkage parameter in the hierarchy (2.22). We summarize the research gaps addressed by the mentioned references in Table 2.3, along with their respective limitations.

The NBP prior places a beta prime (inverted beta) prior distribution over the local shrinkage parameters, i.e., $\boldsymbol{\lambda}^2 \sim \mathrm{B}'(a,b)$. This is equivalent to placing a beta distribution over $\kappa_j = \lambda_j^2(1+\lambda_j^2)^{-1}$ given by the density function

$$\pi(\kappa_j|a,b) = \frac{(\kappa_j)^{a-1}(1-\kappa_j)^{b-1}}{\mathrm{B}(a,b)}, \; a>0, b>0$$

where $\mathrm{B}(a,b)$ denotes the beta function. The hyperparameter a controls the sparsity level of $\boldsymbol{\beta}$ (i.e. smaller a values yield a marginal prior for $\boldsymbol{\beta}$ that concentrates more around $\boldsymbol{\beta}=0$); while the hyperparameter b can be seen as the tail-decay parameter (i.e. smaller values indicate a slower rate of decay at the tails of the marginal distribution) [155]. We can achieve many different prior distributions by varying the values for a and b. Notable examples include the Strawderman–Berger prior ($a=0.5, b=1$), the horseshoe prior ($a=0.5, b=0.5$), and negative exponential gamma (NEG) prior ($a=1, b>0$) as special cases.

2.3 Posterior approximation

Bayesian computation is built upon a twofold process: firstly, defining a prior distribution for all parameters, including both observed and latent variables; and secondly, computing the posterior distribution. Up to this point, we explored numerous prevalent choices for priors. Among these, some lead to posterior distributions that can be analytically computed in closed form (e.g., Equation 2.24), while many others result in complex posterior distributions that lack straightforward computation methods. The latter cases often involve non-standard distributions, posing challenges or rendering analytical solutions infeasible. To address this issue, Monte Carlo methods have emerged as powerful tools for approximating posterior distributions through the process of random sampling. The simulation samples can be used to approximate almost any

Table 2.3: Summary of existing literature on models with the beta prime prior, including research gaps addressed and limitations.

Prior	Notes
Three-parameter beta (TPB) [14]	• Introduced the TPB prior, a flexible generalized Beta distribution, which encompasses many popular priors as special cases. • Recommend fixing a & b, learn model sparsity through tuning τ.
Adaptive normal hypergeometric inverted beta (ANHIB) [188]	• Proved that the Bayes estimator under the ANHIB prior attains the Kullback-Leibler super-efficiency under sparsity and robust shrinkage rules for large observations. • Missing information on the implementation details of the sampler.
Normal-beta prime (NBP) [19]	• Proposed the NBP model as a self-adaptive method that learns the true sparsity level from the data. Utilized the EM algorithms for hyperparameter tuning. • Proposed model does not include the global shrinkage parameter, thereby recommending it to be fixed to 1.
Group inverse-gamma gamma (GIGG) [37]	• Extended the NBP prior to address regression problems with grouping structures. The priors on the group shrinkage parameter $\boldsymbol{\delta}^2$ and $\boldsymbol{\lambda}^2$ are selected such that $\delta_g^2 \lambda_{jg}^2 \sim \mathrm{B}'(a_g, b_g)$. • Proposed model is not equipped to handle overlapping group structures.
Generalized Horseshoe [155]	• Presented efficient samplers for λ_j in the case of the GHS prior. • Application of the method to estimate the hyperparameters were not presented, making the proposed model non-adaptive.

quantity relevant to Bayesian inference, including posterior expectations, variances, quantiles, and marginal densities.

In this section, we discuss two classes of methods for approximating the posterior distribution: sampling approximation and distributional approximation. The former involves Monte Carlo methods (Section 2.3.1) and Markov Chain Monte Carlo (MCMC) techniques (Section 2.3.2). The accuracy of these approximations tends to improve as the number of samples increases, often bounded primarily by computational resources. However, situations might arise where good sampling approximation is not feasible, or iterative simulation approaches prove excessively slow. In such scenarios, distributional approaches (Section 2.3.3) offer quick inference of the posterior.

2.3.1 Monte Carlo sampling methods

Monte Carlo methods constitute an extensive category of computational algorithms that rely on iterative random sampling for numerical results. In the context of Bayesian inference, the key idea is to generate samples from the target posterior distributions using a proposal distribution that is comparatively simpler and one for which the integrals are either known or can be computed numerically. Subsequently, discrete formulas are applied to compute the necessary statistics (e.g.

posterior mean, mode, and median). This subsection provides a concise introduction to the two commonly used Monte Carlo sampling methods: Rejection sampling and importance sampling.

2.3.1.1 Rejection Sampling

Rejection sampling, also known as accept-reject method, is a simple and intuitive technique for generating samples from a target probability distribution $p(x)$ that is known up to a proportionality constant, however, is too complex to sample from directly. The method involves sampling from a simpler proposal distribution $q(x)$ that is easy to sample from. These samples are then subjected to acceptance or rejection based on a criterion defined by the ratio $p(x)/Mq(x)$. Here, M is a constant chosen such that the distribution $Mq(x)$ consistently encloses or serves as an upper bound of $p(x)$ across all possible values of x. In other words, for the rejection sampling approach to be effective, it is necessary that M and $q(x)$ satisfy $p(x) \leq Mq(x)$ for all x.

Suppose we want to obtain a single random draw from the posterior density $p(\theta|y)$, or more commonly, from the unnormalized posterior density $\tilde{p}(\theta|y)$. Given a proposal distribution $q(\theta)$ that allows for easy sampling, along with an upper bound M on $\tilde{p}(\theta|y)/q(\theta)$, the rejection sampling algorithm unfolds as follows:

1. Sample θ from $q(\theta)$.
2. Sample u from a uniform distribution $\mathcal{U}(0, 1)$
3. If $u \leq \frac{\tilde{p}(\theta|y)}{Mq(\theta)}$, accept θ as a draw from p; Otherwise, reject θ and return to step 1.

Repeat the above procedure until the desired number of (accepted) samples N is obtained. With sufficient samples N, the exact target distribution $p(\theta|y)$ can be accurately reconstructed.

A good proposal density $q(\theta)$ should ideally exhibit rough proportionality to $p(\theta|y)$. The best case scenario is when $q(\theta) \propto p(\theta|y)$, for which this results in each sample drawn being accepted with a probability of 1 given a suitable value of M. When q is not closely proportional to p, selecting an upper bound M becomes a strategic task to ensure adequate enclosure of the proposal distribution to the target distribution without excessive bounds that would lead to the rejection of most samples drawn in step 1. A rejection sampler can be highly inefficient when the target distribution differs significantly from the proposal distribution, leading to a high rejection rate. Consequently, rejection sampling proves efficient only when the posterior is sufficiently bounded by a known function proportionate to an easily samplable density. Finding such a function remains a challenge, especially in scenarios with high-dimensional complexity and when dealing with target distributions with heavy tails.

2.3.1.2 Importance Sampling

Importance sampling can be conceptualized as a variant of rejection sampling. While both techniques utilize a proposal distribution, in the case of importance sampling, all samples from the proposal distribution are accepted. However, unlike rejection sampling, importance sampling is not employed to generate samples from the target distribution. Instead, it serves the purpose of approximating expectations.

To elaborate, importance sampling is a methodology used to compute expectations through a weighted average of random samples derived from an approximation of the target distribution, which is represented by what we call the proposal distribution. Consider a scenario where our objective is to determine the posterior expectation of a function $h(\theta)$. This expectation, denoted as $E[h(\theta)|y] = \int h(\theta)p(\theta|y)d\theta$, becomes challenging when direct sampling of θ from the posterior distribution $p(\theta|y)$ is infeasible. This complication inhibits the straightforward use of a simple average of simulated values to evaluate the integral.

However, by introducing a proposal distribution denoted as $q(\theta)$, it becomes possible to rearrange the terms within the target integral such that:

$$E[h(\theta)|y] = \int h(\theta)p(\theta|y)d\theta = \int h(\theta)\frac{p(\theta|y)}{q(\theta)}q(\theta)d\theta = \int h(\theta)w(\theta)q(\theta)d\theta \tag{2.41}$$

where $w(\theta) = \frac{p(\theta|y)}{q(\theta)}$ is the importance/sample weights and the expectation is now with respect to $q(\theta)$ instead of $p(\theta|y)$. The importance weights adjust the contribution of each sampled point to the final estimate. Therefore, we can now estimate the expectation using N draws $\theta^{(1)}, \cdots, \theta^{(N)}$ from $q(\theta)$ by the expression:

$$E[h(\theta)|y] = E_q[w(\theta)h(\theta)|y] = \frac{1}{N}\sum_{j=1}^{N} w(\theta^{(j)})h(\theta^{(j)}) \tag{2.42}$$

where $E_q[\cdot]$ denotes the expectation with respect to $q(\theta)$. In the event whereby the posterior density is only known up to a normalizing constant for which we are working with the unnormalized posterior density $\tilde{p}(\theta|y)$ whereby $p(\theta|y) = \frac{\tilde{p}(\theta|y)}{Z}$, we get

$$E[h(\theta)|y] = \int h(\theta)\frac{\tilde{p}(\theta|y)}{Z}d\theta = \int h(\theta)\frac{\tilde{p}(\theta|y)}{Zq(\theta)}q(\theta)d\theta = \frac{1}{Z}\int h(\theta)\tilde{w}(\theta)q(\theta)d\theta \tag{2.43}$$

where $\tilde{w}(\theta) = \frac{\tilde{p}(\theta|y)}{q(\theta)}$ and the normalizing constant Z can also be approximated using the same technique as in (2.41) such that

$$Z = \frac{1}{N}\sum_{j=1}^{N}\frac{\tilde{p}(\theta|y)}{q(\theta)} = \frac{1}{N}\sum_{j=1}^{N}\tilde{w}(\theta^{(j)})$$

leading to the realization that:

$$E_q[\tilde{w}(\theta)h(\theta)|y] = \frac{\frac{1}{N}\sum_{j=1}^{N}\tilde{w}(\theta^{(j)})h(\theta^{(j)})}{\frac{1}{N}\sum_{j=1}^{N}\tilde{w}(\theta^{(j)})} \tag{2.44}$$

However, similar to rejection sampling, the accuracy of approximations obtained through importance sampling is heavily dependent on the choice of the proposal distribution $q(\theta)$. Often, it is challenging to find a suitable $q(\theta)$ that is both easy to sample from and offers a good approximation to $p(\theta|y)$. This challenge becomes more pronounced as the dimension of θ increases because the relative volume of θ where $p(\theta|y)$ is high becomes extremely small. To address this issue, alternative variants of the importance sampling method have been introduced. Some examples are Sequential Importance Sampling and Annealed Importance Sampling.

2.3.2 Markov chain Monte Carlo (MCMC) sampling methods

Markov chain Monte Carlo (MCMC) methods extend the Monte Carlo framework by introducing the concept of Markov chains[1]. Instead of drawing independent samples, MCMC generates samples from a Markov chain, where each sample depends on the previous one. The goal is to construct a Markov chain that gradually converges to the target distribution. In each simulation step, the generated sample is progressively refined or improved such that it gets closer to the target distribution. Consequently, MCMC sampling methods often incorporate a phase of "burn-in" or "warm-up", wherein an initial set of samples is discarded at the start of the simulation. This practice acknowledges that these initial samples may not yet accurately represent the target distribution, but as the iterations proceed, these samples progressively shift towards better resembling the target distribution.

MCMC methods find their primary utility in sampling from multidimensional distributions, particularly when dealing with high-dimensional scenarios. For single-dimensional distributions, the preference often tilts towards Monte Carlo methods (Section 2.3.1). These methods have the advantage of directly providing independent samples from the distribution, sidestepping the inherent issue of autocorrelated samples in MCMC methods. This section provides a brief discussion of two prominent MCMC sampling methods: the Metropolis-Hastings algorithm (Section 2.3.2.1) and the Gibbs sampler (Section 2.3.2.2).

[1] A Markov chain is a mathematical concept used to describe a sequence of states (or in our cases, random variables $\theta^{(1)}, \theta^{(2)}, \cdots,$), where the future state depends only on the current state and is unaffected by any earlier state. This property is known as the Markov property or the memorylessness property.

2.3.2.1 Metropolis-Hasting

The Metropolis-Hastings (MH) algorithm, introduced by [123] and later generalized by [83], involves navigating through a probability distribution using a random walk by proposing new samples based on a chosen proposal distribution. Each proposed sample is accepted or rejected using an acceptance criterion that considers the ratio of the probability of the proposed sample to the current sample. This criterion determines whether the new sample is more likely to belong to the target distribution than the previous sample, facilitating convergence to the target distribution. To illustrate its application in a Bayesian context, consider the posterior distribution $p(\theta|y)$ of interest, along with a proposal distribution $q(\theta)$. The MH algorithm operates as follows:

1. Pick an initial sample $\theta^{(0)} \sim q(\theta)$
2. For iteration $i = 1, 2, \cdots$
   - Propose: $\theta^* \sim q(\theta^*|\theta^{(i-1)})$
   - Sample u from $\mathcal{U}(0,1)$
   - Acceptance scheme:

$$\theta^{(i)} = \begin{cases} \theta^* & \text{if } u < \min\left(1, \frac{p(\theta^*|y)q(\theta^*|\theta^{(i-1)})}{p(\theta^{(i-1)}|y)q(\theta^{(i-1)}|\theta^*)}\right) \\ \theta^{(i-1)} & \text{otherwise} \end{cases}$$

In this context, the normalization term of $p(\theta|y)$ is not a requirement, as it cancels out in the acceptance ratio, thereby allowing the use of the unnormalized posterior $\tilde{p}(\theta|y)$. Much like other sampling methods we've explored, selecting an appropriate proposal distribution for the Metropolis-Hastings (MH) algorithm involves a delicate balance. The proposal distribution we opt for shapes the magnitude of transitions between proposed samples. A choice of a larger step size can lead to excessive rejections, while a smaller step size can hinder effective exploration (typically, an acceptance rate between 40% and 70% is indicative of a suitable proposal). This aspect represents a limitation of the MH algorithm. Furthermore, MH often has difficulties exploring multimodal distributions.

More recently, various variants of the MH algorithm have emerged, categorized by three different types of proposals as discussed in Titsias and Papaspiliopoulos [169]: (i) likelihood-informed proposals (e.g., preconditioned Metropolis-adjusted Langevin algorithm (pMALA) [147] and the manifold MALA [72]); (ii) prior-informed proposals (e.g., [24]); and (iii) proposals informed by both prior and likelihood (e.g., the preconditioned Crank-Nicolson Langevin samplers [49] and the auxiliary gradient-based sampling algorithms [169]). Schmidt and Makalic [155] subsequently adapted the auxiliary gradient based-sampling schemes [169] to the global-local shrinkage framework, a key component of the research presented in Chapter 6. This adaptation forms the basis of the sampler proposed in Chapter 6 to estimate the hyperparameters for the normal-beta prime prior.

Gradient-assisted MH for the global-local shrinkage framework The auxiliary gradient based-sampling algorithms proposed by Titsias and Papaspiliopoulos [169] are constructed using a combination of auxiliary variables and Taylor expansions. The core idea behind this approach is to treat the proposals used in standard MCMC sampler as auxiliary random variables and use this augmented target in conjunction with a first-order Taylor series expansion of the likelihood and a marginalization step to build a Metropolis-Hastings proposal density that is both likelihood and prior informed. Specifically, within the global-local shrinkage framework 2.22, the target density is of the form

$$p(\boldsymbol{\beta}) \propto \exp(f(\boldsymbol{\beta};\beta_0,\sigma^2))N(\boldsymbol{\beta}|0,\lambda_j^2\tau^2\sigma^2)$$

where $\exp f(\boldsymbol{\beta};\beta_0,\sigma^2)$ is the likelihood and $N(\boldsymbol{\beta}|0,\lambda_j^2\tau^2\sigma^2)$ is the Gaussian prior. Schmidt and Makalic [155] adapts this representation to provide straightforward sampling procedures suitable for a wide range of non-Gaussian data distributions. In Chapter 6, we have a different target distribution, for which instead of $p(\boldsymbol{\beta})$ being the subject of interest, our focus is on $p(\boldsymbol{\lambda})$.

2.3.2.2 Gibbs sampler

The Gibbs sampler, introduced by Geman and Geman [68] and later formalized by Gelfand and Smith [65], is a specialized MCMC algorithm that simplifies sampling from complex high-dimensional distributions by iteratively drawing samples from the conditional distributions of individual variables while holding the other variables fixed. This method, applicable to distributions with at least two dimensions, can be viewed as a special case of the MH algorithm with an acceptance probability fixed at one. When dealing with high-dimensional problems involving the estimation of multiple parameters $(\theta_1,\theta_2,\cdots)$, direct sampling from the joint posterior distribution $p(\theta_1,\theta_2,\cdots|y)$ is frequently a challenging endeavor. However, in many scenarios, it's possible to analytically derive the conditional posterior distributions of individual random variables θ_j, frequently resulting in well-known distribution forms. In such cases, directly sampling from the conditional posterior distributions of the parameters becomes a feasible approach. These conditional distributions often are conjugate distributions, making simulation even more straightforward. The algorithm can be outlined as follows:

1. Initialize $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)},\theta_2^{(0)},\cdots)$
2. For iteration $i=1,2,\cdots$ sample $\boldsymbol{\theta}^{(i)}$ such that:

$$\begin{aligned}
\theta_1^{(i)} &\sim \pi(\theta_1|\theta_2^{(i-1)},\theta_3^{(i-1)},\cdots)\\
\theta_2^{(i)} &\sim \pi(\theta_2|\theta_1^{(i)},\theta_3^{(i-1)},\cdots)
\end{aligned}$$

$$\begin{aligned}\theta_3^{(i)} &\sim \pi(\theta_3|\theta_1^{(i)}, \theta_2^{(i)}, \cdots) \\ &\vdots\end{aligned}$$

The core of the Gibbs sampling approach lies in decomposing the challenge of sampling from complex joint posterior distributions into a sequence of steps that involves sampling from lower-dimensional conditional posterior distributions. By sequentially traversing each variable (or a block of variables), we can sample from the conditional posterior distribution with the remaining variables set to their current values. While the feasibility of direct sampling depends on the availability of simple conditional posterior distributions, Gibbs sampler computations can often benefit from the introduction of auxiliary variables. An illustrative example of this strategy is the sampling scheme introduced by Makalic and Schmidt [112] for the horseshoe prior, wherein auxiliary variables are introduced to establish conjugate conditional posterior distributions for all parameters, making the application of Gibbs sampling relatively straightforward. In cases where a straightforward conditional distribution remains elusive even with auxiliary variables, turning to the MH algorithm could serve as a suitable alternative. This forms the basis of the "Metropolis-within-Gibbs" sampling method, which is essentially a hybrid technique that combines Gibbs sampling and the MH algorithm. Within the Metropolis-within-Gibbs approach, Gibbs steps are used to sample from the easily accessible conditional posterior distributions whenever possible. When dealing with variables for which direct Gibbs sampling is not straightforward due to complex or non-standard conditional distributions, the MH steps are employed to propose updates across multiple dimensions. This adaptability enables the exploration of challenging parameter spaces that may be difficult to address using either Gibbs sampling or MH alone.

When variables in the distribution are highly correlated, Gibbs samplers might require a large number of iterations to effectively explore the joint distribution. In such cases, more advanced MCMC sampling methods like Hamiltonian Monte Carlo (HMC) [56, 130] or No-U-Turn Sampler (NUTS) [87] could be more efficient.

2.3.3 Non-sampling approaches

Moving beyond sampling, there exists a diverse range of techniques to approximate posterior distributions. Instead of drawing samples directly from the posterior distribution, non-sampling methods typically revolve around approximating key posterior moments (e.g. mean, mode, and variance) through analytical or optimization methods. The strength of these methods lies in their computational efficiency and scalability, making them particularly well-suited for scenarios involving high-dimensional spaces and large datasets, whereby sampling methods like MCMC can be time-consuming. Non-sampling methods are valuable for quick inferences, often serve as good starting points for Markov chain simulation algorithms, and are particularly useful for large problems where iterative simulation approaches are too slow. However, a common critique of non-sampling methods is their handling of parameter uncertainty. These methods often don't provide a comprehensive representation of the full parameter posterior distribution. This raises questions about the trustworthiness of the approximate Bayesian inference that follows compared to the actual Bayesian inference. Unfortunately, the built-in approximations carry an inherent bias that is generally hard to correct, even with increased computing resources. This issue isn't exclusive to non-sampling techniques—it also applies to sampling methods. However, sampling methods, such as MCMC, tend to become more accurate with more drawn samples. Consequently, sampling-based methods (Section 2.3.1-2.3.2) present an appealing choice when accuracy is a top priority. Conversely, when efficiency takes precedence, non-sampling methods explored in this section offer practicality because in many situations, estimating posterior moments proves sufficient without the need to explore the full posterior distribution. At a conceptual level, the alignment of Bayesian posterior medians/modes with penalized likelihood estimators reinforces the intriguing interplay between Bayesian and frequentist frameworks.

This section discusses two well-known techniques grounded in distributional approximations: variational Bayes (VB) and the Expectation-Maximization (EM) algorithm. It's important to note that while not covered in this section, there exist other methods like Laplace approximation and Expectation Propagation (EP) that contribute to the broader landscape of non-sampling distribution approximation methods.

2.3.3.1 Variational Bayes

Variational Bayes (VB) [35, 95] methods approximate the intractable posterior distribution using proposal distributions $q(\boldsymbol{\theta})$ selected from some tractable family of distribution $\mathcal{Q}$. This predefined family is typically chosen based on mathematical convenience or computational feasibility with options like the exponential family often being favored. The core objective of VB is to find a specific $q^*(\boldsymbol{\theta}) \in \mathcal{Q}$ that is closest to the true posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$, based on the Kullback-Leibler (KL) divergence measure.

The optimal VB approximation $q^*(\boldsymbol{\theta})$ is obtained by minimizing the KL divergence from $q(\boldsymbol{\theta})$ to $p(\boldsymbol{\theta}|\mathbf{y})$:

$$q^*(\boldsymbol{\theta}) = \underset{q(\boldsymbol{\theta})\in\mathcal{Q}}{\operatorname{argmin}} \left\{ \mathrm{KL}\left(q(\boldsymbol{\theta}) \| p(\boldsymbol{\theta}|\mathbf{y}) \right) \right\}$$

Given that the posterior distribution can be expressed as $p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$, the minimization of the KL divergence can be reformulated as:

$$\begin{aligned}
\mathrm{KL}\left(q(\boldsymbol{\theta})\|p(\boldsymbol{\theta}|\mathbf{y})\right) &= \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\theta|y)} d\theta && (2.45)\\
&= \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})p(y)}{p(y|\theta)p(\theta)} d\theta \\
&= -\int q(\boldsymbol{\theta}) \log \frac{p(y|\theta)p(\theta)}{q(\boldsymbol{\theta})} d\theta + \log\ p(y) \\
&= -\mathcal{LB}(\boldsymbol{\theta}) + \log\ p(\mathbf{y}) && (2.46)
\end{aligned}$$

Here, $\mathcal{LB}(\boldsymbol{\theta})$ is the evidence lower bound (ELBO), named so because it serves as a lower bound for the log evidence (aka marginal data density $p(\mathbf{y})$). Since whereby $\mathcal{LB}(\boldsymbol{\theta})$ is the evidence lower bound (ELBO) for which the name comes about because it is a lower bound for the log evidence (marginal data density $p(\mathbf{y})$). This stems from the fact that the KL divergence is always non-negative, and since log $p(\mathbf{y})$ is a negative constant (the logarithm of a value between 0 and 1 is negative), it follows that $\mathcal{LB}(\boldsymbol{\theta}) \leq \log\ p(\mathbf{y})$. Therefore, minimizing the KL divergence in 2.45 is equivalent to maximizing the ELBO of the form:

$$\mathcal{LB}(\boldsymbol{\theta}) = \mathbb{E}_{q(\theta)}[\log p(y|\theta)] + \mathbb{E}_{q(\theta)}[\log p(\theta)] - \mathbb{E}_{q(\theta)}[\log q(\boldsymbol{\theta})] \tag{2.47}$$

where we denote $\mathbb{E}_{q(\theta)}[\cdot]$ as expectations taken wrt $q(\boldsymbol{\theta})$. In equation 2.46, we see that $\mathcal{LB}(\boldsymbol{\theta}) = \log\ p(\mathbf{y})$ if and only if $\mathrm{KL}\left(q(\boldsymbol{\theta})\|p(\boldsymbol{\theta}|\mathbf{y})\right)$ and in this case the $q(\boldsymbol{\theta}) == p(\boldsymbol{\theta}|\mathbf{y})$ therefore $\mathcal{LB}$ will always be a lower bound of log $p(\mathbf{y})$.

A variety of approaches exist for defining the class of distributions for $q(\boldsymbol{\theta})$. The complexity of the chosen variational family, $\mathcal{Q}$, significantly impacts the optimization process. Opting for a simpler family simplifies optimization, while a more complex family offers the potential to enhance the fidelity of the approximation at the cost of a more challenging variational optimization problem. The simplest and widely used variation is the Mean Field Variational Bayes (MFVB), wherein each variable θ_j is constrained to be independent and the variation distribution over the random variables can be factorized as:

$$q(\boldsymbol{\theta}) = \prod_{j=1}^{p} q_j(\theta_j) \tag{2.48}$$

This leads us to focus on optimizing the Evidence Lower Bound (ELBO) within this factorized distribution. Through a coordinate ascent optimization approach, it can then be demonstrated that the optimal densities $q(\theta_j)$ adhere to the equation:

$$\log(q_j(\theta_j)) \propto \mathbb{E}_{-j}[\log p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})] \tag{2.49}$$

where $\mathbb{E}_{-j}[\cdot]$ denotes expectations taken wrt all variables except θ_j. This iterative process is repeated for all p random variables until the algorithm converges - the ELBO hasn't changed significantly from one iteration to the next. In each iteration, the density of each variable is refined by considering the unnormalized joint posterior while keeping all other parameters fixed. The algorithm unfolds as follows:

1. for $j \in 1, \cdots, p$:
$$q_j(\theta_j) \quad \propto \quad \exp\{\mathbb{E}_{-j}[\log p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})]\}$$
2. $\text{ELBO}(q(\boldsymbol{\theta})) = \mathbb{E}_{q(\theta)}[\log p(y|\theta)] + \mathbb{E}_{q(\theta)}[\log p(\theta)] - \mathbb{E}_{q(\theta)}[\log q(\boldsymbol{\theta})]$
3. Repeat Step 1 and 2 until $\text{ELBO}_{(\text{t})} - \text{ELBO}_{(\text{t}-1)} \leq \epsilon$

However, the standard mean-field approximation suffers from a significant drawback attributed to its constrained distribution class, resulting in compromised accuracy. To address this shortcoming, extensions to the mean-field approximation have been introduced. These extensions relax the constraints on the approximating distributions $\mathcal{Q}$, allowing for variable dependencies and a more complex yet tractable structure. This approach is commonly referred to as structured or fixed-form Variational Bayes and has been explored in works such as [21, 151]. Another variation involves exploring mixtures of variational densities [33, 91]. While these alternatives give rise to more complex variational optimization problems, they are accompanied by improved approximations.

A method closely aligned with VB is Expectation Propagation (EP) [124]. In EP, the task of approximating a complex probability distribution is also reformulated as an optimization problem, utilizing an iterative approach that capitalizes on the factorized structure of the target distribution. A key distinction between VB and EP lies in their optimization objectives: while VB minimizes $\text{KL}\left(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}|\mathbf{y})\right)$, EP minimizes $\text{KL}\left(p(\boldsymbol{\theta}|\mathbf{y})|q(\boldsymbol{\theta})\right)$.

2.3.3.2 Expectation-Maximization (EM) algorithm

The Expectation-Maximization (EM) algorithm [54] is an iterative procedure for solving the maximum likelihood estimates (MLE) of parameters, in cases involving incomplete (missing) data. This algorithm exhibits broader applicability — it can also be used in models with latent variables[2]. In such cases, the latent variables are treated as missing and we proceed to apply the EM algorithm. In Bayesian statistical contexts, the EM algorithm is often used to find the mode of the posterior marginal distributions of parameters.

Suppose that the complete data set consists of $(\mathbf{y}, \mathbf{Z})$ for which $\mathbf{y}$ is the observed data and $\mathbf{Z}$ is the latent (missing) variable. The complete-data log-likelihood is then denoted by $\log p(\mathbf{y}, \boldsymbol{Z}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the unknown parameter vector for which its MLE is of interest. Specifically:

$$\begin{aligned}
\boldsymbol{\theta}_{\text{MLE}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \log p(\mathbf{y}|\boldsymbol{\theta}) \right\}, && (2.50)\\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \log \int p(\mathbf{y}, \boldsymbol{Z}|\boldsymbol{\theta}) d\mathbf{Z} \right\}. && (2.51)
\end{aligned}$$

Although straightforward in theory, maximizing such a complete data log-likelihood encounters challenges in practice. The actual complete dataset $\mathbf{y}, \boldsymbol{Z}$ remains elusive, with only incomplete data $\mathbf{y}$ available. The same principle applies when extended to estimate the Maximum A Posteriori (MAP) estimate of $\boldsymbol{\theta}$:

$$\begin{aligned}
\boldsymbol{\theta}_{\text{MAP}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\mathbf{y}|\boldsymbol{\theta}) p(\boldsymbol{\theta})\\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\boldsymbol{\theta}|\mathbf{y})\\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \log \int p(\mathbf{Z}, \boldsymbol{\theta}|\mathbf{y}) \mathbf{dZ} \right\}
\end{aligned} \tag{2.52}$$

where the only difference between MLE and MAP estimates is the consideration of a prior distribution $p(\boldsymbol{\theta})$ on the parameter. This inclusion can be viewed in non-Bayesian terms as introducing a regularization term through the log prior [128, Sec. 6.5]. Consequently, in the context of Bayesian analysis, the EM algorithm relies on the conditional posterior distribution $p(\boldsymbol{Z}|\mathbf{y}, \boldsymbol{\theta})$ to infer the values of $\boldsymbol{Z}$.

The EM algorithm iteratively alternates between two steps - the expectation step (E-step) and the maximization step (M-step); The former computes the expected log-likelihood for the complete data and the latter optimizes the parameters:

1. Randomly initialise the parameters, $\boldsymbol{\theta}_{\text{old}}$

2. Repeat until a convergence criterion is satisfied:

[2] At the highest level of generality within Bayesian statistics, there is no difference between a parameter and a latent variable. Both are regarded as unobserved random variables for which the objective is to compute a posterior distribution, given the observed random variables (i.e., data). In the context of the EM framework, latent variables are viewed as hidden variables that do not directly influence the likelihood. This notion is applicable to all Bayesian models, but it doesn't introduce any substantial conceptual distinction, particularly in terms of method implementation. Latent variables are integrated out during this process, and the act of maximizing the likelihood should only involve model parameters. This principle is also evident in the EM steps: the E-step involves marginalizing latent variables, while the M-step optimizes model parameters after having the latent variables imputed with their expectations.

- **E-step:** Compute the expected value of $\log p(\mathbf{y}, \mathbf{Z}|\boldsymbol{\theta})$ given the observed data $\mathbf{y}$, and the current parameter estimate, $\boldsymbol{\theta}_{\text{old}}$. This expectation is denoted as the Q-function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}})$:

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}}) &= \mathbb{E}_{\mathbf{Z}|\mathbf{y},\boldsymbol{\theta}_{\text{old}}}\left[\log p(\mathbf{y}, \mathbf{Z}|\boldsymbol{\theta})\right] && (2.53) \\ &= \int p(\mathbf{Z}|\mathbf{y}, \boldsymbol{\theta}_{\text{old}}) \log p(\mathbf{y}, \mathbf{Z}|\boldsymbol{\theta}) d\mathbf{Z} && (2.54) \end{aligned}$$

  where $p(\mathbf{Z}|\mathbf{y}, \boldsymbol{\theta}_{\text{old}})$ is the conditional density of the latent (missing) variable $\mathbf{Z}$ given the observed data $\mathbf{y}$, and assuming that $\boldsymbol{\theta} = \boldsymbol{\theta}_{\text{old}}$. For MAP estimation, the Q-function includes an additional term: $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}}) + \log p(\boldsymbol{\theta})$.

- **M-step:** Maximizing the Q-function wrt $\boldsymbol{\theta}$ to revise the parameter estimate $\boldsymbol{\theta}_{\text{new}}$:

$$\boldsymbol{\theta}_{\text{new}} \leftarrow \underset{\boldsymbol{\theta}}{\text{argmax}}\; Q(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}})$$

- Update the parameter $\boldsymbol{\theta}_{\text{old}} \leftarrow \boldsymbol{\theta}_{\text{new}}$

In cases where complex computations are involved in either the E-step or M-step of the EM algorithm, various adaptations have been developed to address this complexity. For instance, when the E-step presents challenges, such as lacking a closed-form solution for the Q-function, the Monte Carlo EM (MCEM) algorithm [179] offers a good alternative. In this approach, the E-step is replaced with a Monte Carlo (MC) process, involving the following steps:

1. Draw M independent missing/latent values $\mathbf{Z} = z_1, \cdots, z_M$, from the conditional distribution $p(\mathbf{Z}|\mathbf{y}; \boldsymbol{\theta}_{\text{old}})$

2. Approximate the Q-function as

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}}) \approx Q_M(\boldsymbol{\theta}, \boldsymbol{\theta}_{\text{old}}) = \frac{1}{M} \sum_{m=1}^{M} \log p(\mathbf{y}, z_{(m)}|\boldsymbol{\theta}).$$

It's crucial to cautiously select the value of M and monitor algorithm convergence [179]. The Stochastic EM arises as a special case when $M = 1$, where the E-step is substituted with a sampling step — each iteration draws one sample from the conditional posterior. Conversely, for challenging M-step scenarios, the Expectation-Conditional Maximization (ECM) algorithm [122] offers a solution. ECM replaces complex M-steps with computationally simpler conditional maximization (CM) steps, which can involve closed-form solutions or iterative approaches [118]. Despite ECM's potentially slower convergence rate compared to the standard EM algorithm in terms of iterations, it stands out for its computational efficiency, as CM maximizations often pertain to low-dimensional spaces, resulting in faster computational times.

The EM algorithm is guaranteed to converge to a local optimum, and its sensitivity to initialization becomes evident when dealing with a non-convex objective function (Q-function). When the log-likelihood function is suspected to exhibit multiple local optimums, a recommended approach is to run the EM algorithm several times, each instance initiated with a different $\boldsymbol{\theta}_{\text{old}}$. Subsequently, the MLE of $\boldsymbol{\theta}$ is chosen from the collection of local maxima yielded by these distinct EM algorithm runs, ensuring a more robust outcome.

Relation to VB. VB methods are frequently compared to EM methods because of their analogous procedures involving alternating iterative procedures that successively converge on some optimum parameter values. [35] highlighted that the first two terms of the ELBO in Equation (2.47) correspond to the expected complete log-likelihood, aligning with the core optimization goal of the EM algorithm. However, a key distinction lies in the fact that EM operates under the assumption that the ELBO equals the log-likelihood (or log evidence) $p(\mathbf{y})$ when $q(\theta = \mathbf{Z}) = p(\theta = \mathbf{Z}|y)$. EM follows an iterative procedure that alternates between computing the expected complete log-likelihood according to $p(\mathbf{Z}|\mathbf{y}, \boldsymbol{\theta})$ (E-step) and optimizing it with respect to the model parameters, $\boldsymbol{\theta}$ (M-step). Unlike VB, EM assumes that the expectation under $p(\mathbf{Z}|\mathbf{y}, \boldsymbol{\theta})$ is tractable. Conversely, unlike EM, VB doesn't inherently estimate fixed model parameters denoted as θ here. VB treats both model parameters and latent variables as interchangeable unobserved random variables, aiming to approximate their joint posterior distribution.

MAP estimation is not invariant to reparameterisation The MAP estimate, also known as the posterior mode, is the most popular point estimate of an unknown parameter $\boldsymbol{\theta}$. This preference arises because this estimation problem often reduces to a solvable optimization task for which efficient algorithms are readily available. Given some data $\mathbf{y}$, where the data likelihood is defined by $p(\mathbf{y}|\boldsymbol{\theta})$, and a prior distribution $\pi(\boldsymbol{\theta})$ is placed on the parameter, it is important to note that the parameterization of $\boldsymbol{\theta}$ is not unique. Any one-to-one transformation, denoted as $\boldsymbol{\Phi} = f(\boldsymbol{\theta})$, where $\boldsymbol{\theta} = f^{-1}(\boldsymbol{\Phi})$, results in an equivalent parameterization $\boldsymbol{\Phi}$.

In the case where $\boldsymbol{\theta}$ is a discrete random variable, and $\boldsymbol{\Phi} = f(\boldsymbol{\theta})$, the probability mass function for $\boldsymbol{\Phi}$ can be computed by summing the probability masses corresponding to all values of $\boldsymbol{\theta}$ that map to a given value ϕ using $f(t) = \phi$. This is expressed as:

$$p(\boldsymbol{\Phi} = \phi) = \sum_{t: f(t)=\phi} p(\boldsymbol{\theta} = t) \tag{2.55}$$

However, when $\boldsymbol{\theta}$ is a continuous variable, this summation is no longer applicable because $p(\boldsymbol{\theta} = t)$ is now a probability density function, not a probability mass function. This distinction arises because the probability of $\boldsymbol{\theta}$ taking on any specific real number is zero. Instead, a density function describes how densely the probability is distributed across an interval $\boldsymbol{\theta} \in (a, b)$, given by:

$$p(a < \boldsymbol{\theta} < b) = \int_b^a p(t)dt.$$

For small dt, $p(a < \boldsymbol{\theta} < a + dt) \approx p(t)dt$. Consequently, when we compute the posterior for $\boldsymbol{\theta}$ and define $\boldsymbol{\Phi} = f(\boldsymbol{\theta})$, the distribution for $\boldsymbol{\Phi}$ is determined by:

$$p(\boldsymbol{\Phi})d\boldsymbol{\Phi} = p(\boldsymbol{\theta})d\boldsymbol{\theta} \tag{2.56}$$

$$p(\boldsymbol{\Phi}) = p(\boldsymbol{\theta}) \left| \frac{d\boldsymbol{\theta}}{d\boldsymbol{\Phi}} \right| \tag{2.57}$$

Here, $|\frac{d\boldsymbol{\theta}}{d\boldsymbol{\Phi}}|$ represents the Jacobian term, which quantifies the change in the size of the unit volume transformed by the function f. Murphy [128] provided examples illustrating that the mode of the transformed distribution does not correspond to the transformation of the mode of the original distribution. Changing the parameterization from one representation to an equivalent one results in a change in the MAP estimate. This can be problematic, particularly when dealing with measurements in arbitrary units.

Chapter 3

Bayesian Penalized Linear Regression via EM

In this chapter, we introduce a novel expectation-maximization (EM) procedure for computing the Maximum A Posteriori (MAP) estimates of the model parameters in a penalized linear model. This approach combines two key concepts: (i) Bayesian penalized linear models tend to conform to the hierarchical structure outlined in (3.1-3.2), where the prior on the regression coefficient $\boldsymbol{\beta}$ often has a scale mixture of normals representations. This implies that $\boldsymbol{\beta}$ follows a normal distribution while the choice of the mixing density on the shrinkage parameters defines the type of penalty applied to the linear model; (ii) we treat $\boldsymbol{\beta}$ as the latent variable, compute its expected value (E-step), and optimize for the shrinkage parameter (M-step). Due to the consistent presence of a normal distribution prior on $\boldsymbol{\beta}$ under (3.2), a straightforward expectation of $\boldsymbol{\beta}$ always exists. A further strength of our proposed approach is that the M-step depends only on the nature of the prior placed on the shrinkage parameters and it is independent of the form of the likelihood. We also introduce several simple modifications of this EM procedure that allow for straightforward extension to generalized linear models. In Chapter 4, we demonstrate the application of this EM procedure to Bayesian Ridge regression, and in Chapter 5, we explore its adaptation to the horseshoe prior and Laplace prior. For both of these applications, we presented experimental results on simulated and real-world data. Notably, our approach demonstrates comparable or even superior performance to state-of-the-art equivalent estimation methods in terms of statistical efficacy and computational efficiency.

3.1 Introduction

Shrinkage methods have gained significance in statistical learning due to the growing demand for the analysis of high-dimensional data with the number of parameters exceeding the sample size. In such scenarios, it is often assumed that the parameter vector is sparse, implying that many components within the vector are considered insignificant and should be excluded (i.e. parameter estimates of exactly zero) for good model estimation. This need for sparsity has led to the preference for sparsity-inducing shrinkage methods like the lasso [165], smoothly clipped absolute deviation (SCAD) [61] and the minimax concave penalty (MCP) [190]. These methods are known for their ability to simultaneously perform variable selection and coefficient estimation. However, these approaches primarily adhere to a frequentist perspective, typically relying on an additional principle such as cross-validation to select the degree of regularisation. They often lack immediate access to reliable measures of uncertainty to complement the point estimates they produce [43]. In contrast, Bayesian inference naturally quantifies uncertainty directly through the posterior distribution. Bayesian penalized regression offers readily available uncertainty estimates, automatic estimation of the penalty parameters, and more flexibility in terms of penalties that can be considered [173].

In a Bayesian linear regression model, the standard linear regression model (1.2) corresponds to the following likelihood

$$\mathbf{y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2 \quad \sim \quad N_n \left(\mathbf{X}\boldsymbol{\beta}, \; \sigma^2 \boldsymbol{I}_n \right) \tag{3.1}$$

where $N_k(\cdot,\cdot)$ is the k-variate Gaussian distribution. The key distinction between Bayesian and standard linear regression lies in their treatment of uncertainty. Standard linear regression provides point estimates for parameters, assuming they are fixed and known exactly, while Bayesian regression treats parameters as random variables and provides a full probability distribution over the possible values of these parameters. In this context, prior distributions are assigned to the regression coefficients $\boldsymbol{\beta}$, leading to distinct forms of penalties. For instance, the Lasso estimates 2.5 and the ridge estimates 2.10 correspond to the posterior mode estimates under i.i.d. Laplace priors and Gaussian priors respectively. These priors, including other widely used shrinkage priors (as introduced in the literature review Chapter 2.2), fall in the class of global-local shrinkage priors with each β_j assigned to a continuous shrinkage prior centered at $\beta_j = 0$ that can be represented in an SMN form as follows:

$$\begin{aligned} \beta_j \,|\, \tau^2, \lambda_j^2, \sigma^2 &\sim N\left(0, \; \tau^2 \lambda_j^2 \sigma^2\right) \\ \lambda_j^2 &\sim \pi(\lambda_j^2) d\lambda^2 \\ \tau^2 &\sim \pi(\tau^2) d\tau^2. \end{aligned} \tag{3.2}$$

Here, τ represents the *global* shrinkage parameter that controls the overall degree of shrinkage, λ_j is the *local* shrinkage parameter associated with the j-th predictor and it controls the shrinkage for individual coefficients, and $\pi(\cdot)$ is an appropriate prior distribution of choice assigned to the shrinkage parameters. The previously mentioned Laplace prior, denoted as $\boldsymbol{\beta} \sim \text{La}(0, \sigma\tau)$, can be decomposed into hierarchy 3.2 by introducing an exponential mixing density with $\lambda_j^2 \sim \text{Exp}(2)$.

While many well-known shrinkage priors may promote sparsity by concentrating probability around sparse coefficient vectors, the resulting posterior means or medians will never themselves be sparse. Sparse estimates can be achieved by instead considering the posterior mode, i.e., maximum a posteriori (MAP) estimation [13, 28, 62]. However, most implementations of Bayesian MAP-based sparse estimators are not flexible in the sense that they only work for linear models when paired with a specified prior. These methods do not easily generalize to other regression models and are difficult to apply to priors that lack an analytic form (e.g., horseshoe prior, Strawderman-Berger prior, and normal-Gamma prior). Alternatively, simple thresholding ("sparsification") rules for the posterior mean or median of β_js can be used to produce sparse estimates, but these methods often lack theoretical justification, and inference can be highly sensitive to the choice of the threshold [42, 104]. To address these limitations:

1. we propose a novel expectation-maximization (EM) procedure to solve for the *exact* posterior mode of a regression model applicable to any priors with scale mixture normals of the form 3.2. We provide detailed applications to the ridge prior in Chapter 4 and to the horseshoe prior in Chapter 5.

2. we introduce several simple modifications of our base EM procedure that allow for straightforward extension to models beyond the usual Gaussian linear model, e.g., generalized linear models.

3.2 Background and Related Work

The EM algorithm [54] is one of the most widely used methods for MAP estimation of sparse Bayesian linear models, with [62] and [98] being classic papers on this approach. A particular strength, in comparison to approximate methods such as variational Bayes [95, 176], is that it is guaranteed to converge to exact posterior modes (a stationary point in the likelihood) whenever it can be applied [54]. By introducing appropriate latent variables, Figueiredo [62] uses the hierarchical decomposition of the Laplace prior to derive an efficient EM algorithm for estimating sparse lasso estimates. Following this work, many authors have attempted similar procedures to achieve sparse estimation using various priors and likelihood models that admit a scale mixture of normals (SMN) representation. This includes the lasso prior [134], the generalised double Pareto prior [13] and the horseshoe-like prior [28].

Given the (unnormalized) joint posterior distribution of the hierarchy (3.1)-(3.2),

$$p(\boldsymbol{\beta}, \tau, \boldsymbol{\lambda}|\mathbf{y}) \;\propto\; p(\mathbf{y}|\boldsymbol{\beta}) \cdot \pi(\boldsymbol{\beta}|\boldsymbol{\lambda}, \tau) \cdot \pi(\boldsymbol{\lambda}) \cdot \pi(\tau), \tag{3.3}$$

the conventional EM approach to find the posterior mode estimates treats the hyperparameters (i.e., the shrinkage parameters) $\boldsymbol{\lambda}$ as "missing data" (i.e., the latent variables). This approach iteratively finds the expected values of the latent variables and solves the maximization problem

$$\underset{\boldsymbol{\beta}}{\operatorname{argmax}}\ \mathbb{E}_{\boldsymbol{\lambda}}\left[\log p(\boldsymbol{\beta}, \tau, \boldsymbol{\lambda}|\mathbf{y}) \mid \hat{\boldsymbol{\beta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}, \mathbf{y}\right]$$

to produce a sequence of estimates for the regression coefficients $\boldsymbol{\beta}$. This approach is effective because the scale-mixture form of the local-global prior means that conditional on $\boldsymbol{\lambda}$, the posterior for $\boldsymbol{\beta}$ is Gaussian and maximization is (relatively) straightforward. In the case that the likelihood is non-Gaussian, but itself admits a scale-mixture-of-normal (SMN) representation, such as the Laplace or logistic regression model, this approach can be adapted appropriately.

However, the applicability of this approach is challenged when the prior assigned to $\boldsymbol{\beta}$ lacks a simple analytic form (e.g. the horseshoe prior). In such scenarios, the expected value of the shrinkage parameter, $\boldsymbol{\lambda}$, will not have a closed form and may be difficult or impossible to compute. To address this challenge, an alternative is to seek an approximated prior that not only mimics the behavior of the complex prior but also possesses a closed-form density function and a scale-mixture representation that allows for the implementation of the EM algorithm. A significant drawback, however, is that the accuracy of the estimates obtained using this approximated prior depends on how closely it resembles the original complex prior. For instance, consider the horseshoe prior [42]. To address its analytical complexity, [28] introduced a “horseshoe-like” prior, featuring a closed-form density function and a scale-mixture representation. In their work, [28] applied the conventional EM approach to obtain MAP estimates using this approximated prior. However, in Chapter 5, we demonstrate certain potential weaknesses in this method. Through experiments conducted on simulated and real data, our approach demonstrates comparable or superior statistical performance and computational efficiency when compared to the “horseshoe-like” prior method.

3.3 The Basic EM Algorithm

In this section, we present a novel, general EM procedure to compute the MAP estimate for the linear model. The key innovation underlying our approach is to treat the coefficients, $\boldsymbol{\beta}$, as latent variables; this is in contrast to the usual application of the EM algorithm [62] that treats the shrinkage hyperparameters, $\boldsymbol{\lambda}$, as missing data. As is standard in penalized regression, and without any loss of generality, we assume that the predictors are standardized to have mean zero and standard deviation one, and the target has a mean of zero, i.e., the estimate of the intercept is simply $\hat{\beta}_0 = (1/n)\sum y_i$. This means we can simply ignore the intercept when estimating the remaining coefficients $\boldsymbol{\beta}$.

Here we consider the linear regression model defined in the hierarchy (3.1)-(3.2). The proposed EM algorithm solves for the posterior mode estimates by iterating through the following two steps until convergence is achieved:

E-step. We take the expectation of the complete negative log-posterior (with respect to the missing variable $\boldsymbol{\beta}$), conditional on the current values of $\boldsymbol{\lambda}$, τ^2 and σ^2, and the observed data, $\mathbf{y}$; the resulting quantity is called the "Q-function":

$$\begin{aligned}
&Q(\boldsymbol{\lambda}, \tau, \sigma^2 | \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}) \\
&= \mathbb{E}_{\boldsymbol{\beta}} \left[-\log p(\boldsymbol{\beta}, \boldsymbol{\lambda}, \tau, \sigma^2 \,|\, \mathbf{y}) \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}, \mathbf{y} \right] \\
&= \left(\frac{n+p}{2} \right) \log \sigma^2 + \frac{\mathbb{E}_{\boldsymbol{\beta}} \left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right]}{2\sigma^2} + \frac{p}{2} \log \tau^2 \\
&\quad + \frac{1}{2} \sum_{j=1}^{p} \log \lambda_j^2 + \frac{1}{2\sigma^2 \tau^2} \sum_{j=1}^{p} \frac{\mathbb{E}_{\boldsymbol{\beta}} \left[\beta_j^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right]}{\lambda_j^2} - \log \pi(\boldsymbol{\lambda}, \tau) \qquad (3.4)
\end{aligned}$$

where $\pi(\boldsymbol{\lambda}, \tau)$ is the joint prior distribution for the hyperparameters. For notational simplicity, we use $\mathbb{E}\left[\beta_j^2\right]$ and $\mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right]$ for the conditional expectations of β_j^2 and the sum of squared residuals, respectively, throughout the sequel.

M-step. Update the parameter estimates by minimizing the Q-function with respect to the shrinkage hyperparameters and noise variance, i.e.,

$$\{\hat{\boldsymbol{\lambda}}^{(t+1)}, \hat{\tau}^{(t+1)}, \hat{\sigma}^{2^{(t+1)}}\} = \underset{\boldsymbol{\lambda}, \tau, \sigma^2}{\arg\min} \left\{ Q\left(\boldsymbol{\lambda}, \tau, \sigma^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right) \right\} \qquad (3.5)$$

Implementation of this EM algorithm requires only knowledge of the negative log-prior of choice $-\log \pi(\boldsymbol{\lambda}, \tau)$, the conditional expectations $\mathbb{E}\left[\boldsymbol{\beta}^{\mathbf{2}}\right]$ and $\mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right]$. In Section 3.4, we discuss several different approaches to compute the conditional expectations for the E-step. This algorithm is quite general, with only the M-step depending on the choice of prior for the coefficients β_j. Application to the specific case of the ridge prior, the horseshoe prior and the Laplace prior is discussed in Chapter 4, Chapter 5.1 and Chapter 5.2 respectively.

As outlined in Algorithm 1, the overall algorithm iterates the E-step and M-step until a convergence criterion is satisfied. Once convergence is achieved, we use the mode of the posterior distribution of $\boldsymbol{\beta}$, conditional on the final values of $\hat{\boldsymbol{\lambda}}^{(t)}$, $\hat{\tau}^{(t)}$ and $\hat{\sigma}^{2^{(t)}}$ as our point estimate. Given that this conditional distribution is Gaussian, the mode is just the mean of the normal distribution (3.7) given by (3.9). If a sparse estimate is desired, we set components of the final conditional posterior mode that are very small (smaller in absolute value than a small fraction of the standard error) to exactly zero. While this step is not strictly necessary, as the posterior mode will eventually converge to exact sparsity with enough iterations, it significantly reduces run time. In this thesis, we used $|\beta_j| < (5\sqrt{n})^{-1}$ as the threshold for all experiments.

Algorithm 1: Basic EM algorithm

```
Input      : Standardised predictors $\mathbf{X} \in \mathbb{R}^{n \times p}$, centered targets $\mathbf{y} \in \mathbb{R}^n$
Initialise :
    1. $\boldsymbol{\beta}^{(0)} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$
    2. $\mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right] = 10^{10}$
    3. $\delta = 10^{10}$
while $\delta < \epsilon$ do          // we set the tolerance parameter, $\epsilon$ to $10^{-5}$
    // E-step
    Compute the conditional expectation of $\mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right]$ and $\mathbb{E}\left[\boldsymbol{\beta}^2\right]$ using either one
      of the approaches described in Section 3.4.
    // M-step
    Update the shrinkage parameters $\hat{\boldsymbol{\lambda}}^2$, $\hat{\tau}^2$ and $\hat{\sigma}^2$ by solving the optimisation problem
      detailed in Section 3.3.
    $\hat{\boldsymbol{\beta}} = \mathbb{E}\left[\boldsymbol{\beta}\right]$          // Refer to equation (3.9)
    $\delta = \sum_{j=1}^{p}(|\hat{\beta}_j^{(t)} - \hat{\beta}_j^{(t+1)}|)/(1 + \sum_{j=1}^{p}(|\hat{\beta}_j^{(t+1)}|))$
end while
if $|\hat{\beta}_j| < (5\sqrt{n})^{-1}$ then
    $\hat{\beta}_j = 0$
end if
return $\hat{\boldsymbol{\beta}}$
```

MAP estimation is not invariant to reparameterization As highlighted in Chapter 2.3.3.2, a subtle issue inherent to MAP estimation is its sensitivity to the chosen parameterization of the probability distribution. Changing the parameterization from one representation to an equivalent parameterization results in a change in the MAP estimate. The idea behind the proposed EM algorithm is not to directly determine the posterior mode of $\boldsymbol{\beta}$; instead, we treat it as missing data, find its expectation (integrating it out of the equation), and then solve for the posterior mode estimate of the conditional posterior distribution of the hyperparameters $\boldsymbol{\lambda}, \tau$ and σ. We subsequently plug these estimates back into the well-known $\mathbb{E}\left[\boldsymbol{\beta}|\boldsymbol{\lambda}_{\mathrm{MAP}}, \tau_{\mathrm{MAP}}, \sigma^2_{\mathrm{MAP}}\right]$ given in Equation (3.9) which will be the resulting posterior mode estimate for $\boldsymbol{\beta}$. In this context, we are essentially addressing the minimization problem as follows:

$$\begin{aligned} &\underset{\boldsymbol{\lambda},\tau,\sigma^2}{\operatorname{argmin}}\ \mathbb{E}_{\boldsymbol{\beta}}\left[-\log p(\boldsymbol{\beta}, \tau, \boldsymbol{\lambda}|\mathbf{y}) \mid \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}, \mathbf{y}\right] \\ &= \underset{\boldsymbol{\lambda},\tau,\sigma^2}{\operatorname{argmin}}\ \left\{\int -\log\ p(\boldsymbol{\beta}, \boldsymbol{\lambda}, \tau, \sigma^2|\mathbf{y})p(\boldsymbol{\beta}|\hat{\boldsymbol{\lambda}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}})d\boldsymbol{\beta}\right\} \\ &= \underset{\boldsymbol{\lambda},\tau,\sigma^2}{\operatorname{argmin}}\ \left\{-\log\ p(\boldsymbol{\lambda}, \tau, \sigma^2|\mathbf{y})\right\} \end{aligned} \tag{3.6}$$

Consequently, we observe that given any one-to-one transformation of $\boldsymbol{\lambda}$, denoted as $f(\boldsymbol{\lambda})$, $\mathbb{E}\left[\boldsymbol{\beta}|\boldsymbol{\lambda},\tau,\sigma^2\right] \neq \mathbb{E}\left[\boldsymbol{\beta}|f(\boldsymbol{\lambda}),\tau,\sigma^2\right]$. Although altering the parameterization of τ and σ^2 also results in different $\boldsymbol{\beta}$ estimates, its impact seems less significant as compared to reparameterizing $\boldsymbol{\lambda}$. This is likely because τ and σ^2 involve single parameter estimates, while $\boldsymbol{\lambda}$ influences every $\boldsymbol{\beta}$ coefficient, making it more influential and sensitive to changes in parameterization.

In Chapter 5.3, we explore the properties of the estimates resulting from different parameterizations of $\boldsymbol{\lambda}$ for the horseshoe and lasso priors. These parameterizations include $\boldsymbol{\lambda}$, $\boldsymbol{\lambda}^2$, and a logarithmic transformation. Our analysis reveals different sets of estimates for each parameterization, each exhibiting varying sparsity level.

3.4 Computing the conditional expectations

The conditional expected values $\mathbb{E}\left[\boldsymbol{\beta}^{\mathbf{2}}\right]$ and $\mathbb{E}\left[||\mathbf{y}-\mathbf{X}\boldsymbol{\beta}||^2\right]$ depend on the conditional posterior distribution of the regression coefficients $\boldsymbol{\beta} \in \mathbb{R}^p$ [112]:

$$\begin{aligned}\boldsymbol{\beta}\,|\,\boldsymbol{\lambda},\tau,\sigma,\mathbf{y} &\sim N_p(\mathbf{A}^{-1}\mathbf{X}^T\mathbf{y},\sigma^2\mathbf{A}^{-1}) \\ \mathbf{A} &= (\mathbf{X}^T\mathbf{X}+\boldsymbol{\Lambda}_*^{-1}) \\ \boldsymbol{\Lambda}_* &= \tau^2\boldsymbol{\Lambda}\end{aligned} \tag{3.7}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1^2,\cdots,\lambda_{\mathrm{p}}^2)$. One important point to note here is that the posterior distribution of $\boldsymbol{\beta}$ does not depend on the choice of prior applied to λ_j. This implies that regardless of the (marginal) prior assigned to $\boldsymbol{\beta}$, as long as it has an SMN representation, both $p(\mathbf{y}|\boldsymbol{\beta})$ and $\pi(\boldsymbol{\beta}|\boldsymbol{\lambda},\tau)$ will always be Gaussian densities, e.g., it does not matter whether the prior assigned to $\boldsymbol{\beta}$ is a Laplace prior (to solve for the Lasso) or a horseshoe prior, the E-step will remain the same. Changing the marginal priors on the regression coefficients only requires straightforward modification to the M-step for updates on the shrinkage parameters (see Table 5.1 for a comprehensive summary of all M-step updates corresponding to the different priors explored in this thesis). This is an interesting advantage of our EM approach as the E-step is frequently very difficult to implement, particularly when dealing with conditional distributions that lack a standard density.

3.4.1 Exact expectations.

The expected value of β_j^2 can be solved for by taking the sum of the variance and the square of the expected value of β_j:

$$\mathbb{E}\left[\beta_j^2\,|\,\boldsymbol{\lambda}^{(t)},\tau^{(t)}\right] = \mathrm{Var}[\beta_j] + \mathbb{E}\left[\beta_j\right]^2. \tag{3.8}$$

Due to the properties of Gaussian distributions, $\mathrm{Var}[\beta_j] = (\mathrm{Cov}[\boldsymbol{\beta}])_{j,j}$, and

$$\begin{aligned} \mathrm{Cov}[\boldsymbol{\beta}] &= \sigma^2 \mathbf{A}^{-1}, \\ \mathbb{E}\left[\beta_j\right] &= \left(\mathbf{A}^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}\right)_j. \end{aligned} \tag{3.9}$$

Direct computation of this expectation value is potentially computationally expensive and numerically unstable because it involves inverting the $p \times p$ matrix, $\mathbf{A}$. Rue's efficient algorithm [150] avoids explicitly computing the inverse of $\mathbf{A}$ when sampling from multivariate normal distributions of the form (3.7). Subsequently, Bhattacharya et. al [32] proposed a similar approach, leveraging the matrix inversion lemma for improved computational complexity when $p > n$. Both of these algorithms can be trivially modified to compute the exact mean and covariance matrix of the conditional posterior distribution of the hierarchy (3.7). The details are provided below, with clarification on notation: $a\backslash b$ denotes the back-solve operation, and a/b denotes the forward-solve:

Rue's algorithm. The basic idea for Rue's algorithm is to use the Cholesky decomposition of the covariance matrix $\mathbf{A^{-1}}$ and solve a series of linear systems, resulting in the computation of (3.8) as follows:

1. Compute the upper triangular Cholesky decomposition of the covariance matrix $\mathbf{A} = \mathbf{U}^{\mathrm{T}}\mathbf{U}$.
2. Compute $\mathbb{E}\left[\boldsymbol{\beta}\right] = \mathbf{U}\backslash\left[\mathbf{U}^{T}/\left(\mathbf{X}^{\mathrm{T}}\mathbf{y}\right)\right]$.
3. If the exact expectation is required, $\mathbf{C} = \mathbf{U}^{\mathrm{T}}/\mathbf{I}_p$, $\mathrm{Cov}[\boldsymbol{\beta}] = \mathbf{U}\backslash\mathbf{C}$, and $\mathrm{Var}[\beta_j] = \sigma^2||\mathbf{c}_j||_2^2$, otherwise, refer to the expressions for the approximate expectations detailed in Section 3.4.2.

Bhattacharya's algorithm. By using the matrix inversion lemma (also known as Woodbury matrix identity), Bhattacharya's algorithm reformulates the conditional posterior (3.9) as follows:

$$\mathrm{Cov}[\boldsymbol{\beta}] = \sigma^2\left(\boldsymbol{\Lambda}_* - \boldsymbol{\Lambda}_*\mathbf{X}^{\mathrm{T}}\mathbf{W}^{-1}\mathbf{X}\boldsymbol{\Lambda}_*\right) \tag{3.10}$$

$$\mathbb{E}\left[\beta_j\right] = \left[\boldsymbol{\Lambda}_*\mathbf{X}^{\mathrm{T}}\mathbf{W}^{-1}\mathbf{y}\right]_j \tag{3.11}$$

whereby $\mathbf{W} = \mathbf{I} + \mathbf{X}\boldsymbol{\Lambda}_*\mathbf{X}^{\mathrm{T}}$. To compute these posterior statistics:

1. Compute the upper triangular Cholesky decomposition of $\mathbf{W} = \mathbf{w}^{\mathrm{T}}\mathbf{w}$.
2. Compute $\mathbb{E}\left[\boldsymbol{\beta}\right] = \boldsymbol{\Lambda}_*\mathbf{X}^{\mathrm{T}}(\mathbf{w}\backslash\left[\mathbf{w}^{\mathrm{T}}/\mathbf{y}\right])$.
3. If the exact expectation is required, $\mathbf{C} = \mathbf{w}\backslash\left[\mathbf{w}^{\mathrm{T}}/\mathbf{X}\boldsymbol{\Lambda}_*\right]$, $\mathrm{Cov}[\boldsymbol{\beta}] = \sigma^2\left(\boldsymbol{\Lambda}_* - \boldsymbol{\Lambda}_*\mathbf{X}^{\mathrm{T}}\mathbf{C}\right)$, and $\mathrm{Var}[\beta_j] = \sigma^2\left(\boldsymbol{\Lambda}_{*jj} - \boldsymbol{\Lambda}_{*jj}\sum_{i=1}^{n}[\mathbf{x}_j\mathbf{c}_j]_i\right)$, otherwise, refer to the expressions for the approximate expectations detailed in Section 3.4.2.

Rue's algorithm involves solving for the Cholesky factorization of $\mathbf{A}$ with complexity $O(p^3)$. Bhattacharya et.al claim that when $\mathbf{\Lambda}_*$ is diagonal, as in the case of the global-local priors (2.22), the complexity of Bhattacharya's algorithm is $O(n^2p)$.

Similarly, the computation of $\mathbb{E}\left[||\mathbf{y}-\mathbf{X}\boldsymbol{\beta}||^2\right]$ involves solving for the expected value of a quadratic form of the regression coefficients, which in turns requires inverting the same matrix, $\mathbf{A}$. From standard results involving expectations of quadratic forms we have:

$$\begin{aligned}\mathbb{E}\left[||\mathbf{y}-\mathbf{X}\boldsymbol{\beta}||^2\right] &= \mathbb{E}\left[\mathbf{y}^{\mathrm{T}}\mathbf{y}-2\mathbf{y}^{\mathrm{T}}\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\boldsymbol{\beta}\right]\\ &= \mathbf{y}^{\mathrm{T}}\mathbf{y}-2\,\mathbf{y}^{\mathrm{T}}\mathbf{X}\mathbb{E}\left[\boldsymbol{\beta}\right]+\mathbb{E}\left[\boldsymbol{\beta}\right]^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X})\mathbb{E}\left[\boldsymbol{\beta}\right]+\mathrm{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\;\mathrm{Cov}[\boldsymbol{\beta}])\\ &= ||\mathbf{y}-\mathbf{X}\mathbb{E}\left[\boldsymbol{\beta}\right]||^2+\mathrm{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\cdot\mathrm{Cov}[\boldsymbol{\beta}]) \qquad (3.12)\end{aligned}$$

where $\mathrm{tr}(\cdot)$ is the usual trace operator.

3.4.2 Approximate expectations.

The two main components required to compute the expectations in the E-step for this linear model are the conditional posterior mean and variance of the regression coefficients $\boldsymbol{\beta}$ under the distribution (3.7). The conditional posterior mean can be found using the Rue's [150] or Bhattacharya's [32] algorithm without having to explicitly solve for the inverse of $\mathbf{A}$, with a simple extension of the same procedures allowing us to solve for the exact covariance matrix $\sigma^2\mathbf{A}^{-1}$. However, the quantities that we need are the conditional variances (i.e., the diagonal elements of the conditional covariance matrix). Therefore, instead of inverting the full covariance matrix $\mathbf{A}$, the conditional variances may be approximately found by inverting only the diagonal elements of $\mathbf{A}$, i.e.,

$$\begin{aligned}\mathrm{Var}[\beta_j] &\approx \sigma^2\frac{1}{A_{j,j}}\\ &= \sigma^2\left(||\mathbf{x}_j||^2+\frac{1}{\tau^2\lambda_j^2}\right)^{-1}.\end{aligned} \qquad (3.13)$$

where $\mathbf{x}_j$ is the j-th column of $\mathbf{X}$. This approximation takes only $O(p)$ operations. In the specific case that the predictors $\mathbf{X}=(\mathbf{x}_1,\cdots,\mathbf{x}_p)$ are orthogonal, $\mathbf{X}^T\mathbf{X}=c\,\mathbf{I}_p$ and the approximation (3.13) will recover the exact variance of β_j. This approximation also applies to (3.12), i.e., instead of computing the matrix multiplication of the components in the trace function, one may instead consider only the diagonal elements of the matrix, yielding the approximation

$$\mathrm{tr}(\mathbf{X}^T\mathbf{X}\cdot\mathrm{Var}[\boldsymbol{\beta}]) \approx \sigma^2\sum_{j=1}^{p}\left(||\mathbf{x}_j||^2+\frac{1}{\tau^2\lambda_j^2}\right)^{-1}||\mathbf{x}_j||^2. \qquad (3.14)$$

In comparison to the $O(p^2)$ operations required to compute the trace of the product of matrices in (3.12), this approximation requires only $O(p)$ operations.

3.4.3 Stochastic expectation

An alternative approach to address computationally inefficient or infeasible E-step procedures is to consider the stochastic EM (StEM) method [46]. StEM is a simulation-based method in which the E-step is replaced by a simulation step, commonly referred to as the S-step. Instead of taking expectations, StEM imputes a sample drawn from the conditional distribution of the missing data given the observed sample and current parameter.

In this work, we introduced a small modification to the StEM algorithm. Instead of replacing the expectation with a single sample drawn from the conditional posterior distribution, we estimate the current expectation using the exponentially weighted moving average (EWMA) [148] of all previous samples. Thus, to estimate $\mathbb{E}\left[\boldsymbol{\beta}^{\mathbf{2}}|\boldsymbol{\lambda}^{(t)},\tau^{(t)}\right]$, we define $\overline{b_j^2}^{(t+1)}$ as the approximation of this expectation at the $(t+1)^{\text{th}}$ EM recursion for $t \geq 0$. This approximation is computed as follows:

$$\overline{b_j^2}^{(t+1)} = \alpha\, \overline{b_j^2}^{(t)} + (1-\alpha)\, b_j^2, \qquad 0 \leq \alpha \leq 1 \tag{3.15}$$

where $\alpha = (\frac{t}{t+1})^{\psi}$ acts as the smoothing factor, with ψ represents the smoothing power and b_j is a newly drawn β sample at iteration $(t+1)$ from its conditional posterior distribution. When $\psi = 1$, the EWMA behaves similarly to a cumulative moving average, resulting in an equally weighted average of the sequence of b_j^2 samples up to the current time t. Increasing ψ gives more weight to recent samples for a longer period and disregards older observations faster. As the number of iterations t increases, α naturally approaches 1, causing the EWMA to become less responsive to new samples. This property ensures the convergence of our EM algorithm because fixing the smoothing factor to a constant value could lead to unstable estimates of the expectation values. Such instability might cause estimates to fluctuate a lot due to the randomness introduced during the sampling step, potentially resulting in slower convergence or even never converging at all. Therefore, it is recommended to set ψ to a large value, typically greater than 100, as smaller values of ψ may assign less weight to new samples too early in the EM iteration, prematurely halting the exploration of the conditional posterior distribution. This early termination could result in suboptimal solutions in the resulting MAP estimate. Applying the same concept to estimate $\mathbb{E}\left[||y-\mathbf{X}\boldsymbol{\beta}||^2\right]$, we get:

$$\overline{ERSS_j}^{(t+1)} = \alpha\, \overline{ERSS_j}^{(t)} + (1-\alpha)\, ||y-\mathbf{x_j}b_j||^2, \qquad 0 \leq \alpha \leq 1 \tag{3.16}$$

where $\overline{ERSS_j}^{(t+1)}$ represents the estimate of $\mathbb{E}\left[||y-\mathbf{X}\boldsymbol{\beta}||^2\right]$ at the $(t+1)^{\text{th}}$ EM iteration. This approach offers the advantage of requiring only a method to sample $\boldsymbol{\beta}$ from the conditional posterior distribution of the regression coefficient at each EM iteration to estimate all the conditional expectations in the E-step.

Unfortunately, the convergence of the EM algorithm appears to be highly sensitive to the initial parameter values. This sensitivity to initializations is a common issue, as noted by Bhadra et al. [28], who also encountered this challenge in their EM procedure. They highlighted that the absence of a unique solution is a result of the non-convex penalty itself, rather than an artifact caused by a failure of the optimization algorithm. The suggested solution is to rerun the EM procedure with different starting values whenever the algorithm converges to an uninteresting all-zero solution. However, we propose an alternative approach to improve the initialization of the EM algorithm. Specifically, we suggest running the sampler with a minimum of 100 burn-in samples and then using the average of the last k samples to initialize all the necessary parameters of the EM algorithm. This initialization strategy has shown promising results, providing a robust starting point. We have observed reduced sensitivity to the initial values when employing this initialization strategy within the StEM algorithm.

3.5 Extension to Generalised Linear Models

In this section, we extend the EM approach proposed in Section 3.3 to accommodate non-normal data models winthin the framework of generalized linear models (GLMs) [132]. GLMs offer a flexible framework for modeling the conditional mean of the target variable through an appropriate (potentially) non-linear transformation of the linear predictor. Well-known examples of GLMs include binomial logistic regression and Poisson log-linear regression. In general, when the targets are non-Gaussian, the conditional distribution of the coefficients will not have a standard form, and finding the exact expectations $\mathbb{E}\left[\beta_j^2\right]$ is difficult. However, it is often the case that the conditional distribution can be approximated by a heteroskedastic Gaussian distribution, as expressed by the following equation:

$$\begin{aligned} \boldsymbol{\beta} \,|\, \boldsymbol{\lambda}, \tau, \boldsymbol{\omega}, \mathbf{z} &\sim N_p\left(\mathbf{A}_{\boldsymbol{\omega}}^{-1}\mathbf{X}^T\boldsymbol{\Omega}\mathbf{z},\ \mathbf{A}_{\boldsymbol{\omega}}^{-1}\right), \\ \mathbf{A}_{\boldsymbol{\omega}} &= \mathbf{X}^T\boldsymbol{\Omega}\mathbf{X} + \tau^{-2}\boldsymbol{\Lambda}^{-1} \end{aligned} \tag{3.17}$$

where $\boldsymbol{\Omega} = \text{diag}(\boldsymbol{\omega})$, $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ is a vector of weights, and $\mathbf{z}$ is an adjusted version of the targets, $\mathbf{y}$. Via standard central-limit theorem arguments, the accuracy of this approximation increases as the sample size n grows. The weights can be obtained via a linearisation argument (i.e., the well-known IRLS algorithm) or, preferably, via a scale-mixture of normals representation of the likelihood when available. For example, the logistic regression implementation used in Section 5.1.2.2 utilizes the well-known Polyá-gamma representation of logistic regression [142];

under this scheme, the adjusted targets and weights are

$$z_i = (y - 1/2)/\omega_i, \;\; \omega_i = \left(\frac{1}{2\eta_i}\right) \tanh\left(\frac{\eta_i}{2}\right),$$

where $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \beta_0 \mathbf{1}_n$ is the linear predictor. Given appropriate weights and adjusted targets, one may simply approximate the conditional posterior-covariance of the coefficients by $\text{Cov}\,[\boldsymbol{\beta}] \approx \mathbf{A}_{\boldsymbol{\omega}}^{-1}$ and use either (3.9) or (3.13) to obtain approximate expressions for $\mathbb{E}\left[\beta_j^2\right]$. Alternatively, one could potentially utilize a stochastic variant of the EM algorithm [46] to compute the required expectations, as discussed in Section 3.4.3.

The stochastic approach is particularly valuable when dealing with cases where the data model lacks a known or convenient scale mixture representation. Without having access to the appropriate hierarchical decomposition of the non-Gaussian data model, finding the conditional expectation required in the E-step may be infeasible. As such, we could instead estimate the expectation through simulation. Referring to the E- and M-steps given in equations (3.4-3.5), all terms involving $E[\cdot|\cdot]$ can be replaced with the EWMA estimates. The range of regression problems that can be applied to this StEM algorithm depends on the sampling algorithm paired. By integrating this StEM algorithm with the `bayesreg` [111] Gibbs sampling algorithm, our proposed StEM algorithm can be extended to fit a wide range of generalized linear regression models which includes ridge, lasso, and horseshoe regression with logistic, Gaussian, Laplace, Student-t, Poisson or geometric distributed targets.

Chapter 4

Ridge Regression via Expectation-Maximization

Among the global-local shrinkage priors outlined in Chapter 2.2.2, the ridge prior is arguably the simplest, as it relies solely on a global shrinkage parameter and does not involve any local shrinkage. While this prior does not inherently promote sparsity, estimating the posterior mode of the model parameters remains an important problem. Ridge regression is a widely used technique that can outperform sparsity inducing priors in the right settings. In this Chapter, we adapt the EM algorithm introduced in Chapter 3 to Bayesian ridge regression, presenting a novel method for tuning the regularization hyper-parameter, γ, of ridge regression that is faster to compute than leave-one-out cross-validation (LOOCV). This proposed method yields ridge estimates of the regression parameters of equal, or particularly in the setting of sparse covariates, superior quality to those obtained by minimising the LOOCV risk. The LOOCV risk can suffer from multiple and bad local minima for finite n and thus requires the specification of a set of candidate γ, which can fail to provide good solutions. In contrast, we show that the proposed method is guaranteed to find a unique optimal solution for large enough n, under relatively mild conditions, without requiring the specification of any difficult to determine hyper-parameters. This is based on a Bayesian formulation of ridge regression that we prove to have a unimodal posterior for large enough n, allowing for both the optimal γ and the regression coefficients to be jointly learned within an iterative expectation maximization (EM) procedure. Importantly, we show that by utilizing an appropriate preprocessing step, a single iteration of the main EM loop can be implemented in $O(\min(n, p))$ operations, for input data with n rows and p columns. In contrast, evaluating a single value of γ using fast LOOCV costs $O(n \min(n, p))$ operations when using the same preprocessing. This advantage amounts to an asymptotic improvement of a factor of l for l candidate values for γ (in the regime $q, p \in O(\sqrt{n})$ where q is the number of regression targets).

4.1 Introduction

Ridge regression [86] is one of the most widely used statistical learning algorithms. Given training data $\mathbf{X} \in \mathbb{R}^{n \times p}$ and $\mathbf{y} \in \mathbb{R}^n$, ridge regression finds the linear regression coefficients $\hat{\boldsymbol{\beta}}_\gamma$ that minimize the ℓ_2-regularized sum of squared errors, i.e.,

$$\hat{\boldsymbol{\beta}}_\gamma = \operatorname*{arg\,min}_{\boldsymbol{\beta}} \left\{ ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \gamma ||\boldsymbol{\beta}||^2 \right\} . \tag{4.1}$$

In practice, using ridge regression additionally involves estimating the value for the tuning parameter γ that minimizes the expected squared error $\mathbb{E}(\mathbf{x}^{\mathrm{T}}\hat{\boldsymbol{\beta}}_\gamma - y)^2$ for new data $\mathbf{x}$ and y sampled from the same distribution as the training data. This problem is usually approached via the leave-one-out cross-validation (LOOCV) estimator, which can be computed efficiently by exploiting a closed-form solution for the leave-one-out test errors for a given γ. The wide and long-lasting use of the LOOCV approach suggests that it solves the ridge regression problem more or less optimally, both in terms of its statistical performance, as well as its computational complexity.

However, in this work, we show that LOOCV is outperformed by a simple expectation maximization (EM) approach based on a Bayesian formulation of ridge regression. While the two procedures are not equivalent, in the sense that they generally do not produce identical parameter values, the EM estimates tend to be of equal quality or, particularly in sparse regimes, superior to the LOOCV estimates (see Fig. 4.1). Specifically, the LOOCV risk estimates can suffer from potential multiple and bad local minima when using iterative optimization, or misspecified candidates when performing grid search. In contrast, we show that the EM algorithm finds a unique optimal solution for large enough n (outside pathological cases) without requiring any hard to specify hyper-parameters, which is a consequence of a more general bound on n (Thm. 4.1) that we establish to guarantee the unimodality of the posterior distribution of Bayesian ridge regression—a result with potentially wider applications. In addition, the EM procedure is asymptotically faster than the LOOCV procedure by a factor of l where l is the number of candidate values for γ to be evaluated (in the regime $p, q \in O(\sqrt{n})$ where p, q, and n are the number of covariates, target variables, and data points, respectively). In practice, even in the usual case of $q = 1$ and $l = O(1)$, the EM algorithm tends to outperform LOOCV computationally by an order of magnitude as we demonstrate on a test suite of datasets from the UCI machine learning repository and the UCR time series classification archive.

While the EM procedure discussed in this Chapter is based on a recently published procedure for learning sparse linear regression models [163] (i.e. the proposed EM procedure in Chapter 3), the adaption of this procedure to ridge regression has not been previously discussed in the literature. Furthermore, a direct adoption would lead to a main loop complexity of $O(p^3)$ that is uncompetitive with LOOCV. Therefore, in addition to evaluating the empirical accuracy and

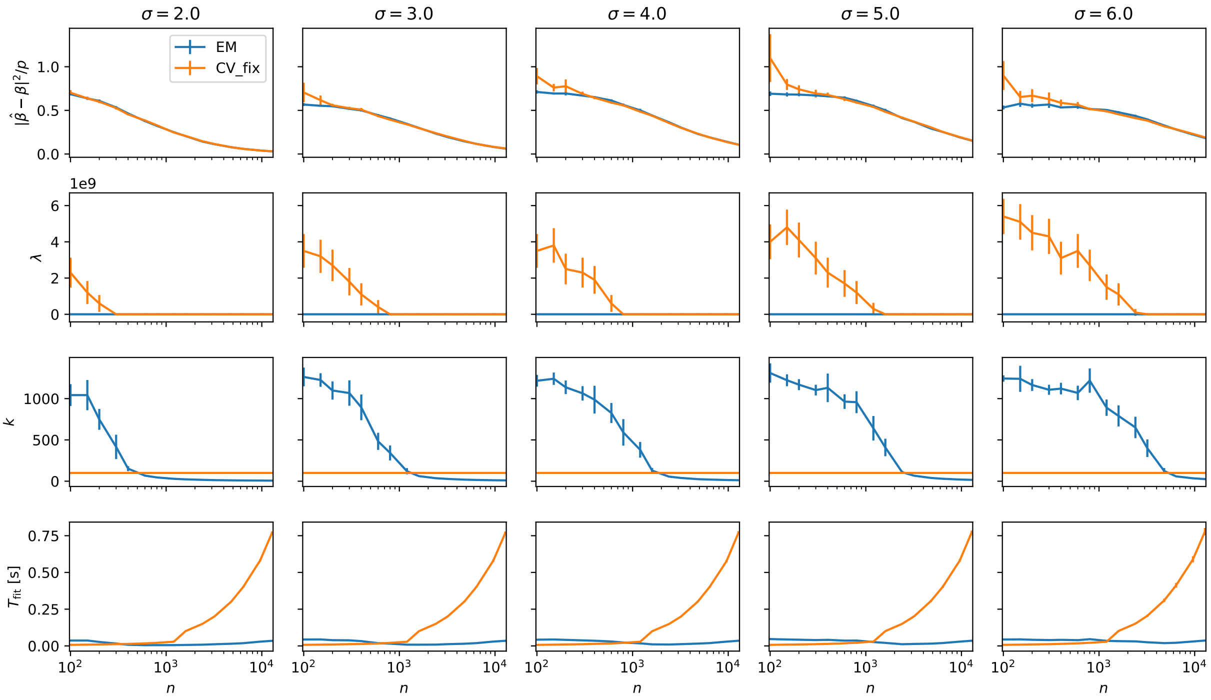


FIGURE 4.1: Comparison of LOOCV (with fixed candidate grid of size 100) and EM for setting with sparse covariate vectors of $\mathbf{x} = (x_1, \ldots, x_{100})$ such that $x_i \sim \text{Ber}(1/100)$ i.i.d. and responses $y|\mathbf{x} \sim N(\beta^{\text{T}}\mathbf{x}, \sigma^2)$ for increasing noise levels σ and sample sizes n. In an initial phase for small n, the number of EM iterations k tends to decrease rapidly from an initial large number until it reaches a small constant (around 10). In this phase, EM is computationally slightly more expensive than LOOCV (third row) but has a better parameter mean squared error (first row) corresponding to less shrinkage (second row). In the subsequent phase, both algorithms have essentially identical parameter estimates but EM outperforms LOOCV in terms of computation by a wide margin.

efficiency of the EM algorithm for ridge regression, the main technical contributions of this work are to show how certain key quantities can be efficiently computed from either a singular value decomposition of the design matrix, when $p \geq n$, or an eigenvalue decomposition of the Gram matrix $\mathbf{X}^{\text{T}}\mathbf{X}$, when $n > p$. This results in an E-step of the algorithm in time $O(r)$ where $r = \min(n, p)$, and an M-step found in closed form and solved in time $O(1)$, yielding an ultra-fast main loop for the EM algorithm.

These computational advantages result in an algorithm that is computationally superior to efficient, optimized implementations of the fast LOOCV algorithm. Our particular implementation of LOOCV actually outperforms the implementation in `scikit-learn` by approximately a factor of two by utilizing a similar preprocessing to the EM approach. This enables an $O(nr)$ evaluation for a single γ (which is still slower than the $O(r)$ evaluation for our new EM algorithm; see Table 4.1 for an overview of asymptotic complexities of LOOCV and EM), and may be of interest to readers by itself. Our implementation of both algorithms, along with all experiment code, are publicly available in the standard package ecosystems of the R and Python platforms, as well as on GitHub[1].

[1] https://github.com/marioboley/fastridge.git

Table 4.1: Time complexities of algorithms; $m = \max(n, p)$, $r = \min(n, p)$, l number of candidate γ for LOOCV, k number of EM iterations and q is the number of the target variables.

Method	main loop	pre-processing	Overall $(p, q \in O(\sqrt{n}))$
Naive adaption of EM	$O(kp^3q)$	$O(p^2n)$	$O(kn^2)$
Proposed BayesEM	$O(krq)$	$O(mr^2)$	$O(kn + n^2)$
Fast LOOCV	$O(lnrq)$	$O(mr^2)$	$O(ln^2)$

In the remainder of this chapter, we first briefly survey the literature of ridge regression with an emphasis on the use of cross validation (Sec. 4.2). Based on the Bayesian interpretation of ridge regression, we then introduce the EM algorithm and discuss its convergence (Sec. 4.3). Finally, we develop fast implementations of both the EM algorithm and LOOCV (Sec. 4.5) and compare them empirically (Sec. 4.6).

4.2 Ridge Regression and Cross Validation

Ridge regression [86] (also known as ℓ_2-regularization) is a popular method for estimation and prediction in linear models. The ridge regression estimates are the solutions to the penalized least-squares problem given in (4.1). The solution to this optimization problem is given by:

$$\hat{\boldsymbol{\beta}}_\gamma = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \gamma \mathbf{I}_p)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}. \tag{4.2}$$

When $\gamma \to 0$, the ridge estimates coincide with the minimum ℓ_2 norm least squares solution [82, 99], which simplifies to the usual least squares estimator in cases where the design matrix $\mathbf{X}$ has full column rank (i.e. $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ is invertible). Conversely, as $\gamma \to \infty$, the amount of shrinkage induced by the penalty increases, with the resulting ridge estimates becoming smaller for larger values of γ. Under fairly general assumptions [90], including misspecified models and random covariates of growing dimension, the ridge estimator is consistent and enjoys finite sample risk bounds for all fixed $\gamma \geq 0$, i.e., it converges almost surely to the prediction risk minimizer, and its squared deviation from this minimizer is bounded for finite n with high probability. However, its performance can still vary greatly with the choice of γ; hence, there is a need to estimate the optimal value from the given training data.

Earlier approaches to this problem [e.g. 8, 30, 59, 77, 84, 93, 96, 97, 127, 187] rely on an explicit estimate of the (assumed homoskedastic) noise variance, following the original idea of Hoerl and Kennard [86]. However, estimating the noise variance can be problematic, especially when p is not much smaller than n [73, 79, 85, 174]. More recent approaches adopt model selection criteria to select the optimal γ without requiring prior knowledge or estimation of the noise variance. These methods involve minimizing a selection criterion of choice, such as the Akaike information criterion (AIC) [3], Bayesian information criterion (BIC) [156], Mallow's conceptual prediction (C_p) criterion [116], and, most commonly, cross validation (CV) [12, 191].

A particularly attractive variant of CV is leave-one-out cross validation (LOOCV), also referred to as the prediction error sum of squares (PRESS) statistic in the statistics literature [9]

$$R_n^{\mathrm{CV}}(\gamma) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_{-i})^2 \tag{4.3}$$

where $\hat{y}_{-i} = \tilde{\mathbf{x}}_i\hat{\boldsymbol{\beta}}_\gamma^{-i}$, and $\hat{\boldsymbol{\beta}}_\gamma^{-i}$ denotes the solution to (4.2) when the i-th data point $(\tilde{\mathbf{x}}_i, y_i)$ is omitted. LOOCV offers several advantages over alternatives such as 10-fold CV: it is deterministic, nearly unbiased [177], and there exists an efficient "shortcut" formula for the LOOCV ridge estimate [135]:

$$R_n^{\mathrm{CV}}(\gamma) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{e_i}{1 - H_{ii}(\gamma)}\right)^2 \tag{4.4}$$

where $\mathbf{H}(\gamma) = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \gamma\mathbf{I}_p)^{-1}\mathbf{X}^{\mathrm{T}}$ is the regularized "hat", or projection, matrix and $\mathbf{e} = \mathbf{y} - \mathbf{H}(\gamma)\mathbf{y}$ are the residuals of the ridge fit using all n data points. As it only requires the diagonal entries of the hat matrix, Eq. (4.4) allows for the computation of the PRESS statistic with the same time complexity $O(p^3 + np^2)$ as a single ridge regression fit.

Moreover, unless $p/n \to 1$, the LOOCV ridge regression risk as a function of γ converges uniformly (almost surely) to the true risk function on $[0, \infty)$ and therefore optimizing it consistently estimates the optimal γ [82, 135]. However, for finite n, the LOOCV risk can be multimodal and, even worse, there can exist local minima that are almost as bad as the worst γ [161]. Therefore, iterative algorithms like gradient descent cannot be reliably used for the optimization, giving theoretical justification for the pre-dominant approach of optimizing over a finite grid of candidates $L = (\gamma_1, \ldots, \gamma_l)$. Unfortunately, despite the true risk function being smooth and unimodal, a naïvely chosen finite grid cannot be guaranteed to contain any candidate with a risk value close to the optimum. While this might not pose a problem for small n when the error in estimating the true risk via LOOCV is likely large, it can potentially become a dominating source of error for growing n and p. Therefore, letting l grow moderately with the sample size appears necessary, turning it into a relevant factor in the asymptotic time complexity.

As a further disadvantage, LOOCV (or CV in general) is sensitive to sparse covariates, as illustrated in Figure 4.1 where the performance of LOOCV, relative to the proposed EM algorithm, degrades as the noise variance σ^2 grows. In the sparse covariate setting, a situation common in genomics, the information about each coefficient is concentrated in only a few observations. As LOOCV drops an observation to estimate future prediction error, the variance of the CV score can be very large when the predictor matrix is very sparse, as the estimates depend on only a small number of the remaining observations. In the most extreme case, known as the normal multiple means problem [92], $\mathbf{X} = \mathbf{I}_n$, and all the information about each coefficient is concentrated in a single observation. In this setting, the LOOCV score reduces to $\sum y_i^2$, and provides no information about how to select γ. In contrast, the proposed EM approach explicitly ties together the coefficients via the probabilistic Bayesian interpretation of γ as the inverse-variance of the unknown coefficient vector. This "borrowing of strength" means that the procedure provides a sensible estimate of γ even in the case of multiple means (see Sec. 4.4.1).

4.3 Bayesian Ridge Regression

The ridge estimator (4.2) has a well-known Bayesian interpretation; specifically, if we assume that the coefficients are *a priori* normally distributed with mean zero and common variance $\tau^2\sigma^2$ we obtain a Bayesian version of the usual ridge regression procedure, i.e.,

$$\begin{aligned}
\mathbf{y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2 &\sim N_n\left(\mathbf{X}\boldsymbol{\beta},\ \sigma^2\mathbf{I}_n\right), \\
\boldsymbol{\beta} \,|\, \tau^2, \sigma^2 &\sim N_p\left(0,\ \tau^2\sigma^2\mathbf{I}_p\right), \\
\sigma^2 &\sim \sigma^{-2} d\sigma^2, \\
\tau^2 &\sim \pi(\tau^2) d\tau^2,
\end{aligned} \tag{4.5}$$

where $\pi(\cdot)$ is an appropriate prior distribution assigned to the variance hyperparameter τ^2. For a given $\tau > 0$ and $\sigma > 0$, the conditional posterior distribution of $\boldsymbol{\beta}$ is also normal [109]

$$\begin{aligned}
\boldsymbol{\beta} \,|\, \tau^2, \sigma^2, \mathbf{y} &\sim N_p(\hat{\boldsymbol{\beta}}_\tau,\ \sigma^2\mathbf{A}_\tau^{-1}), \\
\hat{\boldsymbol{\beta}}_\tau &= \mathbf{A}_\tau^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \\
\mathbf{A}_\tau &= (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \tau^{-2}\mathbf{I}_p),
\end{aligned} \tag{4.6}$$

where the posterior mode (and mean) $\hat{\boldsymbol{\beta}}_\tau$ is equivalent to the ridge estimate $\hat{\boldsymbol{\beta}}_\gamma$ with penalty $\gamma = 1/\tau^2$ (we rely on the variable name in the notation $\hat{\boldsymbol{\beta}}_x$ to indicate whether it refers to (4.6) or (4.2)).

Shrinkage Prior To estimate the τ^2 hyperparameter in the Bayesian framework, we first choose a prior distribution for the hypervariance τ^2. We assume that no strong prior knowledge on the degree of shrinkage of the regression coefficients is available, and instead assign the recommended default beta-prime prior distribution for τ^2 [66, 141] with probability density function:

$$\pi(\tau^2) = \frac{(\tau^2)^{a-1}(1+\tau^2)^{-a-b}}{B(a,b)},\ a > 0, b > 0, \tag{4.7}$$

where $B(a,b)$ is the beta function. Specifically, we choose $a = b = 1/2$, which corresponds to a standard half-Cauchy prior on τ. The half-Cauchy is a heavy-tailed, weakly informative prior that is frequently recommended as a default choice for scale-type hyperparameters such as τ

[141]. Further, this estimator is very insensitive to the choice of a or b. As demonstrated by Theorem 6.1 in [22], the marginal prior density over $\boldsymbol{\beta}$, $\int \pi(\boldsymbol{\beta}|\tau^2)\pi(\tau^2|a,b)d\tau^2 = \pi(\boldsymbol{\beta}|a,b)$ has polynomial tails in $\|\boldsymbol{\beta}\|^2$ for all $a > 0, b > 0$, and has Cauchy or heavier tails for $b \leq 1/2$. This type of polynomial-tailed prior distribution over the norm of the coefficients is insensitive to the overall scale of the coefficients, which is likely unknown *a priori*. This robustness is in contrast to other standard choices of prior distributions for τ^2 such as the inverse-gamma distribution [e.g., 134, 159] which are highly sensitive to the choice of hyperparameters [141].

Unimodality and Consistency The asymptotic properties of the posterior distributions in Gaussian linear models (4.5) have been extensively researched [36, 69, 70, 172]. These studies reveal that in linear models, the posterior distribution of $\boldsymbol{\beta}$ is consistent, and converges asymptotically to a normal distribution centered on the true parameter value. When p is fixed, this assertion can be established through the Bernstein-Von Mises theorem [172, Sec. 10.2]. Our specific problem (4.5) satisfies the conditions for this theorem to hold: 1) both the Gaussian-linear model $p(y|\boldsymbol{\beta},\sigma^2)$ and the marginal distribution $\int p(y|\boldsymbol{\beta},\sigma^2)\pi(\boldsymbol{\beta}|\tau^2)d\boldsymbol{\beta} = p(y|\tau^2,\sigma^2)$ are identifiable; 2) they have well defined Fisher information matrices; and 3) the priors over $\boldsymbol{\beta}$ and τ are absolutely continuous. Further, these asymptotic properties remain valid when the number of predictors p_n is allowed to grow with the sample size n at a sufficiently slower rate [36, 69].

The following theorem provides a simple bound on the number of samples required to guarantee that the posterior distribution for the Bayesian ridge regression hierarchy given by (4.5) has only one mode outside a small environment around zero.

Theorem 4.1. *Let $\epsilon > 0$, and let ζ_n be the smallest eigenvalue of $\mathbf{X}^{\mathrm{T}}\mathbf{X}/n$. If $\zeta_n > 0$ and $\epsilon > 4/(n\zeta_n)$ then the joint posterior $p(\boldsymbol{\beta},\sigma^2,\tau^2|\mathbf{y})$ has a unique mode with $\tau^2 \geq \epsilon$. In particular, if $\zeta_n \geq cn^{-\alpha}$ with $\alpha < 1$ and $c > 0$ then there is a unique mode with $\tau^2 \geq \epsilon$ if $n > (4/(c\epsilon))^{1/(1-\alpha)}$.*

Proof. For sufficiently large n, a continuous injective reparameterization of the negative log joint posterior of (4.5) & (4.7) is convex when restricted to $\tau^2 \geq \epsilon$. This is sufficient, since unimodality is preserved by strictly monotone transformations and continuous injective reparameterizations. Specifically, for the presented hierarchical model, the negative log joint posterior up to an additive constant is

$$\frac{n+p+2}{2}\log\sigma^2 + \frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|^2 + \frac{p+2-2a}{2}\log\tau^2 + \frac{\|\boldsymbol{\beta}\|^2}{2\sigma^2\tau^2} + (a+b)\log(1+\tau^2)$$

and reparameterizing with $\phi = \beta/\sigma$, $\rho = 1/\sigma$ and $\chi = 1/\tau$ and reorganising terms yields

$$-(n+p+2)\log\rho - (p+2-2a)\log\chi + (a+b)\log(1+\chi^{-2}) + \frac{1}{2}\|\rho\mathbf{y}-\mathbf{X}\boldsymbol{\phi}\|^2 + \frac{\|\chi\boldsymbol{\phi}\|^2}{2}\ .$$

The first three terms can easily be checked to be convex via a second derivative test, for which the convexity of the second term is contingent on the condition that $a < 1 + p/2$, a condition that holds true in our specific scenario with $a = b = 1/2$. For the last two terms, the combined Hessian is of block form $[\mathbf{A}, \mathbf{B}; \mathbf{B}^{\mathrm{T}}, \mathbf{C}]$ with $\mathbf{A} = \mathbf{X}^{\mathrm{T}}\mathbf{X} + \chi^2 \mathbf{I}_p$, $\mathbf{B} = 2\chi\boldsymbol{\phi}$, and $\mathbf{C} = \|\boldsymbol{\phi}\|^2$. Symmetric matrices of this form are positive definite if $\mathbf{A}$ and its Schur complement

$$\mathbf{C} - \mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B} = \|\boldsymbol{\phi}\|^2 - 4\boldsymbol{\phi}^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X}/\chi^2 + \mathbf{I}_p)^{-1}\boldsymbol{\phi}$$

are positive definite. Clearly, $\mathbf{A}$ is positive definite. Moreover, for $n > 4/(\epsilon^2 \zeta_n)$, we have

$$\begin{aligned}
\boldsymbol{\phi}^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X}/\chi^2 + I)^{-1}\boldsymbol{\phi} &= \boldsymbol{\phi}^{\mathrm{T}}(\mathbf{V}\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}/\chi^2 + I)^{-1}\boldsymbol{\phi} \\
&= \boldsymbol{\phi}^{\mathrm{T}}\mathbf{V}(\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}/\chi^2 + I)^{-1}\mathbf{V}^{\mathrm{T}}\boldsymbol{\phi} \\
&\leq \boldsymbol{\phi}^{\mathrm{T}}\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\phi}\chi^2/(n\zeta_n) \\
&= ((\mathbf{I} - \mathbf{V}\mathbf{V}^{\mathrm{T}})\boldsymbol{\phi} + \mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\phi})^{\mathrm{T}}\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\phi}\chi^2/(n\zeta_n) \\
&= \|\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\phi}\|^2\chi^2/(n\zeta_n) \\
&\leq \|\boldsymbol{\phi}\|^2\chi^2/(n\zeta_n) \\
&< \|\boldsymbol{\phi}\|^2/(\epsilon^2 n\zeta_n) \\
&< \|\boldsymbol{\phi}\|^2/4
\end{aligned}$$

where we used the fact that $\mathbf{V}\mathbf{V}^{\mathrm{T}}$ is the orthogonal projection onto the column space of $\mathbf{V}$. The overall inequality implies the required positivity of the Schur complement. □

In other words, all sub-optimal non-zero posterior modes vanish for large enough n if the smallest eigenvalue of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ grows at least proportionally to some positive power of n. This is a very mild assumption that is typically satisfied in fixed as well as random design settings, e.g., with high probability when the smallest marginal covariate variance is bounded away from zero.

4.4 Bayesian Ridge Regression via EM

Given the restricted unimodality of the joint posterior (4.5) for large enough n, in conjunction with its asymptotic concentration around the optimal $\boldsymbol{\beta}_0$, estimating the model parameters via an EM algorithm appears attractive, as they are guaranteed to converge to an exact posterior mode. In particular, in the non-degenerate case that $\boldsymbol{\beta}_0 \neq 0$, there exist $\tau^2 = \epsilon^2 > 0$, such that for large enough, but finite n, the posterior concentrates around $(\boldsymbol{\beta}_0, \tau^2)$, and thus $\boldsymbol{\beta}_0$ is identified by EM if initialized with a large enough τ^2.

Specifically, we use the novel approach [163] in which the coefficients $\boldsymbol{\beta}$ are treated as "missing data", and τ^2 and σ^2 as parameters to be estimated. Given the hierarchy (4.5), the resulting

Bayesian EM algorithm then solves for the posterior mode estimates of $\boldsymbol{\beta}$ by repeatedly iterating through the following two steps until convergence:

E-step. Find the parameters of the *Q-function*, i.e., the expected complete negative log-posterior (with respect to $\boldsymbol{\beta}$), conditional on the current estimates of $\hat{\tau}_t^2$ and $\hat{\sigma}_t^2$, and the observed data $\mathbf{y}$:

$$\begin{aligned} Q(\tau^2, \sigma^2 | \hat{\tau}_t^2, \hat{\sigma}_t^2) &= \mathbb{E}_{\boldsymbol{\beta}}\left[-\log p(\boldsymbol{\beta}, \tau^2, \sigma^2 \,|\, \mathbf{y}) \,|\, \hat{\tau}_t^2, \hat{\sigma}_t^2, \mathbf{y} \right] \\ &= \left(\frac{n+p+2}{2} \right) \log \sigma^2 + \frac{\text{ESS}}{2\sigma^2} + \frac{p+1}{2} \log \tau^2 + \frac{\text{ESN}}{2\sigma^2\tau^2} + \log(1+\tau^2) \end{aligned} \tag{4.8}$$

where the quantities to be computed are the (conditionally) expected sum of squared errors $\text{ESS} = \mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \,|\, \hat{\tau}_t^2, \hat{\sigma}_t^2 \right]$ and the expected squared norm $\text{ESN} = \mathbb{E}\left[\|\boldsymbol{\beta}\|^2 \,|\, \hat{\tau}_t^2, \hat{\sigma}_t^2 \right]$. Denoting by $\text{tr}(\cdot)$ the trace operator, one can show (see Sec. 4.4.2) that these quantities can be computed as

$$\text{ESS} = ||\mathbf{y} - \mathbf{X}\,\hat{\boldsymbol{\beta}}_\tau||^2 + \sigma^2 \text{tr}(\mathbf{X}^{\text{T}}\mathbf{X}\mathbf{A}_\tau^{-1}) \qquad \text{and} \qquad \text{ESN} = \sigma^2 \text{tr}(\mathbf{A}_\tau^{-1}) + \|\hat{\boldsymbol{\beta}}_\tau\|^2 \ . \tag{4.9}$$

M-step. Update the parameter estimates by minimizing the Q-function with respect to the shrinkage hyperparameter τ^2 and noise variance σ^2, i.e.,

$$\{\hat{\tau}_{t+1}^2, \hat{\sigma}_{t+1}^2\} = \underset{\tau^2, \sigma^2}{\arg\min} \left\{ Q\left(\tau^2, \sigma^2 \,|\, \hat{\tau}_t^2, \hat{\sigma}_t^2\right) \right\}. \tag{4.10}$$

Instead of numerically optimizing the two-dimensional Q-function (4.10), we can derive closed-form solutions for both parameters by first finding $\hat{\sigma}^2(\tau^2)$, i.e., the update for σ^2, as a function of τ^2, and then substituting this into the Q-function. This yields a Q-function that is no longer dependent on σ^2, and solving for $\hat{\tau}^2$ is straightforward. The resulting parameter updates in the M-step are given by:

$$\hat{\sigma}^2 = \frac{\tau^2 \text{ESS} + \text{ESN}}{(n+p+2)\tau^2} \qquad \text{and} \qquad \hat{\tau}^2 = \frac{(n-1)\text{ESN} - (1+p)\text{ESS} + \sqrt{g}}{(6+2p)\text{ESS}}, \tag{4.11}$$

where $g = (4n+4)\text{ESN}\,(3+p)\text{ESS} + ((1-n)\text{ESN} + (p+1)\text{ESS})^2$. The derivations of these formulae are presented in Section 4.4.2.

From (4.11), we see that updating the parameter estimates in the M-step requires only constant time. Therefore, the overall efficiency of the EM algorithm is determined by the computational complexity of the E-step. Computing the parameters of the Q-functions directly via (4.9) requires inverting $\mathbf{A}_\tau$, resulting in $O(p^3)$ operations. In Section 4.5, we show how to substantially improve this approach via singular value decomposition.

4.4.1 EM Procedure for the Normal Means Model

The Bayesian EM procedure provides sensible estimates of the regularization parameter even in the setting of the normal multiple means problem with known variance σ^2. In this setting, the LOOCV is unable to provide any guidance on how to choose γ due to all the information for each regression parameter being concentrated in a single observation. We use the τ parameterisation of the hyperparameter, rather than τ^2, as the resulting estimator has an easy to analyse form.

In the normal multiple means model, we are given $(y_i|\beta_i) \sim N(\beta_i, 1)$, i.e., $\mathbf{y}$ is a p-dimensional normally distributed vector with mean $\boldsymbol{\beta}$ and identity covariance matrix. The conditional posterior distribution of $\boldsymbol{\beta}$ is:

$$\boldsymbol{\beta}|\mathbf{y},\tau \;\sim\; N\left((1-\kappa)\boldsymbol{y}, \sigma^2(1-\kappa)\right) \tag{4.12}$$

where $\kappa = 1/(1+\tau^2)$. Under this setting, Strawderman [162] proved that if $p \geq 3$, then any estimator of the form

$$\left(1 - r\left(\frac{1}{2}||y||^2\right)\frac{p-2}{||y||^2}\right) y \tag{4.13}$$

where $0 \leq r\left(\frac{1}{2}||y||^2\right) \leq 2$ and $r(\cdot)$ is non-decreasing, is minimax, i.e., it dominates least-squares. We will now show that our EM procedure not only yields reasonable estimates in this setting, in contrast to LOOCV, but that these estimates are minimax, and hence dominate least-sqaures.

For the normal means model, we can obtain a closed form solution for the optimum τ, by solving for the stationary point for which $\tau_{t+1} = \tau_t$, with $\tau \sim C^+(0,1)$:

$$\begin{aligned}
\underset{\tau}{\arg\min}\left\{\mathbb{E}_{\boldsymbol{\beta}}\left[-\log\, p(y|\beta,\tau) - \log\, p(\beta|\tau) - \log\, \pi(\tau)\right]\right\} &= \tau \\
\underset{\tau}{\arg\min}\left\{\frac{p}{2}\log\tau^2 + \frac{w}{2\tau^2} + \log(1+\tau^2)\right\} &= \tau \\
\sqrt{\frac{w - p + \sqrt{p^2 + 8w + 2pw + w^2}}{2(2+p)}} &= \tau,
\end{aligned}$$

and with $w = \sum_{j=1}^p \mathbb{E}\left[\beta_j^2\right] = (1-\kappa)^2 s + (1-\kappa)p$, $s = ||y||^2$ and $\tau = \sqrt{\frac{1-\kappa}{\kappa}}$. This yields

$$\sqrt{\frac{\left(\sqrt{p((\kappa-2)^2 p - 8\kappa + 8) - 2(\kappa-1)^2 s((\kappa-2)p-4) + (\kappa-1)^4 s^2} - \kappa p + (\kappa-1)^2 s\right)}{2(2+p)}} = \sqrt{\frac{1-\kappa}{\kappa}}$$

with solution $\kappa = (p+2)/s$. Plugging this κ solution into (4.12), we note that the resulting estimator of $\boldsymbol{\beta}$ (4.12) is of the form (4.13) with

$$\begin{aligned}
r\left(\frac{1}{2}||y||^2\right) &= \left(\frac{p+2}{||y||^2}\right) / \left(\frac{p-2}{||y||^2}\right) \\
&= \frac{p+2}{p-2}
\end{aligned}$$

As we have $r\left(\frac{1}{2}||y||^2\right) \leq 2$ when $p \geq 6$, the EM ridge estimator is minimax in this setting for $p \geq 6$.

4.4.2 Derivation of the EM key quantities

In this subsection, we detail the step-by-step derivation of the key quantities required for the proposed EM algorithm as presented in Section 4.4. Specifically, we provide the derivation of the ESS and ESN quantities (4.9) for the E-step, as well as the update formulas (4.11) for $\hat{\sigma}^2$ and $\hat{\tau}^2$ for the M-step.

Derivation of ESN Here we show that

$$\begin{aligned}\sum_{j=1}^{p} \mathbb{E}\left[\beta_j^2 \mid \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}\right] &= \operatorname{tr}\left(\operatorname{Cov}[\boldsymbol{\beta}]\right) + \sum_{j=1}^{p} \mathbb{E}\left[\beta_j\right]^2 \\ &= \sigma^2 \operatorname{tr}(\mathbf{A}_\tau^{-1}) + \|\hat{\boldsymbol{\beta}}_\tau\|^2\end{aligned}$$

This is a rather straightforward proof. We use the fact that given a random variable x; the expected squared value of x is $\mathbb{E}\left[x^2\right] = \operatorname{Var}[x] + \mathbb{E}\left[x\right]^2$.

Derivation of ESS Here we show that

$$\mathbb{E}_{\boldsymbol{\beta}}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \mid \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}\right] = ||\mathbf{y} - \mathbf{X}\,\mathbb{E}\left[\boldsymbol{\beta}\right]||^2 + \operatorname{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\operatorname{Cov}[\boldsymbol{\beta}]) \tag{4.14}$$

We first present Lemma 4.2 on the quadratic forms of random variables in:

Lemma 4.2. *Let* $\mathbf{b}$ *be a p-dimensional random vector and* $\mathbf{A}$ *be a p-dimensional symmetric matrix. If* $\mathbb{E}\left[\mathbf{b}\right] = \boldsymbol{\mu}$ *and* $\operatorname{Var}(\mathbf{b}) = \boldsymbol{\Sigma}$*, then* $\mathbb{E}\left[\mathbf{b}^{\mathrm{T}}\mathbf{A}\mathbf{b}\right] = \operatorname{tr}(\mathbf{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu}$.

Now, we expand the left-hand side of Equation 4.14 :

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\beta}}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right] &= \mathbb{E}_{\boldsymbol{\beta}}\left[(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right] \\ &= \mathbb{E}_{\boldsymbol{\beta}}\left[\mathbf{y}^{\mathrm{T}}\mathbf{y} - 2\mathbf{y}^{\mathrm{T}}\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\boldsymbol{\beta}\right] \\ &= \mathbf{y}^{\mathrm{T}}\mathbf{y} - 2\,\mathbf{y}^{\mathrm{T}}\mathbf{X}\mathbb{E}\left[\boldsymbol{\beta}\right] + \mathbb{E}\left[\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\boldsymbol{\beta}\right]\end{aligned} \tag{4.15}$$

The use of lemma 4.2 allows Equation 4.15 to be rewritten as

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\beta}}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right] &= \mathbf{y}^{\mathrm{T}}\mathbf{y} - 2\,\mathbf{y}^{\mathrm{T}}\mathbf{X}\mathbb{E}\left[\boldsymbol{\beta}\right] + \mathbb{E}\left[\boldsymbol{\beta}\right]^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X})\mathbb{E}\left[\boldsymbol{\beta}\right] + \operatorname{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\operatorname{Cov}[\boldsymbol{\beta}]) \\ &= ||\mathbf{y} - \mathbf{X}\,\mathbb{E}\left[\boldsymbol{\beta}\right]||^2 + \operatorname{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\operatorname{Cov}[\boldsymbol{\beta}]) \\ &= ||\mathbf{y} - \mathbf{X}\,\hat{\boldsymbol{\beta}}_\tau||^2 + \sigma^2\operatorname{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{A}_\tau^{-1})\end{aligned}$$

Derivation of Equation 4.11 (Solving for the parameter updates) Rather than solving a two-dimensional numerical optimization problem (4.10), we show that given a fixed τ^2, we can find a closed formed solution for σ^2, and vice versa. To start off, we need to find the solution for σ^2 as a function of τ^2. First, find the negative logarithm of the joint probability distribution of hierarchy (4.5):

$$\underset{\sigma^2}{\arg\min} \left\{ \mathbb{E}_{\boldsymbol{\beta}} \left[-\log\ p(y|X,\beta,\sigma^2) - \log\ p(\beta|\tau^2,\sigma^2) - \log\ p(\sigma^2) - \log\ \pi(\tau^2) \right] \right\}. \tag{4.16}$$

Dropping terms that do not depend on σ^2 yields:

$$\begin{aligned} &\underset{\sigma^2}{\arg\min} \left\{ \mathbb{E}_{\boldsymbol{\beta}} \left[-\log\ p(y|X,\beta,\sigma^2) - \log\ p(\beta|\tau^2,\sigma^2) - \log\ p(\sigma^2) \right] \right\} \\ &= \underset{\sigma^2}{\arg\min} \left\{ \left(\frac{n+p}{2} \right) \log \sigma^2 + \frac{\text{ESS}}{2\sigma^2} + \frac{\text{ESN}}{2\sigma^2\tau^2} + \log \sigma^2 \right\} \\ &= \underset{\sigma^2}{\arg\min} \left\{ \left(\frac{n+p+2}{2} \right) \log \sigma^2 + \frac{\text{ESS}}{2\sigma^2} + \frac{\text{ESN}}{2\sigma^2\tau^2} \right\}. \end{aligned} \tag{4.17}$$

Solving the above minimization problem involves differentiating the negative logarithm with respect to σ^2 and solving for σ^2 that set the derivative to zero. This gives us:

$$\begin{aligned} \frac{\partial}{\partial \sigma^2} \left\{ \left(\frac{n+p+2}{2} \right) \log \sigma^2 + \frac{\text{ESS}}{2\sigma^2} + \frac{\text{ESN}}{2\sigma^2\tau^2} \right\} &= 0 \\ \frac{2+n+p}{2\sigma^2} - \frac{\text{ESS}}{2(\sigma^2)^2} - \frac{\text{ESN}}{2(\sigma^2)^2\tau^2} &= 0 \\ \hat{\sigma}^2 &= \frac{\tau^2\text{ESS} + \text{ESN}}{(n+p+2)\tau^2} \end{aligned} \tag{4.18}$$

Next, to obtain the M-step updates for the shrinkage parameter τ^2, we repeat the same procedure - find the negative logarithm of the joint probability distribution and remove terms that do not depend on either σ^2 or τ^2:

$$\begin{aligned} &\underset{\tau^2}{\arg\min} \left\{ \mathbb{E}_{\boldsymbol{\beta}} \left[-\log\ p(y|X,\beta,\sigma^2) - \log\ p(\beta|\tau^2,\sigma^2) - \log\ p(\sigma^2) - \log\ \pi(\tau^2) \right] \right\} \\ &= \underset{\tau^2}{\arg\min} \left\{ \left(\frac{n+p+2}{2} \right) \log \sigma^2 + \frac{\text{ESS}}{2\sigma^2} + \frac{\text{ESN}}{2\sigma^2\tau^2} + \frac{p}{2} \log \tau^2 + \log(1+\tau^2) + \frac{\log \tau^2}{2} \right\} \end{aligned} \tag{4.19}$$

Substiting the solution for σ^2 (4.18) into equation (4.19), yields a Q-function that depends only on τ^2. We eliminate the dependency on σ^2 by finding the optimal σ^2 as a function of τ^2 and substitute it into the Q-function of (4.19):

$$\underset{\tau^2}{\arg\min} \left\{ \frac{1}{2} \left[(1+p) \log \tau^2 + 2\log(1+\tau^2) + (n+p+2) \left(1 + \log \left(\frac{\text{ESN} + \tau^2\text{ESS}}{(n+p+2)\tau^2} \right) \right) \right] \right\} \tag{4.20}$$

Differentiating (4.20) with respect to τ^2 and solving for the τ^2 that set the derivative to zero yields:

$$\frac{\partial}{\partial \tau^2}\left\{\frac{1}{2}\left[(1+p)\log\tau^2+2\log(1+\tau^2)+(n+p+2)\left(1+\log\left(\frac{\mathrm{ESN}+\tau^2\mathrm{ESS}}{(n+p+2)\tau^2}\right)\right)\right]\right\}=0$$

$$\frac{(3\mathrm{ESS}+\mathrm{ESS}p)(\tau^2)^2+(\mathrm{ESN}-\mathrm{ESN}n+\mathrm{ESS}+\mathrm{ESS}p)\tau^2-\mathrm{ESN}-\mathrm{ESN}n}{2\tau^2(1+\tau^2)(\mathrm{ESN}+\tau^2\mathrm{ESS})}=0. \quad (4.21)$$

The τ^2 update is the positive solution to the quadratic equation (in terms of τ^2) (4.21):

$$\hat{\tau}^2=\frac{(n-1)\mathrm{ESN}-(1+p)\mathrm{ESS}+\sqrt{(4n+4)\mathrm{ESN}(3+p)\mathrm{ESS}+((1-n)\mathrm{ESN}+(p+1)\mathrm{ESS})^2}}{(6+2p)\mathrm{ESS}}$$

4.5 Fast Implementations via Singular Value Decomposition

To obtain efficient implementations of the E-Step of the EM algorithm as well as of the LOOCV shortcut formula, one can exploit the fact that the ridge solution is preserved under orthogonal transformations. Specifically, let $r=\min(n,p)$ and $m=\max(n,p)$ and let $\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}=\mathbf{X}$ be a compact singular value decomposition (SVD) of $\mathbf{X}$. That is, $\mathbf{U}\in\mathbb{R}^{n\times r}$ and $\mathbf{V}\in\mathbb{R}^{p\times r}$ are semi-orthonormal column matrices, i.e., $\mathbf{U}^{\mathrm{T}}\mathbf{U}=\mathbf{I}_n$ and $\mathbf{V}^{\mathrm{T}}\mathbf{V}=\mathbf{I}_p$, and $\boldsymbol{\Sigma}=\mathrm{diag}(s_1,\dots,s_r)\in\mathbb{R}^{r\times r}$ is a diagonal matrix that contains the non-zero singular values $\mathbf{s}=(s_1,\dots,s_r)$ of $\mathbf{X}$. With this decomposition, and an additional $O(nr)$ pre-processing step to compute $\mathbf{c}=\boldsymbol{\Sigma}\mathbf{U}^{\mathrm{T}}\mathbf{y}$, we can compute the ridge solution $\boldsymbol{\alpha}_\tau\in\mathbb{R}^r$ for a given τ with respect to the rotated inputs $\mathbf{XV}$ in time $O(r)$ via

$$\hat{\boldsymbol{\alpha}}_\tau=(\boldsymbol{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{U}\boldsymbol{\Sigma}+\tau^{-2}\mathbf{I})^{-1}\boldsymbol{\Sigma}\mathbf{U}^{\mathrm{T}}\mathbf{y}=(\boldsymbol{\Sigma}^2+\tau^{-2}\mathbf{I})^{-1}\mathbf{c}=\left(1/(s_j^2+\tau^{-2})\right)_{j=1}^{r}\odot\mathbf{c} \quad (4.22)$$

where $\mathbf{a}\odot\mathbf{b}$ denotes the element-wise Hadamard product of vectors $\mathbf{a}$ and $\mathbf{b}$. The compact SVD itself can be obtained in time $O(mr^2)$ via an eigendecomposition of either $\mathbf{X}^{\mathrm{T}}\mathbf{X}=\mathbf{V}\boldsymbol{\Sigma}^2\mathbf{V}^T$ in case $n\geq p$ or $\mathbf{X}\mathbf{X}^{\mathrm{T}}=\mathbf{U}\boldsymbol{\Sigma}^2\mathbf{U}^{\mathrm{T}}$ in case $n<p$ followed by the computation of the missing $\mathbf{U}=\mathbf{XV}\boldsymbol{\Sigma}^{-1}$ or $\mathbf{V}=\mathbf{X}^{\mathrm{T}}\mathbf{U}\boldsymbol{\Sigma}^{-1}$.

In summary, after an $O(mr^2)$ pre-processing step, we can obtain rotated ridge solutions for an individual candidate τ in time $O(r)$. Moreover, for the optimal τ^*, we can find the ridge solution $\hat{\boldsymbol{\beta}}_{\tau^*}=\mathbf{V}\hat{\boldsymbol{\alpha}}_{\tau^*}$ with respect to the original input matrix via an $O(pr)$ post-processing step. Below we show how the key statistics that have to be computed per candidate τ (and σ) can be computed efficiently based on $\hat{\boldsymbol{\alpha}}_\tau$, the pre-computed $\mathbf{c}$, and SVD. For the EM algorithm, these are the posterior squared norm and sum of squared errors, and for the LOOCV algorithm, this is the PRESS statistic. While the main focus of this work is the EM algorithm, the fast computation of the PRESS shortcut formula appears to be not widely known (e.g., the current implementation in both `scikit-learn` and `glmnet` do not use it) and may therefore be of independent interest.

ESN For the posterior expected squared norm ESN $= \sigma^2 \mathrm{tr}(\mathbf{A}_\tau^{-1}) + \|\hat{\boldsymbol{\beta}}_\tau\|^2$, we first observe that $\|\hat{\boldsymbol{\beta}}_\tau\|^2 = \|\mathbf{V}\hat{\boldsymbol{\alpha}}_\tau\|^2 = \|\hat{\boldsymbol{\alpha}}_\tau\|^2$, and then note that the trace can be computed as

$$\begin{aligned}\mathrm{tr}(\mathbf{A}_\tau^{-1}) &= \mathrm{tr}(\mathbf{V}_p(\boldsymbol{\Sigma}_p^2 + \tau^{-2}\mathbf{I}_p)^{-1}\mathbf{V}_p^{\mathrm{T}}) \\ &= \tau^2 \max(p-n,0) + \sum_{j=1}^{r} 1/(s_j^2 + \tau^{-2}),\end{aligned} \tag{4.23}$$

where in the first equation we denote by $\mathbf{V}_p$, $\boldsymbol{\Sigma}_p^2$ the full matrices of eigenvectors and eigenvalues of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ (including potential zeros), and in the second equation we used the cyclical property of the trace. Thus, all quantities required for ESN can be computed in time $O(r)$ given the SVD and $\hat{\boldsymbol{\alpha}}_\tau$.

ESS For the posterior expected sum of squared errors ESS $= \|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_\tau\|^2 + \sigma^2 \mathrm{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{A}_\tau^{-1})$, we can compute the residual sum of squares term via

$$\begin{aligned}\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_\tau\|^2 &= \|\mathbf{y}\|^2 - 2\mathbf{y}^T\mathbf{U}\boldsymbol{\Sigma}\hat{\boldsymbol{\alpha}}_\tau + \|\mathbf{U}\boldsymbol{\Sigma}\hat{\boldsymbol{\alpha}}_\tau\|^2 \\ &= \|\mathbf{y}\|^2 - 2\hat{\boldsymbol{\alpha}}_\tau^{\mathrm{T}}\mathbf{c} + \|\mathbf{s} \odot \hat{\boldsymbol{\alpha}}_\tau\|^2,\end{aligned} \tag{4.24}$$

where we use $\hat{\boldsymbol{\beta}}_\tau = \mathbf{V}\hat{\boldsymbol{\alpha}}_\tau$ and $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ in the first equation and the orthonormality of $\mathbf{U}$ and the definition of $\mathbf{c} = \boldsymbol{\Sigma}\mathbf{U}^{\mathrm{T}}\mathbf{y}$ in the second. Finally, for the trace term, we find that

$$\begin{aligned}\mathrm{tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{A}_\tau^{-1}) &= \mathrm{tr}(\mathbf{V}\boldsymbol{\Sigma}^2\mathbf{V}^{\mathrm{T}}(\mathbf{V}(\boldsymbol{\Sigma}^2 + \tau^{-2}\mathbf{I}_p)\mathbf{V}^{\mathrm{T}})^{-1}) \\ &= \sum_{j=1}^{r} s_j^2/(s_j^2 + \tau^{-2}).\end{aligned} \tag{4.25}$$

PRESS The shortcut formula of the PRESS statistic (4.4) for a candidate γ requires the computation of the diagonal elements of the hat matrix $\mathbf{H}(\gamma) = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \gamma\mathbf{I}_p)^{-1}\mathbf{X}^{\mathrm{T}}$ as well as the residual vector $\mathbf{e} = \mathbf{y} - \mathbf{H}\mathbf{y}$. With the SVD, the first simplifies to

$$\begin{aligned}\mathbf{H}(\gamma) &= \mathbf{U}\boldsymbol{\Sigma}(\boldsymbol{\Sigma}^2 + \gamma\mathbf{I}_r)^{-1}\boldsymbol{\Sigma}\mathbf{U}^{\mathrm{T}} \\ &= \mathbf{U}\,\mathrm{diag}\left(\frac{s_1^2}{s_1^2 + \gamma}, \dots, \frac{s_r^2}{s_r^2 + \gamma}\right)\mathbf{U}^{\mathrm{T}}\end{aligned}$$

where we use the fact that diagonal matrices commute. This allows to compute the desired diagonal elements h_{ii} in time $O(r)$ via

$$h_{ii}(\gamma) = \sum_{j=1}^{r} u_{ij}^2 s_j^2/(s_j^2 + \gamma) \tag{4.26}$$

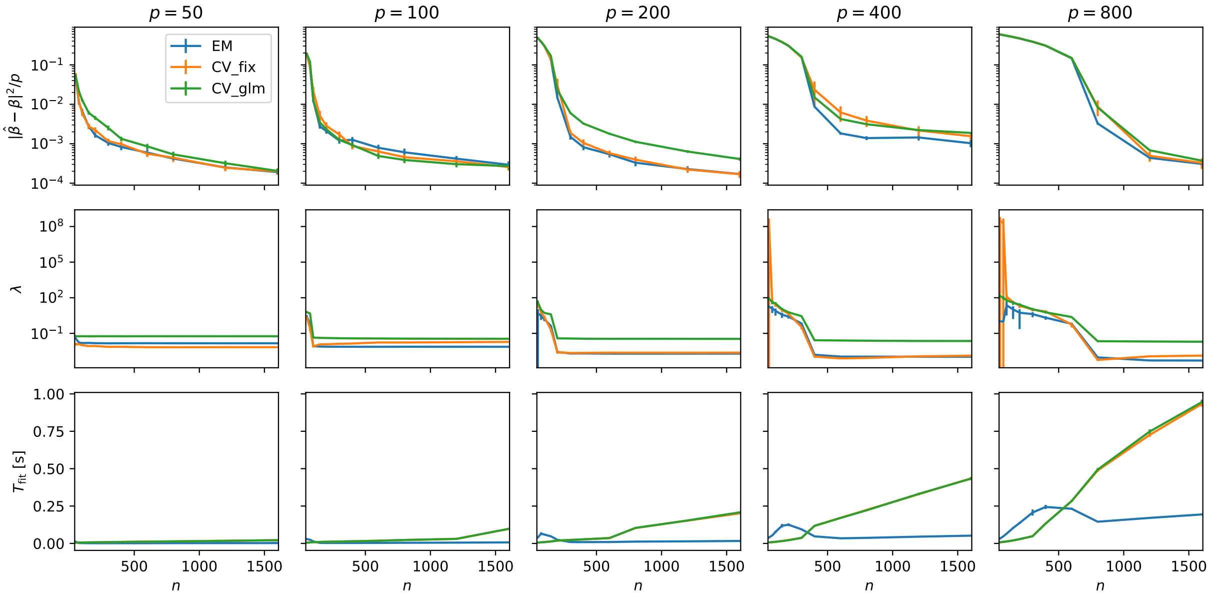


Figure 4.2: Comparison of EM to LOOCV variants for increasing n and p for settings with $\mathbf{x} \sim N_p(\mathbf{0}, \Sigma)$ and $\mathbf{y}|\mathbf{x} \sim N(\mathbf{x}^T\boldsymbol{\beta}, 0.25)$ with random $\boldsymbol{\beta} \sim N_p(\mathbf{0}, \mathbf{I})$ and $\Sigma \sim W_p(I, p)$.

where u_{ij} denotes the elements of $\mathbf{U}$. Computing the residual vector is easily done via the rotated ridge solution $\mathbf{e} = \mathbf{y} - \mathbf{U}\boldsymbol{\Sigma}\hat{\boldsymbol{\alpha}}_\gamma$. However, this still requires $O(nr)$ operations, simply because there are n residuals to compute.

Thus, in summary, by combining the pre-processing with the fast computation of the PRESS statistic, we obtain an overall $O(mr^2 + lqnr)$ implementation of ridge regression via LOOCV where l denotes the number of candidates γ and q the number of regression target variables. In contrast, for the EM algorithm, by combining the fast computation of ESS and ESN, we end up with an overall complexity of $O(mr^2 + kqr)$ where k denotes the number of EM iterations. If we further assume that $k = o(n)$, which is supported by experimental results, see Sec. 4.6, and that both $q, p = O(\sqrt{n})$ there is an asymptotic advantage of a factor of l of the EM approach. This regime is common in settings where more data allows for more fine-grained input as well as output measurements, e.g., in satellite time series classification via multiple target regression [52, 136]. All time complexities are summarized in Table 4.1 and detailed pseudocode for both the fast LOOCV algorithm and the fast EM algorithm is provided in Algorithm 2 and 3 respectively.

4.6 Empirical Evaluation

In this section, we compare the predictive performance and computational cost of LOOCV against the proposed EM method. We present numerical results on both synthetic and real-world datasets. To implement the LOOCV estimator, we use a predefined grid, $L = (\gamma_1, \ldots, \gamma_l)$. We use the two most common methods for this task: (i) fixed grid - arbitrarily selecting a very small value as

$\gamma_{\min}$, a large value as $\gamma_{\max}$, and construct a sequence of l values from $\gamma_{\max}$ to $\gamma_{\min}$ on log scale; (ii) data-driven grid - find the smallest value of $\gamma_{\max}$ that sets all the regression coefficient vector to zero [2] (i.e. $\hat{\boldsymbol{\beta}} = 0$), multiply this value by a ratio such that $\gamma_{\min} = \kappa\gamma_{\max}$ and create a sequence from $\gamma_{\max}$ to $\gamma_{\min}$ on log scale. The latter method is implemented in the `glmnet` package in combination with an adaptive κ coefficient

$$\kappa = \begin{cases} 0.0001 & , \text{ if } n \geq p \\ 0.01 & , \text{ otherwise} \end{cases} ,$$

which we replicate here as input to our fast LOOCV algorithm (Algorithm 2) to efficiently recover the `glmnet` LOOCV ridge estimate.[3]

We consider a fixed grid of $\boldsymbol{\gamma} = (10^{-10}, \ldots, 10^{10})$ and the grid based on the `glmnet` heuristic; in both cases, we use a sequence of length 100. The latter is a data-driven grid, so we will have a different penalty grid for each simulated or real data set. Our EM algorithm does not require a predefined penalty grid, but it needs a convergence threshold which we set to be $\epsilon = 10^{-8}$. All experiments in this section are performed in Python and the R statistical platform. Datasets and code for the experimental results is publicly available. As is standard in penalized regression, and without any loss of generality, we standardized the data before model fitting. This means that the predictors are standardized to have zero mean, standard deviation of one, and the target has a mean of zero, i.e., the intercept estimate is simply $\hat{\beta}_0 = (1/n) \sum y_i$.

4.6.1 Simulated Data

In this section, we use a simulation study to investigate the behavior of EM and LOOCV as a function of the sample size, n, and two other parameters of interest: the number of covariates p, and the noise level of the target variable. In particular, we are interested in the parameter estimation performance, the corresponding γ-values, and the computational cost. To gain further insights into the latter, the number of iterations performed by the EM algorithm is of particular interest, as we do not have quantitative bounds for its behavior. We consider two settings that vary in the level of sparsity and correlation structure of the covariates. The first setting (Fig. 4.1) assumes i.i.d Bernoulli distributed covariates with small success probabilities that result in sparse covariate matrices, while the second setting (Fig. 4.2) assumes normally distributed covariates with random non-zero covariances. In both cases, the target variable is conditionally normal with mean $\mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}$ for a random $\boldsymbol{\beta}$ drawn from a standard multivariate normal distribution.

[2] For ridge regression, $\gamma_{\max} = \infty$. Following the glmnet package, the sequence of γ is derived for $\alpha = 0.001$. The penalty function used by glmnet is $\gamma[(1-\alpha)\|\boldsymbol{\beta}\|_2^2 + \alpha\|\boldsymbol{\beta}\|_1]$, where $\alpha = 0$ corresponds to ridge regression.

[3] `glmnet` LOOCV is computed directly by model refitting via coordinate-wise descent which can be slow.

Looking at the results, a common feature of both settings is that the computational complexity of the EM algorithm is a non-monotone function in n. In contrast to LOOCV, the behavior of EM shows distinctive phases where the complexity temporarily decreases with n before it settles into the, usually expected, monotonically increasing phase. As can be seen, this is due to the behavior of the number of iterations k, which peaks for small values of n before it declines rapidly to a small constant (around 10) when the cost of the pre-processing begins to dominate. The occurrence of these phases is more pronounced for both growing p and growing σ. This behavior is likely due to the convergence to normality of the posterior distribution as the sample size $n \to \infty$, with convergence being slower for large p.

An interesting observation is that CV with the employed `glmnet` grid heuristic fails, in the sense that the resulting ridge estimator does not appear to be consistent for large p in Setting 2. This is due to the minimum value of γ produced by the `glmnet` heuristic being too large, and the resulting ridge estimates being overshrunk. This clearly underlines the difficulty of choosing a robust dynamic grid – a problem that our EM algorithm avoids completely.

4.6.2 Real Data

We evaluated our EM method on 24 real-world datasets. This includes 21 datasets from the UCI machine learning repository [16] (unless referenced otherwise) for normal linear regression tasks and 3 time-series datasets from the UCR repository [51] for multitarget regression tasks. The latter is a multilabel classification problem in which the feature matrix was generated by the state-of-the-art HYDRA [53] time series classification procedure (which by default uses LOOCV ridge regression for classification), and we train q ridge regression models in a one-versus-all fashion, where q is the number of target classes. The datasets were chosen such that they covered a wide range of sample sizes, n, and number of predictors, p. We compared our EM algorithm against the fast LOOCV in terms of predictive performance, measured in R^2 (and classification accuracy) on the test data, and computational efficiency.

Our linear regression experiments involve 3 settings: (i) standard linear regression; (ii) second-order multivariate polynomial regression with added interactions and second-degree polynomial transformations of variables, and (iii) third-order multivariate polynomial regression with added three-way interactions and cubic polynomial transformations. For each experiment, we repeated the process 100 times and used a random 70/30 train-test split. Due to memory limitations, we limit our design matrix size to a maximum of 35 million entries. If the number of transformed predictors exceeded this limit, we uniformly sub-sampled the interaction variables to ensure that $p^* \leq 35000000/(0.7n)$, and then fit the model using the sampled variables. Note that we always keep the original variables (main effects) and sub-sampled the interactions. In the case of multitarget regression, we performed a random 70/30 train-test split and repeated the experiment

Table 4.2: Real datasets experiment results. The first column is the dataset (abbreviated, refer to Appendix A for the full name); the number of target variables q, the training sample size n, the number of features p; T is the ratio of time t_{CV}/t_{EM} ; p^* is the number of features including interactions; EM, Fix, and GLM are the R^2 values on the test data for the three procedures.

			Linear				2nd Order					3rd Order				
Dataset (q)	n	p	T	EM	Fix	GLM	p^*	T	EM	Fix	GLM	p^*	T	EM	Fix	GLM
Twitter	408275	77	20	0.94	0.94	0.94	86	16	0.94	0.94	0.94	-	-	-	-	-
Blog	39355	275	13	0.46	0.46	0.46	804	9.1	0.51	0.51	0.51	-	-	-	-	-
CT slices	37450	379	12	0.86	0.86	0.86	930	7.7	0.92	0.91	0.92	-	-	-	-	-
TomsHw	19725	96	17	0.63	0.63	0.63	1775	6.5	0.71	0.71	0.71	-	-	-	-	-
NPD - com	8353	13	13	0.84	0.84	0.84	104	15	1.00	1.00	1.00	559	8.6	1.00	1.00	1.00
NPD - tur	8353	13	14	0.91	0.91	0.91	104	15	1.00	1.00	1.00	559	8.6	1.00	1.00	1.00
PT - motor	4112	19	13	0.15	0.15	0.15	208	12	0.25	0.19	0.21	1539	3.9	-1.09	0.01	0.04
PT - total	4112	19	13	0.17	0.17	0.17	208	12	0.24	0.23	0.21	1539	3.7	-1.38	-0.04	0.00
Abalone	2923	9	13	0.53	0.53	0.53	51	16	0.38	0.35	0.50	209	12	0.28	0.12	0.12
Crime	1395	99	14	0.66	0.66	0.66	5049	1.3	0.67	-0.74	0.66	17652	1.1	0.66	-0.22	0.60
Airfoil	1052	5	17	0.51	0.51	0.51	20	14	0.62	0.62	0.62	55	11	0.73	0.73	0.73
Student	730	39	12	0.18	0.18	0.18	801	3.8	0.19	-0.89	0.16	10693	1.1	0.19	-6.22	-0.18
Concrete	721	8	16	0.61	0.61	0.61	44	11	0.78	0.78	0.78	164	5.5	0.85	0.85	0.85
F.Fires	361	12	2.4	-0.01	-0.01	-0.03	295	0.5	-0.01	-0.01	-0.14	1984	0.3	-0.01	-50.6	-0.45
B.Housing	354	13	11	0.71	0.71	0.71	104	8.1	0.84	0.83	0.80	559	2.2	0.83	-3e2	0.83
Facebook	346	15	15	0.91	0.91	0.91	167	6.6	-5.09	-26.4	-3.99	1087	1.9	-2.53	-2e4	-5.76
Diabetes [1]	309	10	13	0.49	0.49	0.49	65	6.5	0.49	0.48	0.48	285	2.1	0.47	0.47	0.47
R.Estate	289	6	13	0.56	0.56	0.56	27	10	0.65	0.65	0.65	83	5.5	0.65	0.65	0.65
A.MPG	278	8	13	0.81	0.81	0.81	35	8.3	0.86	0.86	0.86	119	4.1	0.86	0.86	0.86
Yacht	215	7	16	0.97	0.97	0.97	27	11	0.97	0.97	0.97	83	6.1	0.98	0.98	0.98
A.mobile	111	25	6.1	0.90	0.89	0.89	1076	1.7	0.90	-4e3	0.89	12924	0.5	0.88	-1e4	0.83
Eye [1]	84	200	2.7	0.50	0.26	0.45	20300	1.3	0.19	0.19	0.19	-	-	-	-	-
Ribo [1]	49	4088	2.3	0.64	0.64	0.64	-	-	-	-	-	-	-	-	-	-
Crop (24) [2]	11760	3072	49	0.75	0.75	0.76	-	-	-	-	-	-	-	-	-	-
ElecD (7) [2]	11645	4096	9.2	0.88	0.88	0.89	-	-	-	-	-	-	-	-	-	-
StarL (3) [2]	6465	7168	2.1	0.98	0.60	0.98	-	-	-	-	-	-	-	-	-	-

[1] This dataset is not from the UCI repository. Data references can be found in the appendix.
[2] Time-series dataset from the UCR repository. EM, Fix, and GLM are the classification accuracy on the test data

30 times. To ensure efficient reproducibility of our experiments, we set a maximum runtime of 3 hours for each dataset. Any settings that exceeded this time limit were consequently excluded from the result table.

Table 4.2 details the results of our experiments; specifically, the ratio of time taken to run fast LOOCV divided by the time taken to run our EM procedure (T), and the R^2 values obtained by both methods on the withheld test set. The number of features, p, and observations, n recorded are values after data preprocessing (missing observations removed, one-hot encoding transformation, etc.). The results demonstrate that our EM algorithm can be up to 49 times faster than the fast LOOCV, with the speed-ups becoming more apparent as the sample size n and the number of target variables q increases. In addition, we see that this advantage in speed does not come at a cost in predictive performance, as our EM approach is comparable to, if not better than, LOOCV in almost all cases (also see Figure 4.3, in which most of R^2 values are distributed along the diagonal line).

An interesting observation is that LOOCV using the fixed grid can occasionally perform extremely poorly (as indicated by large negative R^2 values) while LOOCV using the `glmnet` grid

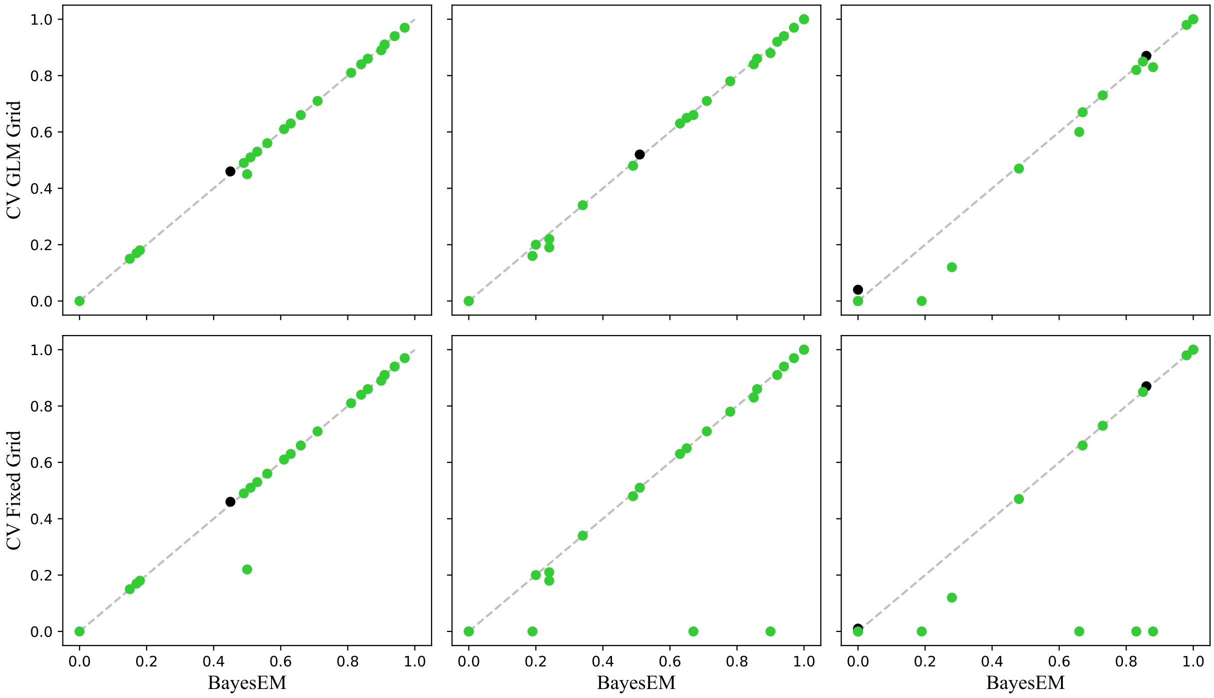


Figure 4.3: Comparison of predictive performance (R^2) of EM algorithm (x-axes) against CV with fixed grid (y-axes, top) and `glmnet` heuristic (y-axis, bottom). Columns correspond to the results of linear features (left), second-order features (middle), and third-order features (right). Negative values are capped at 0. Points skewing toward the bottom right indicate when our EM approach is giving better/same prediction performance as LOOCV (colored in green).

does not seem to exhibit this behavior. This appears likely to be due to the grid chosen using the `glmnet` heuristic. Its performance is artificially improved because it is unable to evaluate sufficiently small values of γ and is not actually selecting the very small γ value that minimizes the LOOCV score. The incorrectly large γ values are providing protection in these examples from undershrinkage.

4.7 Conclusion and Future Work

The introduced EM algorithm is a robust and computationally fast alternative to LOOCV for ridge regression. The unimodality of the posterior guarantees a robust behavior for finite n under mild conditions relative to LOOCV grid search, and the SVD preprocessing enables an overall faster computation with an ultra-fast $O(k \min(n, p))$ main loop. Combining this with a procedure such as orthogonal least-squares to provide a highly efficient forward selection procedure is a promising avenue for future research. As the Q-function is an expected negative log-posterior, it offers a score on which the usefulness of predictors themselves may be assessed, i.e., a model selection criteria, resulting in a potentially very accurate and fast procedure for sparse model learning.

An important open problem is the theoretical analysis of the expected number of EM iterations k that is required for convergence. The empirical evidence suggests that k converges to a constant, and is thus negligible in the asymptotic time complexity. This is in alignment with the convergence of the posterior to a multivariate normal distribution. However, such intuitive and empirical arguments cannot replace a rigorous worst-case analysis.

Algorithm 2: Fast LOOCV ridge with SVD

```
Input  : Standardised predictors $\mathbf{X} \in \mathbb{R}^{n \times p}$, centered targets $\mathbf{y} \in \mathbb{R}^n$ and a grid of
         penalty parameters $L = (\gamma_1, \gamma_2, \ldots, \gamma_l)$
Output : $\boldsymbol{\beta} \in \mathbb{R}^p$

$r = \min(n, p)$                                                                    $O(1)$
if $p \geq n$ then
    $[\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}] = \mathrm{svd}(\mathbf{X})$                                  $O(mr^2)$
    $\mathbf{s} = (\Sigma_{1,1}, \ldots, \Sigma_{r,r})$                                               $O(r)$
    $\mathbf{R} = (s_1\mathbf{u}_1, \ldots, s_r\mathbf{u}_r)$      // column vectors $\mathbf{r}_j$ for $1 \leq j \leq r$.      $O(nr)$
    $\mathbf{c} = (\mathbf{U}^{\mathrm{T}}\mathbf{y}) \odot \mathbf{s}$                                               $O(nr)$
else
    $[\mathbf{V}, \boldsymbol{\Sigma}^2] = \mathrm{eigen}(\mathbf{X}^{\mathrm{T}}\mathbf{X})$                             $O(mr^2)$
    $\mathbf{s}^2 = (\boldsymbol{\Sigma}^2_{1,1}, \ldots, \boldsymbol{\Sigma}^2_{r,r})$                                        $O(r)$
    $\mathbf{R} = \mathbf{X}\mathbf{V}$                                                          $O(nrp)$
    $\mathbf{c} = \mathbf{R}^{\mathrm{T}}\mathbf{y}$                                                          $O(nr)$
    $\mathbf{U} = (\mathbf{r}_1/s_1, \ldots, \mathbf{r}_r/s_r)$                                       $O(nr)$
end if

for $\gamma \in L$ do
    $h_i = \sum_{j=1}^{r} \left( \frac{s_j^2}{s_j^2 + \gamma} \right) u_{ij}^2, \qquad (i = 1, \ldots, n)$                 $O(lnr)$
    $\alpha_j = \frac{c_j}{s_j^2 + \gamma}$                                                      $O(lr)$
    $\mathbf{e} = \mathbf{y} - \mathbf{R}\boldsymbol{\alpha}$                                                   $O(lnr)$
    $\mathrm{CVE}(\gamma) = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{e_i}{1 - h_i} \right)^2$                            $O(ln)$
end for

$\gamma^* = \arg\min_{\gamma \in L} \{\mathrm{CVE}(\gamma)\}$                                            $O(l)$
$\alpha_j = \frac{c_j}{s_j^2 + \gamma^*}$                                                         $O(r)$
$\boldsymbol{\beta} = \mathbf{V}\boldsymbol{\alpha}$                                                             $O(pr)$
return $\boldsymbol{\beta}$
```

Algorithm 3: EM Algorithm with SVD

```
Input  : Standardised predictors $\mathbf{X} \in \mathbb{R}^{n \times p}$, centered targets $\mathbf{y} \in \mathbb{R}^n$ and convergence
         threshold $\epsilon > 0$
Output : $\boldsymbol{\beta} \in \mathbb{R}^p$

$r = \min(n, p)$                                                    $O(1)$
if $p \geq n$ then
    $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \mathrm{svd}(\mathbf{X})$                              $O(mr^2)$
    $\mathbf{s}^2 = (\Sigma_{1,1}^2, \ldots, \Sigma_{r,r}^2)$                                   $O(r)$
    $\mathbf{c} = (\mathbf{U}^{\mathrm{T}}\mathbf{y}) \odot \mathbf{s}$                                         $O(nr)$
else
    $[\mathbf{V}, \mathbf{\Sigma}^2] = \mathrm{eigen}(\mathbf{X}^{\mathrm{T}}\mathbf{X})$                          $O(mr^2)$
    $\mathbf{s}^2 = (\mathbf{\Sigma}_{1,1}^2, \ldots, \mathbf{\Sigma}_{r,r}^2)$                                   $O(r)$
    $\mathbf{c} = \mathbf{V}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{y}$                                            $O(np)$
end if
$Y = \mathbf{y}^{\mathrm{T}}\mathbf{y}$                                                       $O(n)$
$\tau^2 \leftarrow 1$                                                        $O(1)$
$\sigma^2 \leftarrow (1/n)\sum_{i=1}^{n}(y_i - \bar{y})^2$ , $\bar{y} = (1/n)\sum_{i=1}^{n} y_i$                     $O(n)$
$\mathrm{RSS} \leftarrow \infty$                                                    $O(1)$

while $\delta < \epsilon$ do
    $\mathrm{RSS}_{\mathrm{old}} \leftarrow \mathrm{RSS}$                                           $O(k)$
    $\alpha_j \leftarrow \dfrac{c_j}{s_j^2 + 1/\tau^2}$                                         $O(kr)$
    // (E-step)
    $\mathrm{ESN} \leftarrow \sum_{j=1}^{r}\alpha_j^2 + \sigma^2\left(\sum_{j=1}^{r}\dfrac{1}{s_j^2 + \tau^{-2}} + \tau^2\max(p - n, 0)\right)$         $O(kr)$
    $\mathrm{RSS} \leftarrow Y - 2\sum_{j=1}^{r}\alpha_j c_j + \sum_{j=1}^{r}\alpha_j^2 s_j^2$                    $O(kr)$
    $\mathrm{ESS} \leftarrow \mathrm{RSS} + \sigma^2\left(\sum_{j=1}^{r}\dfrac{s_j^2}{s_j^2 + \tau^{-2}}\right)$                  $O(kr)$
    // (M-step)
    $g \leftarrow (4n+4)\mathrm{ESN}\,(3+p)\mathrm{ESS} + ((1-n)\mathrm{ESN} + (p+1)\mathrm{ESS})^2$             $O(k)$
    $\tau^2 \leftarrow \dfrac{(n-1)\mathrm{ESN} - (1+p)\mathrm{ESS} + \sqrt{g}}{(6+2p)\mathrm{ESS}}$                 $O(k)$
    $\sigma^2 \leftarrow \dfrac{\tau^2\mathrm{ESS} + \mathrm{ESN}}{(n+p+2)\tau^2}$                             $O(k)$
    $\delta \leftarrow \dfrac{|\mathrm{RSS}_{\mathrm{old}} - \mathrm{RSS}|}{(1 + |\mathrm{RSS}|)}$                            $O(k)$
end while

$\alpha_j = \dfrac{c_j}{s_j^2 + 1/\tau^2}$                                               $O(r)$
$\boldsymbol{\beta} = \mathbf{V}\boldsymbol{\alpha}$                                                      $O(pr)$
return $\boldsymbol{\beta}$
```

Chapter 5

Sparse Estimation via Expectation-Maximisation

Lasso and horseshoe regularization are highly regarded estimators in the class of global-local shrinkage priors, known for their effectiveness in sparse estimation. Both methods offer strong theoretical guarantees for estimation, prediction, and variable selection [27]. While the literature on the Lasso has a long history with various adaptations aimed at improving its performance (see Chapter 2 or Tibshirani [166] for a more comprehensive review of existing Lasso regularization methods), the theoretical foundations of horseshoe regularization, although promising, remain an actively evolving research area. One key difference between the two sparse estimators is that Lasso employs convex penalization, whereas horseshoe regularization is non-convex. Generally, the horseshoe prior tends to outperform the Laplace prior-based estimator, which is the Bayesian equivalent of Lasso [27]. This superiority can be attributed to the horseshoe's adaptability to sparsity and robustness against large signals, thanks to its heavy tails and probability spike at zero. In contrast, Lasso, due to its convex nature, is known to suffer from over-shrinkage and consistency issues [43]. Non-convex penalties can achieve optimal theoretical performance for sparse estimation without suffering from the same issue. However, they come with a higher computational challenge.

In this chapter, we tackle the challenge of sparse parameter estimation under the horseshoe prior. The horseshoe prior is known to possess many desirable properties for Bayesian estimation of sparse parameter vectors, but it lacks a closed-form density function, making it difficult to find a closed-form solution for the posterior mode. Conventional horseshoe estimators use the posterior mean for parameter estimation, but these estimates are not sparse. We extend the EM algorithm introduced in Chapter 3 to sparse Bayesian regression models, with a specific focus on Bayesian Horseshoe and Bayesian Lasso models. The necessary implementation details for these models are summarized in Table 5.1. This extension leads to a novel sparse horseshoe

Table 5.1: Negative log-prior distribution and EM updates for the proposed Bayesian horseshoe and lasso estimator, where $\mathrm{ESS} = \mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right]$ is the (conditionally) expected sum of squared errors, $\mathrm{ESN} = \mathbb{E}\left[||\boldsymbol{\beta}||^2\right]$ the expected squared norm and $W_j = \mathrm{ESN}_j/(2\sigma^2\tau^2)$. For completeness, we provide the EM updates for the Bayesian ridge estimator discussed in Chapter 4.

Prior	$-\log \pi(\boldsymbol{\lambda}, \tau)$	M-step
Horseshoe	$\sum_{j=1}^{p}\left[\log(1+\lambda_j^2) + \frac{\log \lambda_j^2}{2}\right] + \log(1+\tau^2)$	$\hat{\lambda}_j^{2\,(t+1)} = \frac{\left(\sqrt{1+6W_j+W_j^2}\right) + W_j - 1}{4}$
Lasso	$\sum_{j=1}^{p}\left[\frac{\lambda_j^2}{2}\right] + 2\log\tau^2 + \frac{1}{\tau^2}$	$\hat{\lambda}_j^{2\,(t+1)} = \frac{1}{2}(\sqrt{1+8W_j} - 1)$
Ridge	$\frac{\log \tau^2}{2} + \log(1+\tau^2)$	$\hat{\tau}^2 = \frac{(n-1)\mathrm{ESN} - (1+p)\mathrm{ESS} + \sqrt{g}}{(6+2p)\mathrm{ESS}}$

estimator, representing a pioneering effort to explore posterior mode estimation under the horseshoe prior. Our results demonstrate that our EM algorithm-driven HS posterior mode estimates offer statistically comparable or superior performance while maintaining computational efficiency when compared to state-of-the-art non-convex solvers such as the smoothly clipped absolute deviation (SCAD) [61] and the minimax concave penalty (MCP) [190]. We also extend the application of the EM algorithm to the Bayesian Lasso, showing that the EM Lasso posterior mode estimate converges to the exact Bayesian Lasso posterior mode estimate for large coefficients. Furthermore, we conduct a detailed analysis of how reparameterizing the local shrinkage parameter influences the degree of shrinkage applied to the posterior mode estimates, both in the context of Bayesian Horseshoe and Bayesian Lasso. In Section 5.4, we extend the scope of the EM algorithm to address regression problems with grouping structures.

5.1 Sparse Horseshoe Estimator

Carvalho et al. [42] introduced the use of the horseshoe prior in sparse regression and demonstrated its robustness at handling sparsity with large signals. This is achieved by placing a half-Cauchy prior distribution over both the local and global shrinkage parameters. It was subsequently suggested that the horseshoe prior can be generalized by assigning an inverted-beta prior on $\lambda_j^2 \sim \mathrm{B}^{'}(a, b)$ with probability density function [141]

$$p(\lambda_j^2) = \frac{(\lambda_j^2)^{a-1}(1+\lambda_j^2)^{-a-b}}{B(a,b)} \tag{5.1}$$

where $B(a, b)$ is the beta function, and $a > 0$, $b > 0$. We refer readers to [155] for further details about the effect of these hyperparameters a and b on the origin and tail properties of the generalized horseshoe prior. Our interest is in the particular case that $a = b = 1/2$ as this corresponds to a usual half Cauchy prior for λ_j.

Theoretical analyses of Bayes estimators are often conducted within the framework of the normal means model. This model can be viewed as a special case of the linear regression model with $p = n$ and $\mathbf{X} = I_{n\times n}$ where $I_{n\times n}$ is an identity matrix of order n with the Gram matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ having an orthogonal design. This implies that we observe data $\mathbf{y}$ from the following probability model:

$$\begin{aligned} y_i|\beta_i &\sim N(\beta_i, \sigma^2) \\ \beta_i|\lambda_i, \tau &\sim N(0, \lambda_i^2\tau^2) \\ \lambda_i &\sim \mathrm{C}^+(0,1). \end{aligned} \tag{5.2}$$

In this model, $\boldsymbol{\beta}$ is believed to be sparse, and the objective is to accurately identify and estimate the non-zero components of the unknown regression coefficients $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_n) \in \mathbb{R}^n$. The horseshoe estimator under this model assumes that each local shrinkage parameter λ is conditionally independent with half-Cauchy density where $C^+(0,1)$ denotes a standard half-Cauchy distribution with a probability density function of

$$p(z) = \frac{2}{\pi(1+z^2)},\ z > 0. \tag{5.3}$$

The posterior distribution of β under model (5.2) is normal, with the following mean and variance:

$$\begin{aligned} \mathbb{E}\left[\beta_i \mid y_i, \lambda_i, \tau, \sigma^2\right] &= \left(\frac{\lambda_i^2}{1+\lambda_i^2}\right) y_i + \left(\frac{1}{1+\lambda_i^2}\right) 0 = (1-\kappa_i)y_i, \\ \mathbb{V}\left[\beta_i \mid y_i, \lambda_i, \tau, \sigma^2\right] &= (1-\kappa_i)\sigma^2, \end{aligned} \tag{5.4}$$

where $\kappa_i = 1/(1+\lambda_i^2\tau^2)$ is a shrinkage coefficient which can be interpreted as the amount of shrinkage towards zero, a posterior. Although the horseshoe prior $\pi(\boldsymbol{\beta}|\tau)$ itself lacks an analytic form, Carvalho et al. [42] provided a closed-form expression of the marginal posterior mean under the horseshoe, assuming $\sigma = 1$:

$$\mathbb{E}\left[\beta_i \mid y_i, \tau\right] = y_i \left\{1 - \frac{2\Phi_1(1/2, 1, 3/2, y_i^2/2, 1-1/\tau^2)}{3\Phi_1(1/2, 1, 5/2, y_i^2/2, 1-1/\tau^2)}\right\}.$$

where $\Phi_1(\alpha, \beta, \gamma, x, y)$ is the confluent hypergeometric function of two variables [74, Sec. 9.261]. Numerous studies in the literature have explored the optimality of this posterior mean estimate, with Van Der Pas et al. [171] demonstrating its minimax optimality in estimation under ℓ_2 loss and Datta and Ghosh [50] showing its asymptotic optimality in testing under 0–1 loss. However, little is known about the properties of the posterior mode under the horseshoe prior.

A work closely related to our own in this section is the study by Bhadra et al. [28]. They introduced an approximation to the horseshoe prior, referred to as the "horseshoe-like" (HS-like) prior. This approximation has a closed-form density function and a scale-mixture representation which was used in conjunction with the conventional EM algorithm to estimate the posterior mode. Their research revealed that the posterior mode estimate under the exact horseshoe prior is nearly unbiased, and sparse, but violates the continuity property of Fan and Li [61] (as described in Chapter 2.1), similar to hard thresholding. They found that using the posterior mean as an estimator resolves the continuity issue and improves squared error loss but it does not provide a sparse solution, making it unsuitable for variable selection. Our work focuses on exploring the posterior mode estimates under the exact horseshoe prior and comparing it to their findings.

5.1.1 Horseshoe MAP Estimation via EM

We now demonstrate the application of the EM procedure described in Chapter 3.3 to the horseshoe prior and provide the exact EM updates for each of the shrinkage parameters. Here, we assume no prior knowledge on the sparsity of the regression coefficient and assign the recommended default prior for the global variance parameter [66, 141]:

$$\tau \;\sim\; C^{+}(0,1), \;\; \tau \in (0,1) \tag{5.5}$$

We limit the range of τ to $(0,1)$ following [170]. The degree of sparsity applied to individual regression coefficients depends on the choice of the prior distribution assigned to the local variance (shrinkage) component. In this section, we consider the horseshoe prior in its inverted beta prime form with parameters $a = b = 1/2$, denoted as $\lambda_j^2 \sim \mathrm{B}'(1/2, 1/2)$. The probability density function for this prior is defined in (5.1). Substituting the negative logarithm of this prior distribution into (3.4) – (3.5) and holding τ^2 and σ^2 fixed yields the M-step update for each λ_j^2:

$$\begin{aligned}\hat{\lambda}_j^2 &= \underset{\lambda_j^2}{\operatorname{argmin}} \left\{ \log \lambda_j^2 + \frac{W_j}{\lambda_j^2} + \log(1+\lambda_j^2) \right\} \\ &= \frac{1}{4}\left(\sqrt{1 + 6W_j + W_j^2} + W_j - 1\right) \end{aligned} \tag{5.6}$$

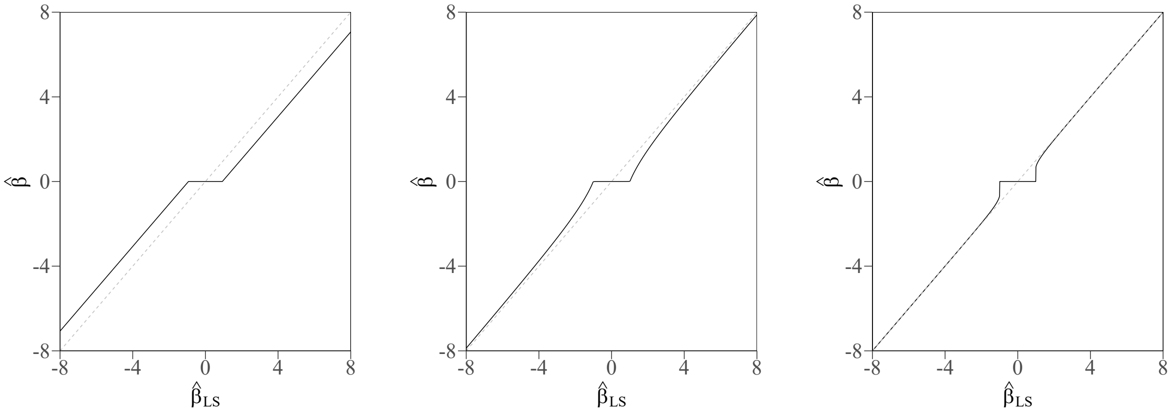
$$(\hat{\tau}^2, \hat{\sigma}^2) = \underset{\tau^2,\sigma^2}{\operatorname{argmin}}\, Q(\tau^2, \sigma^2; \hat{\boldsymbol{\lambda}}^2) \tag{5.7}$$

where $W_j = E[\beta_j^2]/(2\sigma^2\tau^2)$. For convenience, the negative log-prior and the M-step update for the horseshoe prior are summarised in Table 5.1. Given the $\hat{\boldsymbol{\lambda}}^2$ updates (5.6), the $\hat{\tau}$ and $\hat{\sigma}^2$ estimates can be found using numerical optimisation. This two-dimensional optimization problem (5.7) can be further reduced to the following one-dimensional optimization problem by (approximately) estimating σ^2 using the expected residual sum-of-squares given in Equation (3.12) or (3.13):

$$\begin{aligned} \hat{\sigma}^2 &= \mathbb{E}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2\right]/n, \\ \hat{\tau}^2 &= \underset{\tau^2}{\operatorname{argmin}}\, Q(\tau^2; \hat{\boldsymbol{\lambda}}^2, \hat{\sigma}^2). \end{aligned} \tag{5.8}$$

This approximate update for σ^2 is within $O(n^{-1})$ of the exact solution and allows us to substantially reduce the complexity of the optimization problem.

Figure 5.1 compares the shrinkage profiles for the Lasso, horseshoe, and horseshoe-like estimator. The plots demonstrate how the three estimators shrink the least-square (unpenalized) estimate $\hat{\beta}_{\text{LS}}$ towards zero. All three procedures shrink small least-squares estimates to zero. The lasso affects all values of $\hat{\beta}_{\text{LS}}$ by translating them towards zero by the same amount, while the horseshoe and horseshoe-like leave large values of $\hat{\beta}_{\text{LS}}$ largely unshrunk. The horseshoe-like prior exerts more of a hard thresholding-like behavior as $\hat{\beta}_{\text{LS}}$ approaches zero, while the horseshoe prior mimics firm thresholding.



(A) Lasso

(B) Horseshoe

(C) Horseshoe-like

Figure 5.1: The posterior mode estimates $\hat{\beta}$ versus $\hat{\beta}_{\text{LS}}$ for the (a) Lasso, (b) Horseshoe, and (c) Horseshoe-like estimator. For illustration purposes, τ is chosen such that all three estimators give nearly identical shrinkage within approximately 1 unit of the origin.

5.1.2 Experimental Results

We compare the performance of our proposed EM-based horseshoe estimator (HS-EM, using the exact expectations, and HS-apx, using the approximate expectations) against several state-of-the-art sparse methods including non-convex estimators (SCAD, MCP, HS-like estimator [28]), lasso and ridge estimator. We analyze the performance of the estimators in terms of variables selected and prediction accuracy on both simulated and real data. The experiments in this section are designed for variable selection under high dimensional settings, and we use the following metrics:

- **MSE**. Mean squared prediction error; given by $(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\boldsymbol{\Sigma}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})$ for simulated data experiments, and $(1/n)||\hat{\mathbf{y}} - \mathbf{y}||_2^2$ for real data experiments.
- **Time**. Computation time (in seconds).
- **No.V**. Number of variables included in the model.
- **TNZ**. True non-zeros. Number of non-zero coefficients correctly identified.
- **FNZ**. False non-zeros. Number of zero coefficients incorrectly identified as non-zero.

All experiments are performed in the R statistical platform. Datasets and code for the experimental results in this section are publicly available[1]. The proposed EM horseshoe estimator is also implemented in the **bayesreg**[2] Bayesian regression package. We use the `glmnet` package to implement the lasso and ridge, while the `ncvreg` package implements both MCP and SCAD. The hyperparameters of the non-Bayesian techniques are tuned using 10-fold cross-validation. All function arguments are set to the default value unless mentioned otherwise. Our proposed estimator terminates when it satisfies the convergence criterion: $\sum_{j=1}^{p}(|\beta_j^{(t)} - \beta_j^{(t+1)}|)/(1+\sum_{j=1}^{p}(|\beta_j^{(t+1)}|)) < \omega$, where $\beta^{(t)}$ denotes the coefficient estimates at iteration t and ω is the tolerance parameter which we set it to 10^{-5}.

5.1.2.1 Simulated Data

This section compares the prediction and variable selection performance of various sparse estimators with the normal means model and linear regression model across various simulation settings. The experimental setups are based on the simulation study presented in [28].

Normal Means Model Here, $n = 1000$ with the true sparse mean vector, $\boldsymbol{\beta} = \{\boldsymbol{b}_{10}, -\boldsymbol{b}_{10}, \mathbf{0}_{980}\}$ where $b = \{3, 10\}$, $\boldsymbol{b}_{10}$ is a vector of length 10 with entries equal to b, while $\mathbf{0}_{980}$ is a vector of length 980 with entries equal to 0. The data is generated as $(y_i|\beta_i) \sim N(\beta_i, 1)$, for which $\mathbf{y}$ is generated from a normal distribution with mean $\boldsymbol{\beta}$ and a variance of 1. We repeat this simulation 100 times and the results are summarised in Table 5.2. It should be noted that the implementations described in Section 3 are built upon a regression model. This normal means model can be seen as a special case of the regression model with $p = n$ and $X = I_{n\times n}$ where $I_{n\times n}$ is an identity matrix of order n. But naively using the methods described in Chapter 3 to solve this normal means problem is not ideal and slow, because we will have to work with a matrix of order 1000. There will be no changes to the M-step of our EM algorithm since there is no change to the hierarchy of the priors. The only difference in implementation is the computation of the conditional expected distribution [see Equation (5.4)].

[1] Available at https://github.com/shuyu-tew/Sparse-Horseshoe-EM.git
[2] Available at https://cran.r-project.org/web/packages/bayesreg/index.html

Table 5.2 compares the performance of our proposed horseshoe estimator against the HS-Like estimator, the VisuShrink [55] estimates, and the BayesShrink [47] estimates of the lasso. Overall, the HS posterior mode estimates are competitive with or superior to the lasso estimates and outperform the horseshoe-like estimator in terms of MSE and the number of correctly identified zero coefficients. Most of the variables included in the HS-like estimator are zero coefficients incorrectly identified as non-zeros. This suggests model overfitting.

Linear Regression Model We considered $n = 70$, $p = 350$ and simulated 100 different data sets from the linear model, $\mathbf{y} \sim N_n\left(\mathbf{X}\boldsymbol{\beta},\ \sigma^2\boldsymbol{I}_n\right)$ with $\sigma^2 \in \{1, 9\}$. The predictor matrix, $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ was generated from a correlated multivariate normal distribution $N_p(0, \boldsymbol{\Sigma})$ with $\Sigma_{ij} = \rho^{|i-j|}$ and $\rho \in \{0, 0.7\}$. The first ten entries of $\boldsymbol{\beta}$ were set to 3, the next ten were set to -3, and the remaining entries were zero. The results are shown in Table 5.3. Overall, the HS-like estimator performed the best, obtaining the lowest MSE at the expense of substantial overfitting. The good performance in MSE under this setting is not unexpected, as this problem setting favors estimators that overfit (i.e., are less conservative) to a certain extent. Given a sparse vector with only 20 non-zero coefficients, this setting in which all non-zero coefficients have the same magnitude is one of the hardest settings for a sparse method, as excluding any one of the true coefficients is equally damaging. The prediction risk of this problem is $\mathrm{E}[||\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}||^2]$; if a method incorrectly includes an irrelevant predictor in the model, the risk will approximately be the variance of the estimate associated with this variable, i.e., $\mathrm{E}[\hat{\beta}_j^2] \approx \sigma^2/n$; in contrast, incorrectly excluding non-zero predictors will incur an error of 3^2, which is much larger than $\sigma^2/70$ even for the setting of greater noise, i.e., $\sigma^2 = 9$. In general, approximately half of the coefficients included in the HS-like model are false non-zeros (FNZ); while most of the coefficients included in our HS models are true non-zeros with a consistent small difference between TNZ and No.V.

The HS-EM, SCAD, and MCP methods all generally select much fewer variables than the HS-like estimator, with the HS-EM method overfitting less than HS-like and underfitting more than MCP or SCAD. In terms of MSE, our HS-EM method is competitive with, or superior to SCAD and MCP for all settings, despite producing substantially sparser estimates. It appears

Table 5.2: Performance of different sparse estimators (with associated standard error in parentheses) for the simulated data experiments on normal means model. Lasso$_{\mathrm{BS}}$ and Lasso$_{\mathrm{VS}}$ are the VisuShrink and BayesShrink estimates of the lasso respectively.

	HS-EM	HS-like	Lasso$_{\mathrm{BS}}$	Lasso$_{\mathrm{VS}}$	HS-EM	HS-like	Lasso$_{\mathrm{BS}}$	Lasso$_{\mathrm{VS}}$
		$\boldsymbol{\beta} = \{\mathbf{3}_{10}, -\mathbf{3}_{10}, \mathbf{0}_{980}\}$				$\boldsymbol{\beta} = \{\mathbf{10}_{10}, -\mathbf{10}_{10}, \mathbf{0}_{980}\}$		
MSE	148.6(1.6)	882.3(9.06)	**115.3**(2.3)	165.9(0.7)	**26.41**(1.6)	883.3(9.04)	309.1(2.9)	296.5(3.3)
Time	0.38(0.01)	0.003(0.00)	0.00(0.00)	0.00(0.00)	0.15(0.01)	0.002(0.00)	0.00(0.00)	0.00(0.00)
Inc$_{\mathrm{z}}$	0.07(0.03)	528.4(25.2)	25.44(2.1)	0.17(0.05)	0.07(0.03)	534.6(25.6)	471.6(2.3)	0.17(0.05)
No.V	3.86(0.19)	548.2(25.2)	39.5(2.39)	4.99(0.19)	20.07(0.5)	554.6(25.6)	491.6(2.3)	20.17(0.1)

Table 5.3: Performance of different sparse estimators (with associated standard error in parentheses) for the simulated data experiments.

	HS_{EM}	HS_{apxEM}	HS_{stEM}	HS-like	Lasso	MCP	SCAD	Ridge
$(\rho = 0, \sigma^2 = 1)$								
MSE	160.8(3.2)	160.7(2.9)	180.5(0.5)	**119.8**(3.7)	139.1(4.7)	164.1(2.1)	149.3(1.8)	176.2(0.6)
Time	5.22(0.19)	1.27(0.03)	1.38(0.39)	0.28(0.01)	0.17(0.01)	0.21(0.01)	0.32(0.01)	1.10(0.01)
No.V	6.69(0.40)	6.47(0.40)	0.16(0.04)	43.2(0.53)	21.0(1.57)	8.27(0.52)	18.6(0.73)	350(0.00)
TNZ	4.71(0.31)	4.61(0.30)	0.14(0.04)	14.2(0.32)	8.62(0.61)	4.63(0.29)	8.20(0.33)	20.0(0.00)
FNZ	1.98(0.19)	1.86(0.19)	0.02(0.01)	29.1(0.72)	12.4(1.03)	3.64(0.28)	10.4(0.52)	330(0.00)
$(\rho = 0, \sigma^2 = 9)$								
MSE	170.6(3.1)	169.4(2.9)	180.5(0.4)	**141.5**(3.5)	150.1(2.8)	163.4(1.9)	153.3(1.9)	176.5(0.5)
Time	5.57(0.16)	1.29(0.03)	1.62(0.60)	0.26(0.01)	0.17(0.01)	0.21(0.01)	0.33(0.01)	1.11(0.01)
No.V	5.56(0.35)	5.43(0.36)	0.16(0.05)	46.2(0.29)	17.8(1.66)	7.67(0.46)	18.3(0.75)	350(0.00)
TNZ	3.81(0.28)	3.80(0.28)	0.14(0.04)	13.2(0.31)	6.99(0.56)	4.35(0.28)	7.68(0.33)	20.0(0.00)
FNZ	1.75(0.18)	1.63(0.16)	0.02(0.01)	33.1(0.37)	10.8(1.18)	3.32(0.25)	10.6(0.53)	330(0.00)
$(\rho = 0.7, \sigma^2 = 1)$								
MSE	12.3(1.26)	14.9(1.56)	208.9(8.01)	**3.79**(0.57)	6.30(0.48)	78.7(2.81)	71.9(2.92)	423.9(3.64)
Time	2.51(0.14)	0.91(0.03)	36.9(2.30)	0.25(0.01)	0.13(0.01)	0.14(0.01)	0.16(0.01)	1.14(0.01)
No.V	17.1(0.24)	16.5(0.28)	3.7(0.14)	24.9(0.46)	33.9(0.48)	13.7(0.32)	22.2(0 44)	350(0.00)
TNZ	17.0(0.24)	16.4(0.28)	3.67(0.14)	19.7(0.05)	19.8(0.05)	8.14(0.14)	9.67(0.16)	20.0(0.00)
FNZ	0.08(0.03)	0.09(0.03)	0.03(0.02)	5.19(0.51)	14.2(0.46)	5.55(0.31)	12.5(0.44)	330(0.00)
$(\rho = 0.7, \sigma^2 = 9)$								
MSE	33.6(1.81)	37.8(1.90)	198.8(8.3)	36.3(1.38)	**26.6**(1.18)	88.1(3.40)	85.6(3.22)	431.18(3.98)
Time	4.48(0.21)	1.26(0.03)	39.7(2.56)	0.27(0.01)	0.15(0.01)	0.14(0.01)	0.17(0.01)	1.16(0.01)
No.V	13.2(0.28)	12.4(0.28)	3.88(0.14)	40.8(0.27)	28.7(0.58)	13.1(0.28)	22.2(0.48)	350(0.00)
TNZ	12.9(0.28)	12.1(0.27)	3.85(0.14)	18.7(0.11)	18.4(0.08)	7.59(0.13)	8.99(0.16)	20.0(0.00)
FNZ	0.28(0.05)	0.25(0.05)	0.03(0.02)	22.1(0.28)	10.3(0.56)	5.52(0.29)	13.2(0.45)	330(0.00))

that both SCAD and MCP are sensitive to the correlation structure of the data; their performance is comparable to our HS-EM approach when there is no correlation, but when correlation is introduced to the data, both methods performed substantially worse in terms of MSE. Such behavior is not observed in our proposed HS estimator. Of the two proposed HS estimators, HS-apx appears to be approximately three times faster than HS-EM. Otherwise, both estimators give roughly similar performance.

Unfortunately, the stochastic EM version of the HS estimator does not perform as well as the original, non-stochastic version. It tends to result in an excessively sparse solution, essentially setting most coefficients to zero. Currently, the cause for this behavior is not entirely clear. We suspect it may be related to fine-tuning the smoothing factor, as the algorithm appears sensitive to this parameter. Interestingly, this problem does not arise in our experiments with real datasets. Investigating and addressing this discrepancy is an area for future research.

5.1.2.2 Real data

We further analyze the performance of our proposed method on six real-world datasets. All datasets are available for download from the UCI machine learning repository [16] unless mentioned otherwise. Each dataset is randomly split such that the training data has a sample size of n with the remaining $N - n$ datapoints used as testing samples. This procedure is repeated 100 times and the averaged summary statistics for the performance measures are presented in Tables 5.5 and 5.6 for the linear regression and logistic regression experiments, respectively.

Linear Regression Table 5.4 presents the posterior mean estimates and their 95% credible intervals (CI) for the diabetes data [58] ($N = 442, P = 10$). For comparison, we included the posterior mode estimates computed from the proposed HS-EM estimator. The HS-EM posterior mode estimates are very similar to the corresponding posterior means, and all posterior mode estimates are within the corresponding 95% credible intervals; however, the HS-EM posterior mode provides a sparse point estimate as it excludes all variables (except S3) with 95% CIs that include zero.

In addition to the diabetes data, we also analyzed the benchmark Boston housing data ($N = 506, P = 14$), the concrete compressive strength dataset ($N = 1030, P = 9$) and the eye data ($N = 120, P = 200$) [152]. To make the problem more difficult, for each dataset we added a number of noise variables generated using the same procedure described in Section 5.1.2.1; in all cases, we added $p = 15$ additional noise variables, with $\rho = 0.8$. Model fitting is done using all the $P + 15$ (original plus noise) predictors as well as all interactions and possible transformations of the variables (logs, squares, and cubics).

Overall, our proposed HS estimator performs the best, attaining the lowest MSE on all real data, while generally selecting the simplest models (included the lowest number of variables). Similar to the results observed in Section 5.1.2.1, the results for HS-apx are virtually identical to the exact HS-EM procedure but with superior computational efficiency. Although HS_{stEM} performs slightly worse than HS_{EM} and HS_{apxEM}, its results remain comparable and often outperform the

Table 5.4: The posterior mode and mean estimates of the predictors for the diabetes data using the horseshoe estimator. The posterior mean and 95% credible intervals are computed using the `bayesreg` package in R.

	AGE	SEX	BMI	BP	S1	S2	S3	S4	S5	S6
Mean	-0.009	-18.68	5.769	1.034	-0.223	0.013	-0.592	2.419	48.84	0.179
2.50%	-0.341	-30.93	4.371	0.571	-0.937	-0.342	-1.415	-3.462	32.24	-0.225
97.5%	0.326	-5.144	7.109	1.457	0.098	0.656	0.189	11.36	70.14	0.734
Mode	·	-17.54	5.741	1.021	·	·	-0.909	·	43.58	·

Table 5.5: Performance of different sparse estimators (with associated standard error in parentheses) on the 4 datasets for linear regression model.

	HS_{EM}	HS_{apxEM}	HS_{stEM}	HS-like	Lasso	MCP	SCAD	Ridge
Diabetes ($n = 100, p = 385$)								
MSE	**3394**(23.1)	3416(24.9)	3664(37.5)	13515(323)	3645(30.3)	3557(31.6)	3627(33.9)	4177(39.2)
Time	2.92 (0.15)	1.01 (0.02)	4.55 (0.55)	0.32 (0.01)	0.46 (0.01)	1.14 (0.01)	1.43 (0.02)	1.27 (0.01)
No.V	1.56 (0.07)	1.46 (0.06)	1.39 (0.05)	95.5 (0.23)	3.80 (0.24)	2.73 (0.23)	6.41 (0.54)	385 (0.00)
Boston Housing ($n = 100, p = 473$)								
MSE	26.53(0.44)	**26.52**(0.45)	32.41(1.05)	60.69(1.05)	31.71(0.77)	81.57(47.5)	410.3(375)	50.73(0.89)
Time	4.14(0.16)	1.74(0.02)	12.9(1.16)	0.67(0.03)	0.24(0.01)	1.11(0.01)	1.31(0.01)	1.87(0.01)
No.V	2.78(0.19)	2.82(0.11)	2.19(0.08)	47.1(0.55)	4.94(0.44)	4.75(0.41)	10.9(0.87)	473(0.00)
Concrete ($n = 100, p = 327$)								
MSE	**72.44** (1.95)	72.88 (1.96)	109.1 (5.17)	234.2 (19.6)	82.26 (1.41)	97.86 (4.47)	139.6 (32.5)	177.3 (1.91)
Time	2.39 (0.17)	0.86 (0.03)	12.4 (0.63)	0.29 (0.01)	0.27 (0.01)	0.73 (0.01)	0.89 (0.01)	0.96 (0.01)
No.V	5.34 (0.08)	5.23 (0.07)	3.70 (0.08)	67.2 (0.41)	9.96 (0.44)	6.81 (0.31)	12.4 (0.53)	327 (0.00)
Eye ($n = 100, p = 200$)								
MSE	**0.90** (0.04)	0.92 (0.04)	1.19 (0.06)	1.92 (0.17)	1.01 (0.08)	0.92 (0.04)	0.96 (0.05)	1.04 (0.07)
Time	0.47 (0.01)	0.21 (0.01)	2.91 (0.31)	0.13 (0.01)	0.25 (0.01)	0.13 (0.01)	0.18 (0.01)	0.36 (0.01)
No.V	3.82 (0.08)	3.58 (0.08)	2.17 (0.06)	0.00 (0.00)	18.5 (0.64)	6.18 (0.25)	9.88 (0.27)	200 (0.00)

non-convex estimators namely HS-like, SCAD, and MCP, while also selecting simpler models with fewer predictors.

SCAD and MCP exhibit comparable performance to HS in terms of prediction error for the diabetes data and eye data, but have relatively poor performance for Boston and concrete data. Interestingly, while HS-like dominated the other non-convex estimators in terms of MSE on the simulated experiments, it performed quite poorly on the real data analysis, particularly on the diabetes and Boston housing data. The HS-like estimator tended to include the highest number of variables, suggesting it is potentially prone to overfitting as discussed in Section 5.1.2.1. Ridge regression generally performs worse than the sparse estimators, as would be expected given the experimental design.

Logistic Regression We also tested the HS-EM estimator on two binary classification problems: the Pima Indians data ($N = 768, P = 8$) and the heart disease data ($N = 302, P = 14$). Similar to the linear regression analysis, we augment the data with 10 noise variables and model the predictors together with their interactions and transformations. For the heart disease data, we only included the interactions between the variables and not the transformations, as there were a substantial number of categorical predictors. For this experiment, we varied the number of training samples $n \in \{100, 200\}$.

The HS-EM estimator obtains the best classification accuracy and negative log-loss on the Pima data for both training sample sizes. Ridge regression, on the other hand, performs the best in

Table 5.6: Performance of different sparse estimators on the Pima and heart disease data. CA is the classification accuracy and NLL is the negative log-loss.

Dataset		HS_{EM}	HS_{apxEM}	Lasso	MCP	SCAD	Ridge
Pima				$(n = 100, p = 214)$			
	CA	**0.743**(0.01)	0.741(0.01)	0.704(0.01)	0.741(0.01)	0.735(0.01)	0.674(0.01)
	NLL	**358.1**(2.42)	358.8(2.55)	381.5(3.33)	360.9(4.12)	362.8(3.90)	400.7(2.63)
	Time	0.622(0.02)	0.533(0.01)	0.298(0.01)	1.475(0.07)	2.056(0.07)	0.251(0.01)
	No.V	1.760(0.09)	1.700(0.09)	3.780(0.25)	3.900(0.27)	8.480(0.59)	214.0(0.00)
				$(n = 200, p = 214)$			
	CA	**0.756**(0.01)	**0.756**(0.01)	0.736(0.01)	0.754(0.01)	**0.756**(0.01)	0.706(0.01)
	NLL	**290.8**(1.97)	291.1(1.96)	306.4(1.52)	292.7(2.32)	291.9(1.69)	320.6(1.01)
	Time	0.838(0.03)	0.733(0.02)	0.835(0.02)	3.150(0.16)	4.075(0.19)	0.424(0.01)
	No.V	3.020(0.15)	3.060(0.15)	4.740(0.25)	4.760(0.32)	8.980(0.59)	214.0(0.00)
Heart				$(n = 100, p = 400)$			
	CA	0.758(0.01)	0.751(0.01)	0.792(0.01)	0.786(0.01)	0.797(0.01)	**0.799**(0.01)
	NLL	107.4(1.73)	110.3(1.66)	99.89(0.95)	96.20(1.28)	**94.36**(0.93)	103.2(1.03)
	Time	3.943(0.09)	3.623(0.11)	0.237(0.01)	1.208(0.05)	1.483(0.06)	0.383(0.01)
	No.V	2.820(0.11)	2.800(0.12)	10.10(0.49)	7.080(0.28)	13.76(0.51)	400.0(0.00)
				$(n = 200, p = 400)$			
	CA	0.798(0.01)	0.788(0.01)	0.813(0.01)	0.802(0.01)	0.805(0.01)	**0.828**(0.01)
	NLL	47.86(0.95)	49.53(0.94)	46.31(0.41)	**44.90**(0.71)	45.05(0.73)	48.25(0.32)
	Time	5.629(0.14)	5.022(0.14)	0.570(0.01)	2.755(0.11)	3.517(0.12)	0.621(0.01)
	No.V	5.520(0.15)	5.480(0.14)	14.28(0.67)	9.340(0.37)	17.38(0.58)	400.0(0.00)

terms of classification accuracy for the heart data. This suggests that many of the predictors and interactions between the variables in the heart data are potentially correlated with the response variable. Therefore, a sparse estimator might not be suitable for modeling this data. Nonetheless, our HS methods give comparable results to the other sparse methods in terms of log-loss, and in general, as the sample size increases, the difference in performance between all estimators becomes less significant.

5.1.3 Discussion

In this section, we adapted the novel EM algorithm that allows us to efficiently find, for the first time, the exact, sparse posterior mode of the horseshoe estimator. The experimental results suggest that in comparison with state-of-the-art non-convex sparse estimators, the HS-EM estimator is quite robust to the underlying structure of the problem. Both the MCP and SCAD algorithms appear sensitive to correlation in the predictors, and the HS-like estimator, based on an approximation to the horseshoe prior, appears sensitive to the signal-to-noise ratio of the problem. In contrast, the HS-EM algorithm, even when not performing the best, appears to always remain competitive with the best performing method while selecting highly sparse models. Comparing the shrinkage profiles of the HS and HS-like estimator (see Figure 5.1), the mixed performance of the HS-like procedure, is possibly due to the fact that HS-like approach more aggressively zeros out coefficients, making it more sensitive to misspecification of the global shrinkage parameter. We also conjecture that the conventional EM framework, in which the shrinkage hyperparameters $\boldsymbol{\lambda}$ are treated as missing data, provides a poorer basis for estimation of the global hyperparameter, in comparison to our approach of treating the coefficients as missing data, which potentially integrates the uncertainty of the parameters more effectively into the Q-function. This is a topic for further investigation.

Due to its prior preference for dense models, ridge regression can perform much worse than sparse alternatives if the underlying problem is sparse. However, it remained competitive or superior to other sparse methods on several of the real datasets. This strongly suggests that a method that is adaptive to the unknown degree of sparsity would be beneficial; we anticipate extending our algorithm to the generalized horseshoe (5.1), and automatically tuning our prior to produce a technique that will adapt to the sparsity of the problem.

Finally, given the relative simplicity of the M-step updates in this framework, it is of interest to consider extensions to non-linear models, such as neural networks. In this case, the required conditional expectations are unlikely to be available in closed form, but given access to a Gibbs sampler (or approximate Gibbs sampler) the procedure can easily be extended to utilize a standard stochastic EM implementation. This could potentially provide a simple, and (in expectation) exact, alternative to variational Bayes for these problems. While our current experimental results show that our stochastic EM algorithm doesn't yet perform as well as the non-stochastic EM estimators, we hold the belief that with careful algorithm tuning, including adjustments like refining the smoothing factor, or by exploring different variations of the MCEM algorithm, this area holds significant promise for future research.

5.2 Sparse Lasso Estimator

The Lasso estimate for linear regression parameters can be interpreted as a Bayesian posterior mode estimate when assigning an independent double-exponential (Laplace) prior to each individual regression parameter β_j:

$$\beta_j | \tau, \sigma \sim \mathrm{La}\left(0,\ \tau\sigma\right). \tag{5.9}$$

There has been a significant amount of research dedicated to this Bayesian representation of the Lasso regression with Park and Casella [134] being among the first to provide the explicit treatment of Bayesian Lasso regression. Park and Casella [134] introduced a Gibbs sampling method to explore the posterior distribution. Subsequently, Hans [78] demonstrated that, for a given set of

$n \geq p$ predictors, the posterior distribution of $\boldsymbol{\beta}$ takes on the form of a mixture of orthant-specific normal distributions. Both studies used the scale mixture of normals representation of the Laplace distribution (with an exponential mixing distribution) to establish a hierarchical model, with Park and Casella [134] advocating the use of the posterior median as the point estimate, while Hans [78] looked into the posterior mean. Furthermore, various hierarchical formulations of Bayesian Lasso have been introduced. For instance, Mallick and Yi [113] presented the scale mixture of the uniform representation of the Laplace distribution with a specific gamma mixing density, and Alhamzawi and Taha Mohammad Ali [7] introduced the scale mixture of the truncated normal representation of the Laplace density with exponential mixing densities. Our work specifically focuses on Bayesian Lasso regression with the scale mixture normal representation of the Laplace prior. Nonetheless, it is important to acknowledge these diverse representations to highlight the extensive research interest in the Bayesian Lasso model, which has found applications in various domains, including negative binomial regression [64], genome-wide association studies [105], and autoregressive models [153], among others.

While the fully Bayesian approach allows for richer inferences about the models compared to most non-Bayesian analyses, practical considerations often call for faster computations focused on posterior modes rather than fully exploring the posterior distribution. Various methods for finding posterior modes are available, including the conventional EM algorithm and its extensions [62, 183], as well as the coordinate-wise descent algorithm [153]. In this section, we introduce an alternative approach to obtaining the Bayesian Lasso posterior mode estimate by adopting the novel EM algorithm introduced in Chapter 3. We examine the properties of our EM-based Bayesian Lasso posterior mode estimate (Lasso-EM) and compare it with existing Lasso estimates.

5.2.1 Lasso MAP Estimation via EM

We demonstrate the application of the EM procedure discussed in Chapter 3 to the Bayesian Lasso, providing explicit EM updates for each of the shrinkage parameters. We consider the scale mixture of normals representation of the Laplace distribution (5.9):

$$\begin{aligned} \beta_j | \lambda_j, \tau^2, \sigma^2 &\sim N(0, \lambda_j^2 \tau^2 \sigma^2), \\ \lambda_j^2 &\sim \mathrm{Exp}(2), \end{aligned} \tag{5.10}$$

for which each individual local shrinkage parameter is assigned an independent exponential hyperprior. Following established conventions [78, 134], we assign an inverse gamma prior on the global shrinkage parameter τ^2:

$$\tau^2 \sim \mathrm{IG}(1, 1)$$

where $\mathrm{IG}(a,b)$ is the inverse-gamma distribution with probability density function

$$p(z|a,b) = \frac{b^a}{\Gamma(a)} z^{-a-1} \exp\left(-\frac{b}{z}\right).$$

Substituting the negative logarithm of the exponential distribution for λ_j^2 and the inverse-gamma distribution for τ into (3.4) – (3.5), holding τ^2 and σ^2 fixed yields the M-step update for each λ_j^2:

$$\begin{aligned}
\hat{\lambda}_j^2 &= \underset{\lambda_j^2}{\operatorname{argmin}} \left\{ \frac{\log \lambda_j^2}{2} + \frac{W_j}{\lambda_j^2} + \frac{\lambda_j^2}{2} \right\} \\
&= \frac{1}{2}(\sqrt{1+8W_j} - 1) && (5.11) \\
(\hat{\tau}^2, \hat{\sigma}^2) &= \underset{\tau^2,\sigma^2}{\operatorname{argmin}}\, Q(\tau^2, \sigma^2; \hat{\boldsymbol{\lambda}}^2) && (5.12)
\end{aligned}$$

where $W_j = E[\beta_j^2]/(2\sigma^2\tau^2)$. The negative log-prior and the M-step update for the laplace prior are summarised in Table 5.1. Given the $\hat{\boldsymbol{\lambda}}^2$ updates (5.6), the $\hat{\tau}$ and $\hat{\sigma}^2$ estimates can be found using numerical optimisation. In line with Section 5.1.1, we can reduce the computational complexity of the optimization problem by (approximately) estimating σ^2 using the expected residual sum-of-squares and effectively reduce the two-dimensional optimization problem (5.12) to a one-dimensional optimization problem (5.8).

In the context of a normal means model, where we observe data in the form of a n dimensional vector $\boldsymbol{y}|\boldsymbol{\beta} \sim N(\boldsymbol{\beta}, \sigma^2)$ and $\boldsymbol{\beta}$ follows the Laplace distribution as specified in (5.9), Hans [78] presented the posterior mean estimate for $\boldsymbol{\beta}$ in terms of the least squares estimate $\hat{\boldsymbol{\beta}}_{\mathrm{OLS}}$:

$$\mathbb{E}\left[\boldsymbol{\beta}|\boldsymbol{y}, \sigma, \tau\right] = \hat{\boldsymbol{\beta}}_{\mathrm{OLS}} + \sigma\tau^{-1}(2\omega - 1). \tag{5.13}$$

The weight ω is computed as follows:

$$\omega = \frac{\boldsymbol{\Phi}(\frac{-\boldsymbol{\mu}_-}{\sigma^2})/\mathrm{N}(0|\boldsymbol{\mu}_-, \sigma^2)}{\boldsymbol{\Phi}(\frac{-\boldsymbol{\mu}_-}{\sigma^2})/\mathrm{N}(0|\boldsymbol{\mu}_-, \sigma^2) + \boldsymbol{\Phi}(\frac{\boldsymbol{\mu}_+}{\sigma^2})/\mathrm{N}(0|\boldsymbol{\mu}_+, \sigma^2)}$$

where $\boldsymbol{\Phi}$ is the standard normal cumulative distribution function, and the two location parameters are $\boldsymbol{\mu}_- = \hat{\boldsymbol{\beta}}_{\mathrm{OLS}} + \sigma\tau^{-1}$ and $\boldsymbol{\mu}_+ = \hat{\boldsymbol{\beta}}_{\mathrm{OLS}} - \sigma\tau^{-1}$. The posterior mode estimates, on the other hand, can be found by minimizing the following expression:

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{\mathrm{MAP}} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ -\log p(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2) - \log \pi(\boldsymbol{\beta}) \right\} \\
&= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \frac{n}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \beta_i)^2 + \log(2\sigma\tau) + \sum_{i=1}^{n}\frac{|\beta_j|}{\sigma\tau} \right\} \\
&= \operatorname{sgn}(\boldsymbol{y})\left(|\boldsymbol{y}| - \frac{\sigma}{\tau}\right)_+ && (5.14)
\end{aligned}$$

where $(x)_+ = \max\{x, 0\}$ and $\mathrm{sgn}(\cdot)$ is the sign function. Within the normal means model framework, we can compare the behavior of the Lasso-EM posterior mode estimate with this exact Bayesian Lasso posterior mode estimate. Theorem 5.1 demonstrates that using the EM updates provided in (5.11) to compute the Bayesian posterior mode estimate, it recovers the exact Bayesian Lasso posterior mode estimate for large observations $\boldsymbol{y}$.

Theorem 5.1. *Suppose $y|\theta, \sigma \sim N(\theta, \sigma^2)$. Let $\hat{\theta}_{\text{MAP}}$ denote the exact posterior mode estimate of the Bayesian Lasso, i.e. where $\theta|\tau, \sigma \sim \mathrm{La}(0, \tau\sigma)$. Let $\hat{\theta}_{\text{EMMAP}}$ be the Bayesian Lasso posterior mode estimated using the updates given in (5.11). Then $\hat{\theta}_{\text{EMMAP}} \to \hat{\theta}_{\text{MAP}}$ as $|y| \to \infty$, .*

Proof. Given the λ^2 update rule in (5.11), we can replace W with $W = E[\theta^2|y, \lambda^2, \tau^2, \sigma^2]/(2\sigma^2\tau^2)$ which gives

$$\hat{\lambda}^2_{t+1} = \left(-1 + \sqrt{1 + \frac{8\,\mathbb{E}\left[\theta^2|y, \lambda^2, \tau^2, \sigma^2\right]}{2\sigma^2\tau^2}}\right)/2.$$

Using equation (5.4) and the property $\mathbb{E}\left[\theta^2\right] = \mathrm{Var}[\theta] + \mathbb{E}\left[\theta\right]^2$, we find that $E[\theta^2|y, \lambda^2, \tau^2, \sigma^2] = (1-\kappa)\sigma^2 + ((1-\kappa_i)y)^2$. By definition, when the EM algorithm converges, the current parameter estimate will be equal to the previous parameter estimate $\hat{\lambda^2_{t+1}} = \lambda^2$. This convergence corresponds to reaching a stationary point where no further changes occur in the estimates. In this case, we denote both the current estimates, $\hat{\lambda^2_{t+1}}$, and the previous parameter estimates, as λ^2.

$$\lambda^2 = \left(-1 + \sqrt{1 + \frac{8\sigma^2(1-\kappa) + 8((1-\kappa)y)^2}{2\sigma^2\tau^2}}\right)/2.$$

With $\kappa = 1/(1 + \lambda^2\tau^2)$, we solve for the λ^2 with the assumption that $\tau^2 > 0$, $\lambda^2 > 0$ and $y > 0$, this gives us:

$$\lambda^2 = \frac{\sqrt{\sigma^2\tau^2 + 4y^2}}{2\sqrt{\sigma^2}\sqrt{\tau^2}} - \frac{1}{2} - \frac{1}{\tau^2}. \tag{5.15}$$

This is the "closed-form" solution for λ^2 of our Bayesian Lasso posterior mode estimate. Using this closed-form solution, the EM-driven Bayesian Lasso posterior mode estimate is obtained as the expected value of θ using the conditional posterior distribution given in equation 5.4. These estimates can be computed as

$$\hat{\theta}_{\text{EMMAP}} = E[\theta|\lambda^2] = y(1 - 1/(1 + \lambda^2\tau^2)). \tag{5.16}$$

Substituting the "closed-form" expression for λ^2 (5.15), we have

$$E[\theta|\lambda^2] = y + \frac{2\sqrt{\sigma^2}y}{\sqrt{\sigma^2\tau^2} - \sqrt{\tau^2}\sqrt{\sigma^2\tau^2 + 4y^2}}. \tag{5.17}$$

The latter term in the equation, i.e., $\frac{2\sqrt{\sigma^2}y}{\sqrt{\sigma^2\tau^2}-\sqrt{\tau^2}\sqrt{\sigma^2\tau^2+4y^2}}$, can be viewed as a threshold term. As $y \to \infty$ and $y \to -\infty$, the limit of this threshold function is:

$$\lim_{y\to\infty} \frac{2\sqrt{\sigma^2}y}{\sqrt{\sigma^2\tau^2}-\sqrt{\tau^2}\sqrt{\sigma^2\tau^2+4y^2}} = -\frac{\sqrt{\sigma^2}}{\sqrt{\tau^2}}$$
$$\lim_{y\to-\infty} \frac{2\sqrt{\sigma^2}y}{\sqrt{\sigma^2\tau^2}-\sqrt{\tau^2}\sqrt{\sigma^2\tau^2+4y^2}} = \frac{\sqrt{\sigma^2}}{\sqrt{\tau^2}}$$

So it follows that

$$\hat{\theta}_{\text{EMMAP}} \to \text{sgn(y)}\left(|\text{y}| - \frac{\sqrt{\sigma^2}}{\sqrt{\tau^2}}\right) \qquad \text{when} \qquad |\text{y}| \to \infty$$

As $|y| \to \infty$, the Bayesian Lasso posterior mode estimates produced by the EM algorithm with the λ^2 updates from equation 5.11 converge to the exact Bayesian lasso MAP estimate given in equation 5.14 which we will repeat here for convenience:

$$\hat{\theta}_{\text{MAP}} = \text{sgn(y)}\left(|\text{y}| - \frac{\sigma}{\tau}\right)_+$$

Hence,

$$\hat{\theta}_{\text{EMMAP}} \to \hat{\theta}_{MAP} \qquad \text{when} \qquad |y| \to \infty.$$

□

Theorem 5.1 establishes the coincidence of the posterior mode estimates produced by the EM algorithm and the exact Bayesian Lasso MAP estimates as $|y| \to \infty$. This convergence is reflected in the shrinkage profile in Figure 5.4 (second row), where, with increasing values of $|y|$, the EM estimates (yellow line) progressively approach the exact Bayesian Lasso posterior mode estimate (blue line). Consequently, the tails of both estimators in the shrinkage profile move closer to one another as $|y| \to \infty$.

5.2.2 Real data results

We assess the performance of the proposed Lasso-EM estimator on 11 real-world datasets from the UCI machine learning repository [16] (unless referenced otherwise), in terms of predictive performance, measured in R^2 on the test data, and computational efficiency, within the context of normal linear regression problems. We compare the Lasso-EM method against the Lasso estimator implemented from the `glmnet` R package with the default settings, in which 10-fold cross-validation is used to select the Lasso regularization parameter. For the proposed EM algorithm, we set the convergence threshold to be $\epsilon = 10^{-5}$. Similar to Chapter 4.6.2, the linear regression experiments involve three settings: (i) standard linear regression; (ii) second-order multivariate polynomial

Table 5.7: Real datasets experiment results. The first column is the dataset (abbreviated, refer to Appendix A for the full name), the training sample size n, the raw number of features p; T_{10} is the time taken ratio of a single 10-fold CV lasso regression fit to a single EM-Lasso fit and $\mathrm{T}_{\mathrm{loo}}$ is the time taken ratio of the LOOCV lasso regression fit to a single EM-Lasso fit; p^* is the number of features including interactions; EM and CV_{10} are the R^2 values on the test data for the EM-Lasso and 10-fold CV lasso respectively.

		Linear						2nd Order				3rd Order			
Dataset	n	p	EM	CV_{10}	$\mathrm{CV}_{\mathrm{loo}}$	T_{10}	$\mathrm{T}_{\mathrm{loo}}$	p^*	EM	CV_{10}	T_{10}	p^*	EM	CV_{10}	T_{10}
PT - motor	4112	19	0.15	0.13	0.13	0.4	35	208	0.25	0.24	0.3	1539	0.26	0.24	0.3
PT - total	4112	19	0.17	0.15	0.15	0.2	22	208	0.24	0.25	0.2	1539	0.26	0.24	0.3
Abalone	2923	9	0.52	0.51	0.50	5.6	333	51	0.42	0.52	0.5	209	0.31	0.52	0.5
Airfoil	1052	5	0.50	0.48	0.48	17	295	20	0.62	0.61	1.4	55	0.63	0.61	1.5
Concrete	721	8	0.60	0.58	0.58	5.4	74	44	0.85	0.85	1.1	164	0.85	0.85	1.0
F.Fires	361	12	-0.02	-0.01	-0.01	2.8	25	295	-0.09	-0.01	0.2	1984	-0.09	-0.01	0.2
B.Housing	354	13	0.71	0.68	0.68	6.9	50	104	0.82	0.82	0.3	559	0.82	0.82	0.2
Diabetes [1]	309	10	0.47	0.44	0.44	3.7	22	65	0.47	0.45	1.9	285	0.47	0.45	2.0
R.Estate	289	6	0.56	0.47	0.47	7.4	43	27	0.65	0.57	0.1	83	0.65	0.57	0.2
A.MPG	278	8	0.81	0.79	0.79	3.3	15	35	0.86	0.85	0.6	119	0.86	0.85	0.5
Yacht	215	7	0.97	0.96	0.96	12	49	27	0.98	0.97	1.5	83	0.98	0.97	1.2
Prostate [1]	64	8	0.53	0.49	0.50	3.9	5.2	47	0.46	0.45	3.2	55	0.44	0.45	2.4

[1] This dataset is not from the UCI repository. Data references can be found in the appendix.

regression with added interactions and second-degree polynomial transformations of variables, and (iii) third-order multivariate polynomial regression with added three-way interactions and cubic polynomial transformations. For each experiment, we repeated the process 100 times and used a random 70/30 train-test split.

Table 5.7 summarizes the experimental results. Specifically, it details the R^2 values on the held-out test set and the ratio of time taken to run 10-fold CV Lasso and LOOCV Lasso to the time taken by our EM procedure (T_{10} and $\mathrm{T}_{\mathrm{loo}}$). The R^2 value for the 10-fold CV Lasso is computed as an average value over 10 repeats, with the minimum and maximum values shown as error bars in Figure 5.2. The number of features, p, and observations, n reported are values after data preprocessing (missing observations removed, one-hot encoding transformation, etc.). The results demonstrate that the proposed EM algorithm offers comparable, if not superior, predictive performance when compared to 10-fold CV Lasso across most datasets. In cases where the EM-Lasso underperforms in comparison to the 10-fold CV Lasso, the latter exhibits significant variability in the R^2 value as well, as evidenced by the substantial error bars associated in Figure 5.2.

The insights from Table 5.7 may initially suggest that the proposed EM-Lasso is computationally less efficient than the 10-fold CV Lasso, potentially being up to 10 times slower. However, T_{10} records the time taken ratio of a single 10-fold CV lasso regression fit to a single EM-Lasso fit. The non-deterministic nature of the 10-fold CV Lasso often requires multiple runs (typically 10 or more repetitions) to obtain reliable results, significantly increasing the total computational

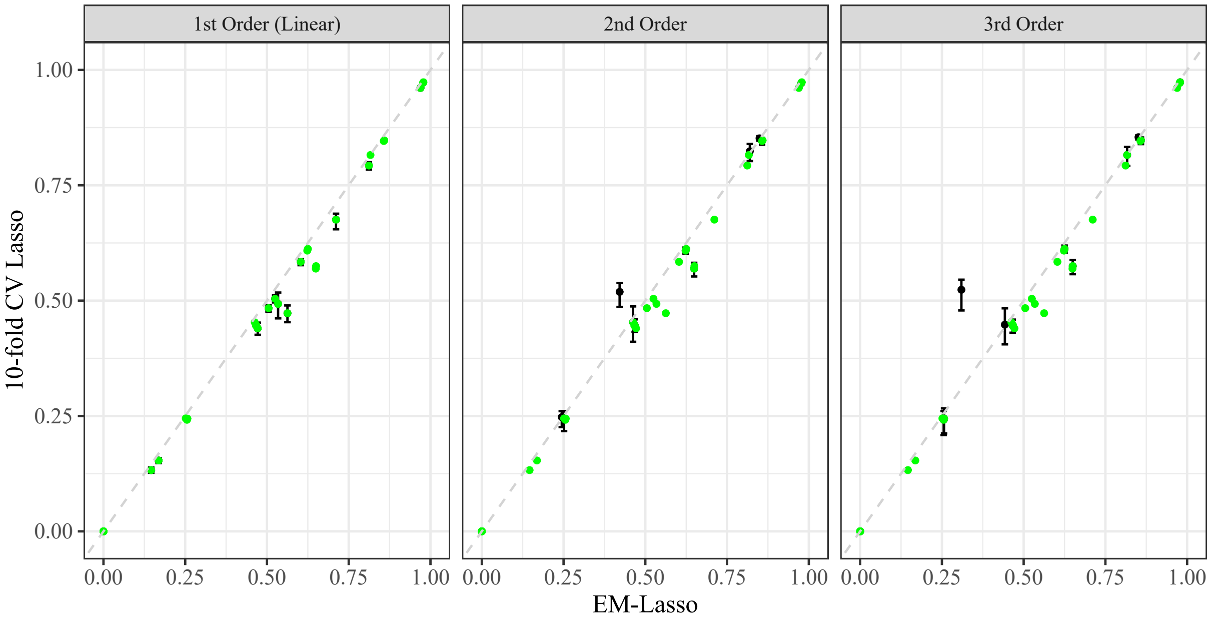


FIGURE 5.2: Comparison of predictive performance (R^2) of EM-Lasso (x-axis) against `glmnet` 10-fold CV Lasso (y-axis). Error bars indicate the minimum and maximum R^2 values observed for the 10-fold CV Lasso across 10 repetitions. Columns correspond to the results of linear features (left), second-order features (middle), and third-order features (right). Negative values are capped at 0. Points skewing toward the bottom right indicate when the EM-Lasso is giving better/same prediction performance as the 10-fold CV Lasso (colored in green).

time. In contrast, EM-Lasso provides deterministic results and offers a computational speed advantage, especially when compared to the total time taken for 10 repetitions (or more) of the 10-fold CV Lasso, without sacrificing predictive performance. Even when compared to a single 10-fold CV Lasso fit, the proposed EM-Lasso can be up to 17 times faster. This advantage becomes particularly evident when compared against the deterministic CV counterpart, LOOCV Lasso, where the EM-Lasso method consistently demonstrates superior computational speed [3]. The proposed EM algorithm is inherently deterministic in the sense that multiple runs on the same training data would give the exact same regression coefficient estimates. Aside from the convergence criteria, there are no other sources of variation in the estimates. Preliminary testing suggests that different error threshold values, whether set at 10^{-5}, 10^{-8}, or other values within a reasonable range, yield identical estimates. In contrast, the 10-fold CV approach can result in a range of estimates across different iterations, further complicated by the choice of possible γ grid.

5.3 MAP estimation across different parameterizations

While our HS-EM and Lasso-EM algorithm is able to locate, exactly, a sparse posterior mode of the horseshoe and Lasso estimator, there is an obvious question as to what mode we are finding.

[3] We only have results for standard linear regression as the application of LOOCV to larger datasets is overly time-consuming and may not be efficient for reproducibility.

Table 5.8: Prior densities of the three different parameterizations of the local shrinkage parameters: λ_j, λ_j^2 and $\upsilon_j = \log \lambda_j$, for both the horseshoe prior and Laplace prior, with its corresponding EM updates. Densities are given up to a constant. $w_j = E[\beta_j^2]/(2\sigma^2\tau^2)$

Prior for β_j	Density	Distribution	EM Updates
Lasso	$p(\lambda_j) = \lambda_j \; e^{-\frac{\lambda_j^2}{2}}$	$\lambda_j \sim \text{Rayleigh}(1)$	$\hat{\lambda}_j^{(t+1)} = (2w_j)^{\frac{1}{4}}$
	$p(\lambda_j^2) = \frac{1}{2}e^{-\frac{\lambda_j^2}{2}}$	$\lambda_j^2 \sim \text{Exp}(2)$	$\hat{\lambda^2}_j^{(t+1)} = \frac{-1+\sqrt{1+8w_j}}{2}$
	$p(\upsilon_j) = e^{2\upsilon_j - \frac{e^{2\upsilon_j}}{2}}$	$-\upsilon_j \sim \text{Gumbel}(-0.35, \frac{1}{2})$	$\hat{\upsilon}_j^{(t+1)} = \log\left(\frac{\sqrt{1+\sqrt{1+8w_j}}}{\sqrt{2}}\right)$
Horseshoe	$p(\lambda_j) = \frac{2}{\pi(1+\lambda_j^2)}$	$\lambda_j \sim \text{C}^+(0,1)$	$\hat{\lambda}_j^{(t+1)} = \frac{\sqrt{2w_j-1+\sqrt{1+20w_j+4w_j^2}}}{\sqrt{6}}$
	$p(\lambda_j^2) = \frac{1}{\pi(1+\lambda_j^2)\sqrt{\lambda_j^2}}$	$\lambda_j^2 \sim \text{B}'\left(\frac{1}{2}, \frac{1}{2}\right)$	$\hat{\lambda^2}_j^{(t+1)} = \frac{w_j-1+\sqrt{1+6w_j+w_j^2}}{4}$
	$p(\upsilon_j) = \frac{2e_j^{\upsilon}}{\pi(1+e^{2\upsilon_j})}$	$\upsilon_j \sim$ Hyperbolic Secant	$\hat{\upsilon}_j^{(t+1)} = \log\left(\frac{\sqrt{w_j+\sqrt{w_j}\sqrt{4+w_j}}}{\sqrt{2}}\right)$

Posterior modes (and means) are not invariant under reparameterization, which suggests that maximizing for λ_j (with appropriate transformation of prior distribution) in place of λ_j^2, for example, will produce a new shrinkage procedure with potentially different properties. In this section, we explore three distinct parameterizations of the local shrinkage parameter - λ_j, λ_j^2, and $\boldsymbol{\upsilon} = \log \lambda_j$. We specifically exclude the $\log \lambda_j^2$ parameterization because it yields the same EM estimates as using the prior in the form of $\boldsymbol{\upsilon}$, as mentioned in Proposition 5.2.

Proposition 5.2. *Let $\upsilon = \log \lambda$ and $\phi = \log \lambda^2$, it follows that $p(\phi) = 2p(\upsilon)$ and $\underset{\upsilon}{\text{argmin}}\{p(\upsilon)\} = \underset{\phi}{\text{argmin}}\{p(\phi)\}$, resulting in identical EM updates.*

Proof. Using the change of variable formula [128, Eq.2.87], we have:

$$p(\phi) = p(\upsilon)\frac{d\upsilon}{d\phi} = 2p(\upsilon) \tag{5.18}$$

□

The behavior of our EM posterior mode estimates can be analyzed by considering the estimation of β within the framework of a normal means problem with $\sigma^2 = 1$ where $y|\beta \sim \text{N}(\beta, 1)$. In Table 5.8, we provide the prior density for all reparameterizations considered for both the Horseshoe prior and the Laplace prior, along with the names of the probability distributions to which the reparameterized priors adhere. Figure 5.3 and 5.4 illustrates the corresponding shrinkage profiles

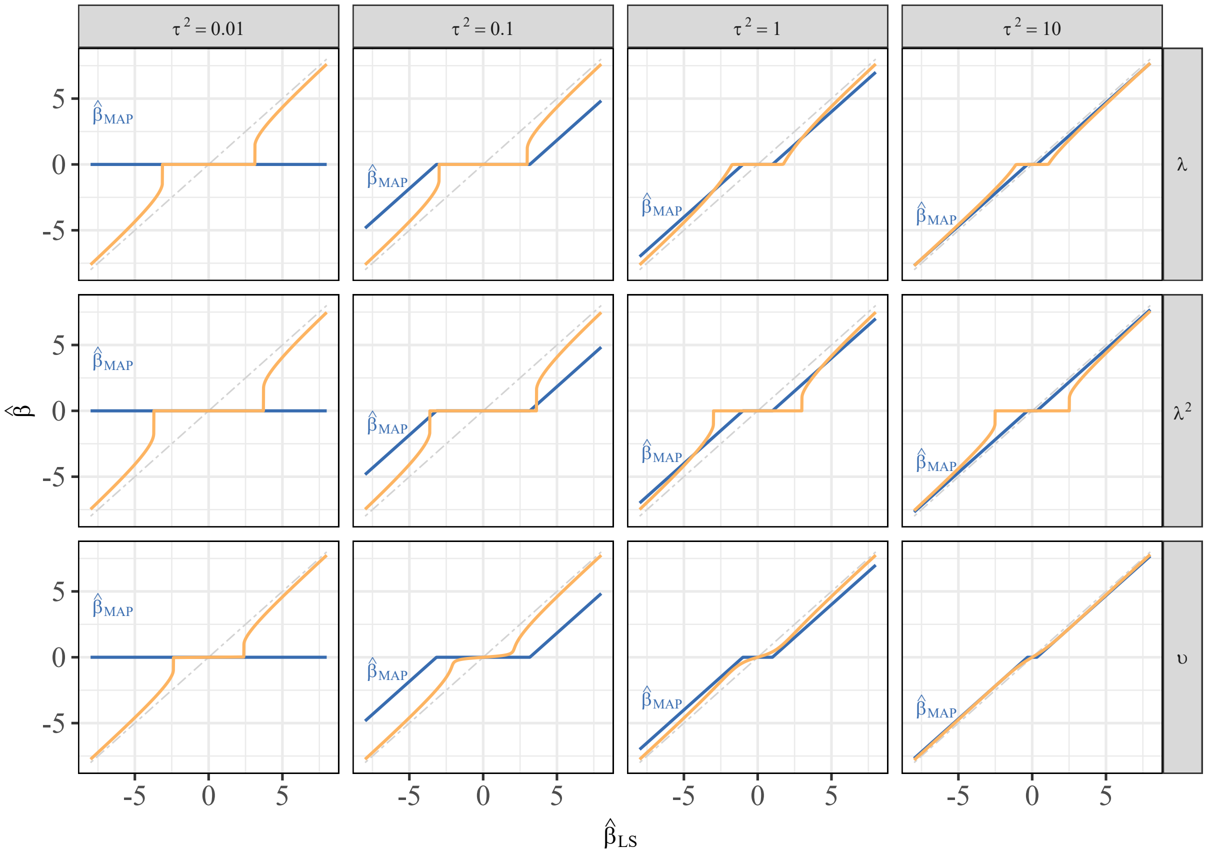


Figure 5.3: Shrinkage profile for the HS-EM estimator with varying τ^2 and three different parameterization of local shrinkage parameter: λ, λ^2 and $\upsilon = \log \lambda$. $\hat{\beta}_{\text{MAP}}$ (blue line) is the Bayesian lasso posterior mode estimates computed using Equation (5.14) and the yellow line is the posterior mode estimate of the HS-EM estimator.

by plotting the EM posterior mode estimates against the maximum likelihood (least-squares) estimates ($\hat{\boldsymbol{\beta}}_{\text{LS}} = \boldsymbol{y}$). Both the Lasso-EM and HS-EM results in a range of estimates each with varying degrees of sparsity. Notably, the Laplace prior with λ and $\log \lambda$ parameterizations yield non-sparse estimates resembling Bayesian Lasso posterior mean estimates $\hat{\boldsymbol{\beta}}_{\text{EAP}}$ of Equation (5.13). For the other sparse EM estimates, we compare them to the exact Bayesian Lasso posterior mode estimate, $\hat{\boldsymbol{\beta}}_{\text{MAP}}$ of Equation (5.14).

While both horseshoe and Lasso shrink small observations, the horseshoe leaves large coefficients unshrunk, whereas Lasso shrinks them by a non-vanishing amount, resulting in a non-zero bias. Furthermore, both HS-EM and Lasso-EM posterior mode estimates demonstrate an increased degree of shrinkage as τ^2 decreases. However, when $\tau^2 = 0.01$, the Bayesian Lasso and the Lasso-EM estimates result in uninformative, all-zero estimates. In contrast, the HS-EM estimates remain unaffected for larger observations. These observations highlight the advantage of the horseshoe being robust at handling sparsity and large signals, even with very small τ^2.

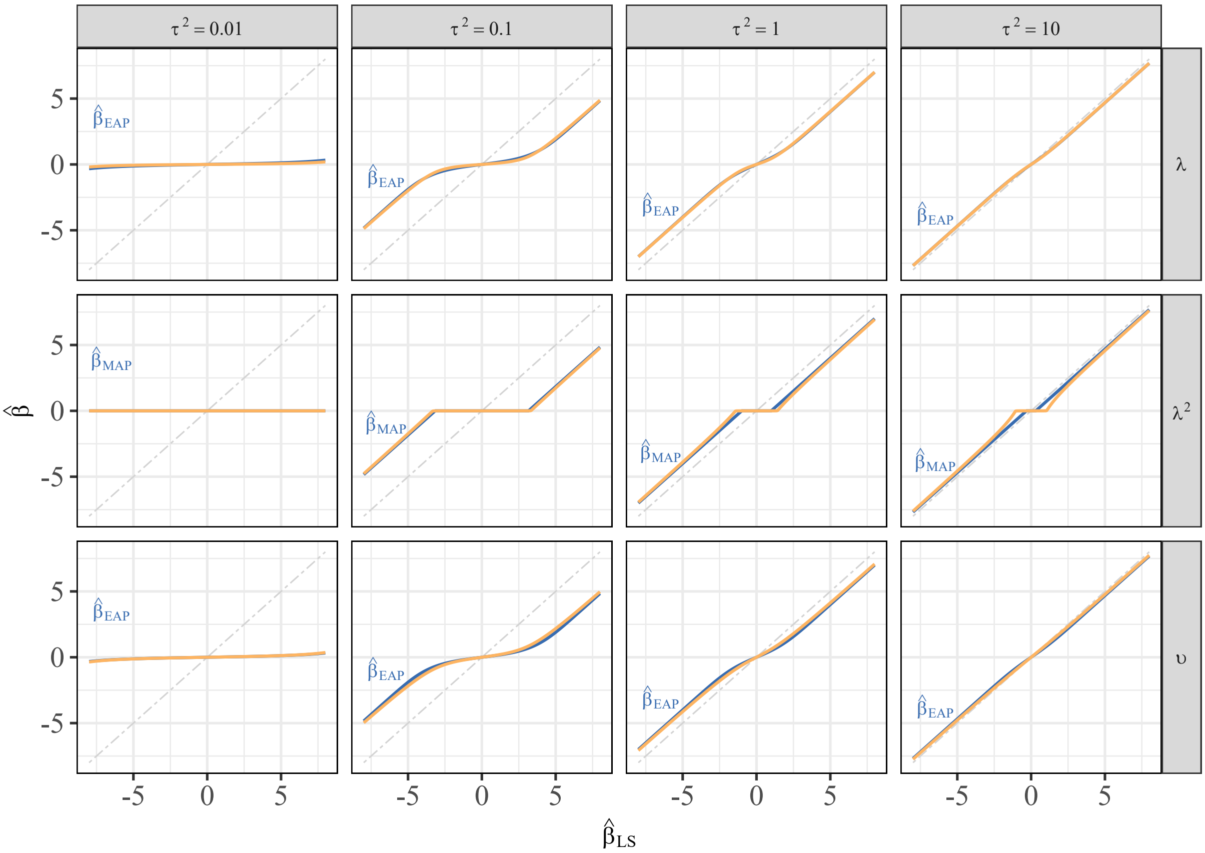


FIGURE 5.4: Shrinkage profile for the Lasso-EM estimator with varying τ^2 and three different parameterization of local shrinkage parameter: λ, λ^2 and $\upsilon = \log \lambda$. $\hat{\beta}_{\text{MAP}}$ is the Bayesian lasso posterior mode estimates computed using Equation (5.14), $\hat{\beta}_{\text{EAP}}$ is the Bayesian lasso posterior mean estimates computed using Equation (5.13) and the yellow line is the posterior mode estimate of the Lasoo-EM estimator.

5.4 Sparse Grouped Regression

To address regression problems involving grouped structures within the framework of the global-local shrinkage hierarchy (3.2), we present the Bayesian grouped half-Cauchy hierarchy of Xu et al. [185]:

$$\begin{aligned}
\mathbf{y}|\boldsymbol{X}, \boldsymbol{\beta}, \sigma^2 &\sim N_n\left(\boldsymbol{X}\boldsymbol{\beta},\ \sigma^2 \boldsymbol{I}_n\right) \\
\sigma^2 &\sim \sigma^{-2} d\sigma^2 \\
\beta_{gj}|\tau^2, \delta_g^2, \lambda_{gj}^2, \sigma^2 &\sim N\left(0,\ \tau^2 \lambda_{gj}^2 \delta_g^2 \sigma^2\right) \\
\lambda_{gj}^2 &\sim \pi(\lambda_{gj}^2) d\lambda_{gj}^2 \\
\delta_g^2 &\sim \pi(\delta_g^2) d\delta_g^2 \\
\tau &\sim \pi(\tau) d\tau.
\end{aligned} \tag{5.19}$$

whereby a new group shrinkage parameter δ_g is introduced. Here $g = 1, 2, \ldots, G$ indexes the groups, β_{gj} represents the regression coefficients for the predictors in group g such that $\boldsymbol{\beta}_g =$

$(\beta_{g1}, \ldots, \beta_{gj}, \ldots, \beta_{gp_g})$ with p_g being the number of variables in group g and the total number of predictors across G groups is $p = \sum_{g=1}^{G} p_g$. The group shrinkage parameter δ_g controls the amount of shrinkage applied to the predictors within group g, while the role of λ_{gj}^2 and τ remains the same: λ_{gj}^2 determines the local shrinkage of individual regression coefficient within each group and τ controls the overall shrinkage. In line with the previous Chapters, we adhere to the same prior specification for the global shrinkage parameter: $\tau \sim C^+(0,1)$ for the horseshoe prior and $\tau^2 \sim \mathrm{IG}(1,1)$ for the Laplace prior.

In the context of the grouped regression hierarchy (5.19), the updated EM procedure is presented as follows:

E-step. Find the parameters of the *Q-function*, i.e., the expected complete negative log-posterior (with respect to $\boldsymbol{\beta}$), conditional on the current estimates of the shrinkage parameters ($\boldsymbol{\lambda}$, $\boldsymbol{\delta}$, τ), noise variance σ^2, and the observed data $\mathbf{y}$:

$$\begin{aligned}
&Q(\boldsymbol{\lambda}, \boldsymbol{\delta}, \tau, \sigma^2 | \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}) \\
&\quad = \mathbb{E}_{\boldsymbol{\beta}}\left[-\log p(\boldsymbol{\beta}, \boldsymbol{\lambda}, \boldsymbol{\delta}, \tau, \sigma^2 \,|\, \mathbf{y}) \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}}, \mathbf{y} \right] \\
&\quad = \left(\frac{n+p+2}{2}\right) \log \sigma^2 + \frac{\mathbb{E}_{\boldsymbol{\beta}}\left[||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right]}{2\sigma^2} + \frac{p}{2} \log \tau^2 \\
&\qquad + \frac{1}{2} \sum_{g=1}^{G} \sum_{j=1}^{p_g} (\log \lambda_{gj}^2 + \log \delta_g^2) + \frac{1}{2\sigma^2\tau^2} \sum_{g=1}^{G} \sum_{j=1}^{p_g} \frac{\mathbb{E}_{\boldsymbol{\beta}}\left[\beta_{gj}^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right]}{\lambda_{gj}^2 \delta_g^2} - \log \pi(\boldsymbol{\lambda}, \boldsymbol{\delta}, \tau)
\end{aligned} \tag{5.20}$$

In the case where $G = 1$ and $\boldsymbol{\delta} = 1$ (i.e. no grouped predictors), this Q-function coincides with (3.4). However, if a prior is placed on δ, when $G = 1$, this Q-function takes on a different structure compared to (3.4), with an additional global shrinkage parameter $\boldsymbol{\delta}$.

M-step. Update the parameter estimates by minimizing the Q-function with respect to the shrinkage hyperparameters and noise variance. Here we adopt a coordinate-wise optimization approach that involves iteratively updating each parameter individually while holding all other parameters fixed. Specifically:

$$\begin{aligned}
\hat{\boldsymbol{\lambda}}^{(t+1)} &= \underset{\boldsymbol{\lambda}}{\arg\min} \left\{ Q\left(\boldsymbol{\lambda} \,|\, \hat{\boldsymbol{\lambda}}^{(t)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right) \right\}, \\
\hat{\boldsymbol{\delta}}^{(t+1)} &= \underset{\boldsymbol{\delta}}{\arg\min} \left\{ Q\left(\boldsymbol{\delta} \,|\, \hat{\boldsymbol{\lambda}}^{(t+1)}, \hat{\boldsymbol{\delta}}^{(t)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right) \right\}, \\
\{\hat{\tau}^{(t+1)}, \hat{\sigma}^{2^{(t+1)}}\} &= \underset{\tau, \sigma^2}{\arg\min} \left\{ Q\left(\tau, \sigma^2 \,|\, \hat{\boldsymbol{\lambda}}^{(t+1)}, \hat{\boldsymbol{\delta}}^{(t+1)}, \hat{\tau}^{(t)}, \hat{\sigma}^{2^{(t)}} \right) \right\}.
\end{aligned} \tag{5.21}$$

The EM-steps outlined here are essentially identical to those introduced in Chapter 3.3. The key difference lies in the introduction of an additional G group shrinkage parameters for estimation. While the optimization problem in the M-step has become slightly more complex, it remains tractable. The computation of conditional expectations, such as $\mathbb{E}\left[\boldsymbol{\beta}^{\mathbf{2}}\right]$ and $\mathbb{E}\left[||\mathbf{y}-\mathbf{X}\boldsymbol{\beta}||^2\right]$, follows the same procedure as discussed in Section 3.4.

In this section, we present the M-step updates for two models: Sparse Horseshoe grouped regression and Sparse Lasso grouped regression. The former corresponds to the Bayesian grouped regression hierarchy (5.19), with a beta prime prior applied to both $\boldsymbol{\lambda}^2$ and $\boldsymbol{\delta}^2$, while the latter involves placing exponential priors instead. A closely related study to this subsection is the work by Xu et al. [185], for which they extend the decoupled shrinkage and selection method for grouped variables to produce sparse estimates using the posterior samples of the parameter estimates. However, this method, though effective, may be significantly slower than our proposed EM methods for sparse grouped regression estimates. Due to time constraints, only the update formulae are presented here. A detailed empirical comparison of the predictive performance and time complexity of these grouped EM procedures is of a primary focus for future work.

5.4.1 Sparse Grouped Horseshoe Regression

The group horseshoe regression model can be written as the Bayesian global-local shrinkage grouped hierarchy 5.19 with a beta prime prior on both the local-shrinkage parameter $\boldsymbol{\lambda}$ and the grouped shrinkage parameter $\boldsymbol{\delta}$:

$$\begin{aligned}\lambda_{gj}^2 &\sim \mathrm{B}'(1/2, 1/2),\\ \delta_g^2 &\sim \mathrm{B}'(1/2, 1/2),\\ \tau &\sim C^+(0,1).\end{aligned} \tag{5.22}$$

Substituting the negative logarithm of these prior distributions into (5.20) – (5.21) the following M-step updates:

$$\hat{\lambda}_{gj}^2 = \underset{\lambda_{gj}^2}{\operatorname{argmin}} \left\{ \log \lambda_{gj}^2 + \frac{W_{gj}}{\lambda_{gj}^2} + \log(1+\lambda_{gj}^2) \right\} = \frac{1}{4}\left(\sqrt{1+6W_{gj}+W_{gj}^2} + W_{gj} - 1\right)$$

$$\hat{\delta}_g^2 = \underset{\delta_g^2}{\operatorname{argmin}} \left\{ \frac{p_g+1}{2}\log \delta_g^2 + \frac{w_g}{\delta_g^2} + \log(1+\delta_g^2) \right\} = \frac{\sqrt{(1+p_g-2w_g)^2 + 8w_g(3+p_g)} + 2w_g - p_g - 1}{2(3+p_g)}$$

$$(\hat{\tau}^2, \hat{\sigma}^2) = \underset{\tau^2,\sigma^2}{\operatorname{argmin}}\, Q(\tau^2, \sigma^2; \hat{\boldsymbol{\lambda}}^2, \hat{\boldsymbol{\delta}}^2)$$

where $W_{gj} = E[\beta_{gj}^2]/(2\sigma^2\tau^2\delta_g^2)$ and $w_g = \sum_{j=1}^{p_g} E[\beta_{gj}^2]/(2\sigma^2\tau^2\lambda_{gj}^2)$. Given the $\hat{\boldsymbol{\lambda}}^2$ and $\hat{\boldsymbol{\delta}}^2$ updates, the $\hat{\tau}$ and $\hat{\sigma}^2$ estimates can be found using numerical optimisation. This two-dimensional optimization problem can be further reduced to a one-dimensional optimization problem by (approximately) estimating σ^2 using the expected residual sum-of-squares as discussed in (5.8).

For a group size of one (i.e., $G = 1$), the proposed grouped horseshoe prior (5.22) does not reduce to the standard horseshoe prior but simplifies to the horseshoe + prior. Consequently, it is not a direct generalization of the horseshoe prior; rather, it can be seen as an extension of the horseshoe + prior.

5.4.2 Sparse Grouped Lasso Regression

The group Lasso regression model can be written as the Bayesian global-local shrinkage grouped hierarchy 5.19 with an exponential prior on both the local-shrinkage parameter $\boldsymbol{\lambda}$ and the grouped shrinkage parameter $\boldsymbol{\delta}$:

$$\begin{aligned} \lambda_{gj}^2 &\sim \text{Exp}(2), \\ \delta_g^2 &\sim \text{Exp}(2), \\ \tau^2 &\sim \text{IG}(1,1). \end{aligned} \tag{5.23}$$

Substituting the negative logarithm of these prior distributions into (5.20) – (5.21) the following M-step updates:

$$\begin{aligned} \hat{\lambda}_{gj}^2 &= \underset{\lambda_{gj}^2}{\text{argmin}} \left\{ \frac{\log \lambda_{gj}^2}{2} + \frac{W_{gj}}{\lambda_{gj}^2} + \frac{\lambda_{gj}^2}{2} \right\} = \frac{1}{2}\left(\sqrt{1+8W_{gj}} - 1\right) \\ \hat{\delta}_g^2 &= \underset{\delta_g^2}{\text{argmin}} \left\{ \frac{p_g \log \delta_g^2}{2} + \frac{w_g}{\delta_g^2} + \frac{\delta_g^2}{2} \right\} = \frac{1}{2}\left(\sqrt{p_g^2 + 8w_g} - p_g\right) \\ (\hat{\tau}^2, \hat{\sigma}^2) &= \underset{\tau^2,\sigma^2}{\text{argmin}}\, Q(\tau^2, \sigma^2; \hat{\boldsymbol{\lambda}}^2, \hat{\boldsymbol{\delta}}^2) \end{aligned}$$

where $W_{gj} = E[\beta_{gj}^2]/(2\sigma^2\tau^2\delta_g^2)$ and $w_g = \sum_{j=1}^{p_g} E[\beta_{gj}^2]/(2\sigma^2\tau^2\lambda_{gj}^2)$.

For a group size of one (i.e., $G = 1$), the specific marginal prior density resulting from the proposed grouped Lasso prior (5.23) remains unknown. Preliminary findings suggest a potential link to the Bessel function or a prior distribution with a similar mathematical construct, but further research is required.

5.5 Conclusion

In this chapter, we present two sparse linear regression estimators via the proposed EM algorithm using the horseshoe prior and the Laplace prior. We offer an alternative sparse Bayesian Lasso estimator with the posterior mode as a point estimate and a novel sparse Bayesian horseshoe estimator, that is (as far as the authors are aware) the first estimator to explore the exact posterior mode estimates under the horseshoe prior. Our results demonstrate that both the proposed sparse Bayesian lasso and sparse Bayesian horseshoe estimators perform comparably to their corresponding state-of-the-art competitors in terms of predictive accuracy and computational efficiency.

Furthermore, we demonstrate that the EM algorithm is invariant to the reparameterization of the local shrinkage parameters. Maximizing for different parameterizations of the shrinkage parameter leads to posterior mode estimates with different behaviors. Specifically, we consider three parameterizations: λ, λ^2, and $\log \lambda$. In the normal means model with the Laplace prior, maximizing for λ^2 produces regression estimates resembling the Bayesian lasso posterior mode, while maximizing for λ or $\log \lambda$ yields non-sparse estimates resembling the Bayesian lasso posterior mean. As for the horseshoe prior, maximizing for all three different parameterizations gives sparse estimates with λ^2 producing the sparsest estimates.

The proposed sparse Bayesian Lasso estimator and the sparse Bayesian horseshoe estimator can be easily extended to handle grouped predictors. We present the EM updates for these grouped estimators in Section 5.4 and defer empirical experiments to future research due to time constraints. In this upcoming study, we will evaluate the performance of these proposed sparse grouped estimators in terms of computational efficiency, predictive accuracy, and variable selection within and across grouped predictors. We will compare the performance to existing sparse grouped regression methods [184, 186].

Chapter 6

Adaptive (Group) Horseshoe Regression

Horseshoe regression is a well-known Bayesian method for estimating sparse linear models. However, when most of the predictors are expected to have non-zero effects on the outcome, horseshoe regression may fall short in terms of predictive performance when compared to dense models such as ridge regression that do not induce sparsity. Therefore, it is crucial, though challenging to determine whether a problem is best modeled *a priori* as sparse or dense. In this chapter, we explore the adaptive normal-beta prime (NBP) prior. This prior has tunable hyperparameters that shape the behavior of its probability density function. Varying the hyperparameters allows the prior to induce a wide range of sparsity levels, from a strong emphasis on sparse solutions to a behavior similar to ridge regression that favors dense models. We extend the adaptive NBP prior to grouped regression model. Specifically, we extend the grouped half-Cauchy hierarchy of Xu et al. [185] by assigning the NBP prior to both the local and group shrinkage parameters, allowing for varying sparsity levels within and across group predictors. We introduce an efficient Metropolis-Hasting based sampler to estimate the NBP prior hyperparameters and empirical experiments with simulated data consistently show the strong performance of the proposed adaptive generalized horseshoe estimator across regression problems with varying sparsity level and signal-to-noise ratios.

6.1 Introduction

Group structures are common in regression analysis. They can appear in the form of categorical predictors represented by groups of dummy variables or in the context of additive modeling, where each predictor can be expressed as a set of basis functions forming a group. In practical applications such as gene expression analysis and financial market modeling, groupings exist naturally in the data. For instance, genes that influence similar traits form groups in gene expression data, while stocks from the same sector form groups in financial data. In these scenarios, group shrinkage plays an important role: when there is insufficient evidence to suggest the significance of predictors within a group, the entire group of predictors is shrunk towards zero. This reduces the noise from individual "spurious predictors" (which tend to appear more frequently in high-dimensional settings), and decreases model complexity, thereby reducing the risk of overfitting.

Within the Bayesian framework, there has been extensive research focusing on the application of continuous shrinkage priors for linear regression problems involving group predictor variables. Traditional approaches, such as the Bayesian group lasso[106, 184], the Bayesian group bridge [114], and the group horseshoe [185] primarily apply shrinkage at the group level and do not consider within-group shrinkage. Given a data matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ with G groups of predictors, Xu et al. [185] introduced a hierarchical Bayesian grouped model for the corresponding target variable $\mathbf{y} \in \mathbb{R}^n$ that integrates group-level local shrinkage parameters along with the global and individual local shrinkage parameters:

$$\begin{aligned} \mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2 &\sim N_n\left(\mathbf{X}\boldsymbol{\beta},\ \sigma^2 \boldsymbol{I}_n\right) \\ \sigma^2 &\sim \sigma^{-2} d\sigma^2 \end{aligned} \tag{6.1}$$

$$\begin{aligned} \beta_{jg} | \tau^2, \lambda_{jg}^2, \delta_g^2, \sigma^2 &\sim N\left(0,\ \tau^2 \lambda_{jg}^2 \delta_g^2 \sigma^2\right) \\ \lambda_j^2 &\sim \pi(\lambda_j^2) d\lambda_j^2 \\ \delta_g^2 &\sim \pi(\delta_g^2) d\delta_g^2 \\ \tau &\sim \pi(\tau) d\tau. \end{aligned} \tag{6.2}$$

where $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_G) \in \mathbb{R}^G$ are the regression coefficients, δ_g is the local shrinkage parameter for the g^{th} group, controlling the shrinkage at group level, λ represents the local shrinkage parameter that controls individual shrinkage, while τ is the global shrinkage parameter that controls the overall shrinkage. This hierarchical structure allows for simultaneous control of shrinkage between and within groups. In their approach, a half-Cauchy prior is used to independently model the local shrinkage parameter at both the individual and group levels. For a group size of one (i.e. $G = p$), this prior configuration simplifies to the horseshoe+ prior, thus we refer to it as the group horseshoe+ prior. Boss et al. [37] further built on this framework by proposing the group inverse-gamma gamma (GIGG) prior. This work utilizes a similar hierarchical structure but assigns inverse-gamma and gamma distributions to the individual and group local shrinkage parameters, respectively. This setup results in a normal beta prime (NBP) shrinkage prior - a generalization of the group horseshoe prior - on the regression coefficient marginally, introducing hyperparameters that control the sparsity level and allowing adaptive shrinkage to be learned from the data. However, this model has limitations: (i) the decomposition of the beta prime prior restricts its applicability to overlapping group structures, and (ii) the proposed Gibbs sampler, which involves sampling from the generalized inverse Gaussian (GIG) distribution, can be computationally inefficient.

In this chapter, we present an efficient alternative for group regression using the NBP prior within the hierarchical Bayesian grouped model proposed by Xu et al. [185]. We assign a beta prime prior for both the individual and group local shrinkage parameters. Unlike the method by Boss et al. [37], which ensures the product of the local shrinkage priors is a beta prime, we demonstrate that this condition is not necessary. It is not crucial for the marginal shrinkage prior on the beta coefficient to take a specific form, such as the normal beta prime. Instead, by simply assigning the beta prime prior to both the local shrinkage parameters, we can achieve similar results. We emphasize that controlling the tail behavior of the priors by adjusting the hyperparameters is more important than restricting the exact marginal form of the shrinkage prior. This concept enables the straightforward extension of adaptive group shrinkage regression to multiple and overlapping groups for the first time (contrary to what was previously mentioned as a limitation in [37]). Additionally, we examine the log-scale interpretation of the local shrinkage parameters under the beta prime prior. This logarithmic transformation offers a clear understanding of the interaction between individual and group-level shrinkage, serving as a guide for selecting appropriate shrinkage priors. It reveals how different prior choices affect the correlation between shrinkage parameters and how the hyperparameters influence this relationship.

In summary, our main contributions are:

- Introduction of a more efficient Bayesian estimator for the normal beta prime (NBP) model in the linear regression contexts, both with and without grouping structures. Notably, the proposed framework introduces adaptivity in group shrinkage regression such that straightforward extension to multiple and overlapping group structures is allowed, although examples of overlapping groups were not explicitly demonstrated in this chapter.

- Proposed an adapted version of the gradient-assisted Metropolis-Hastings algorithm by Schmidt and Makalic [155] to estimate the hyperparameters in the NBP model. We demonstrate that this procedure can be easily integrated into Gibbs samplers for fitting Bayesian regression models.

- We show that our method outperforms existing adaptive beta prime estimators in terms of computational speed and predictive performance, across linear regression problems with and without group structures.

- Investigation of the log-scale interpretation of local shrinkage parameters under the beta prime prior, providing insights into the interaction between individual and group-level shrinkage.

The remainder of this chapter is organized as follows. Section 6.2 introduces the NBP prior and its established theoretical properties. We then present the log-scale interpretation of the shrinkage hyperparameters (Sec. 6.2.2) and compare the NBP prior to other well-known priors, specifically the Laplace and Student-t priors (Sec. 6.2.3). Section 6.3 details the grouped NBP model and Section 6.4 outlines the sampling algorithm required for posterior estimation. Finally, Section 6.5 presents an empirical comparison of the proposed and existing NBP estimators for linear regression problems with and without grouped predictors.

6.2 Normal Beta Prime Prior

The normal beta prime (NBP) prior [19] has been referred to by various names in the literature, such as the three-parameter beta (TPB) prior [14], the adaptive normal hypergeometric inverted beta prior [188], the inverse-gamma gamma prior [37], and the generalized horseshoe prior [155]. Despite these different names, they all describe the same underlying prior model, which involves placing a beta prime distribution over the local shrinkage parameter, λ:

$$\pi(\lambda_j|a, b) = \frac{2\lambda_j^{2a-1}(1+\lambda_j^2)^{-a-b}}{\mathrm{B}(a, b)},\ a > 0, b > 0. \tag{6.3}$$

where $\mathrm{B}(a, b)$ denotes the beta function. The hyperparameter a controls the sparsity level of β_j (i.e. smaller a values yield a marginal prior for β_j that concentrates more around $\beta_j = 0$); while the hyperparameter b can be seen as the tail-decay parameter (i.e. smaller values indicate a slower rate of decay at the tails of the marginal distribution) [155]. By varying a and b, many different prior distributions can be achieved, such as the Strawderman–Berger prior ($a = 0.5, b = 1$), the horseshoe prior ($a = 0.5, b = 0.5$), and normal-exponential-gamma (NEG) prior ($a = 1, b > 0$) [14].

6.2.1 Theoretical Properties

The theoretical studies discussed in this section primarily build upon the normal means problem whereby $\mathrm{y}_j|\beta_j \sim N(\beta_j, 1)$. Here y is a p-dimensional mean vector and β_j is the estimated mean associated with observation y_j. This is a special case of the linear regression model described in (6.1), with $\mathbf{X} = \mathbf{I}$, $p = n$, and $\sigma^2 = 1$. The normal means model serves as an important test case for the theoretical understanding of many shrinkage methods [15, 26, 31, 42, 44, 149, 170, 171] due to its' simple and tractable framework. In this context, we specifically examine and compare the shrinkage properties under sparsity and tail robustness of the NBP prior with the well-known horseshoe prior.

TABLE 6.1: Prior densities for λ_j and $\xi_j = \log(\lambda_j)$ associated with several popular shrinkage rules. Densities are given up to a constant.

Prior for β_j	Density for λ_j	Density for ξ_j
Lasso	$2\lambda_j e^{-\lambda_j^2}$	$2e^{-e^{2\xi_j}+2\xi_j}$
Student-t	$\frac{b^a}{\Gamma(a)} 2\lambda_j^{(-2a-1)} e^{\frac{b}{\lambda_j^2}}$	$\frac{b^a}{\Gamma(a)} 2e^{-2\xi_j a - \frac{b}{e^{2\xi_j}}}$
Horseshoe	$\frac{2}{\pi(1+\lambda_j^2)}$	$\frac{2e^{\xi_j}}{\pi(1+e^{2\xi_j})}$
Generalized Horseshoe	$\frac{2\lambda_j^{2a-1}(1+\lambda_j^2)^{-a-b}}{\mathrm{B}(a,b)}$	$\frac{2\sigma(2\xi_j)^a \sigma(-2\xi_j)^b}{\mathrm{Beta}(a,b)}$

Efficiency for sparsity When the true parameter β_0 is zero, the horseshoe prior attains Kullback-Leibler super-efficiency. This implies that the estimated density using the horseshoe prior converges to the true density faster than the maximum likelihood estimator (MLE) under the assumption that the true mean vector is sparse. The horseshoe prior achieves this super-efficiency due to its asymptote at 0, which efficiently shrinks small coefficients towards zero. In the case of the beta prime prior, Bai and Ghosh [18] demonstrated that the induced marginal prior $\pi(\beta_j|\sigma^2, \tau = 1, a, b)$ exhibits a pole at 0 if and only if $0 < a \leq 1/2$. The strength of this pole increases as $a \to 0$, indicating a stronger concentration of the prior around $\beta_j = 0$. And when $a > 1/2$, the pole disappears, resulting in a non-sparse prior. Consequently, Yu et al. [188] proved that the Bayes estimator under the NBP prior achieves super-efficiency at the origin with the Kullback-Leibler risk bound being smaller than that of the horseshoe prior when $0 < a < 1/2$.

Tail robustness Another appealing property that the NBP prior shares with the horseshoe prior is tail robustness. Boss et al. [37] established that for any combination of a and b values, the marginal beta prime prior has tails that decay at a polynomial rate, indicating heavy-tailed behavior. This heavy tail property of the prior contributes to its robustness in handling large signals or extreme observations. Furthermore, Theorem 3.4 in Yu et al. [188] provides additional evidence of the asymptotic tail robustness of the NBP prior. The theorem specifically focuses on the behavior of the posterior mean estimate under the NBP prior and demonstrates that as $|y| \to \infty$, the influence of the prior on the estimate diminishes. In other words, the NBP estimator is asymptotically unbiased when the signal is large.

6.2.2 Log-scale interpretation

A novel and insightful understanding of the properties of the NBP prior can be achieved by expressing the beta prime prior on the local shrinkage parameter in log-scale, denoted as $\boldsymbol{\xi} = \log(\boldsymbol{\lambda})$.

The probability density function for the transformed local-shrinkage variable, ξ_j is then given by:

$$\pi(\xi_j|a,b) = \frac{2\sigma(2\xi_j)^a\sigma(-2\xi_j)^b}{\mathrm{B}(a,b)}. \tag{6.4}$$

where $\sigma(x) = 1/(1+e^{-x})$ is the standard logistic function. This density is known in the literature as the Type IV generalized logistic distribution. Table 6.1 lists the prior densities for λ_j and ξ_j implied by the different prior choices included in Figure 6.2 and 6.1. The concept of exploring the log-scale interpretation of shrinkage priors was first introduced by Schmidt and Makalic [154]. It offers an intuitive understanding of the tail behavior and concentration properties of the prior distributions.

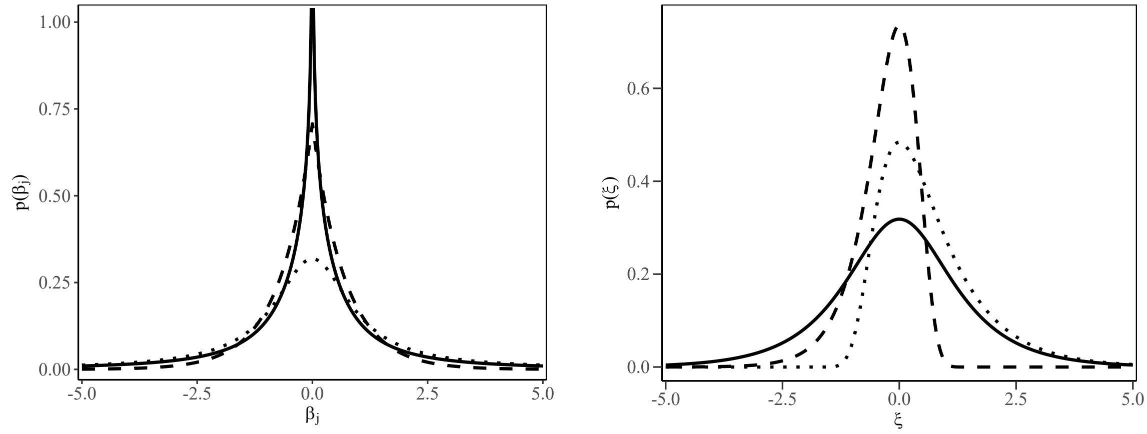
The roles of the hyperparameters a and b for the NBP prior can be better reflected by reparameterizing them in terms of the mean (μ) and sample size (ν). Specifically, we can establish a relationship between a and b with μ and ν as $a = \mu\nu$ and $b = (1-\mu)\nu$ respectively. Figure 6.2 depicts the beta prime prior in log scale with the μ and ν parameterization. Here, μ controls the skewness of the distribution: when $\mu < 0.5$, the distribution skews left (more probability placed on smaller ξ_j), favoring sparsity and concentrating around $\beta_j = 0$; Conversely, $\mu > 0.5$ results in a right-skewed distribution, allowing less sparsity and bias towards larger ξ_j, thus avoiding over-shrinkage/underestimation of large effects; when $\mu = 0.5$, the distribution is symmetric, suggesting no preference over sparsity or non-sparsity. On the other hand, ν is the concentration parameter that controls the level of probability concentrated around μ, with smaller values allowing for more variation. As $\nu \to \infty$, the prior converges to the ridge prior, with minimal variation in ξ and a point mass at μ. Consequently, by appropriately choosing a and b, it becomes possible to approximate the behavior of many well-known priors across a wide range of sparsity levels, from those inducing sparsity to those promoting non-sparsity.

6.2.3 Relation to other shrinkage priors

Here we compare the characteristics and tail behavior of the NBP prior to popular shrinkage priors namely the Laplace prior and the Student-t prior using the log scale parameterization. The Laplace prior, which corresponds to the Bayesian Lasso model, is obtained by placing an exponential prior on λ^2 with a rate parameter of one[1], i.e., $\lambda_j^2 \sim \mathrm{Exp}(1)$; While, the Student-t priors with a degree of freedom γ can be achieved using inverse-gamma mixing, where $\lambda_j^2 \sim \mathrm{IG}(\gamma/2, \gamma/2)$.

The log-scale parameterization provides a clearer visualization of why the Bayesian Lasso introduces bias in estimation when the underlying model coefficients are large, in contrast to the Horseshoe prior. In Figure 6.1a, when observing the marginal prior over β, both the Horseshoe and Laplace priors appear symmetric, making the explanation less apparent. However, in Figure

[1] In previous chapters, the Bayesian Lasso had a rate of two. Here, we use a rate of one to center the transformed prior around zero in log scale. This change doesn't affect the Lasso estimate; it's merely a scaling adjustment.



(A)

(B)

FIGURE 6.1: Comparison of the Lasso (dashed), Student-t$_{\gamma=1}$ (dotted), and horseshoe (solid) densities on the marginal prior over β_j (left) and transformed prior for the local shrinkage parameter with $\xi_j = \log \lambda_j$ (right).

6.1b, we can clearly see that the distribution over ξ for the Bayesian Lasso is asymmetric, with a heavier left-hand tail and a stronger preference for $\xi_j < 0$. This asymmetry reveals that the Bayesian Lasso is biased towards small shrinkage values (λ close to zero), indicating a bias towards zero coefficients.

Similarly, the Student-t prior also exhibits asymmetry in the distribution over ξ, but unlike the Laplace prior, the Student-t prior has a heavier right-hand tail. This preference for large coefficients and having a lower probability for small coefficients is evident from the heavy tails and the absence of a peak at the origin in the distribution plot over β_j. On the other hand, the horseshoe prior places equal preference on both large and small coefficients, as its distribution over ξ is symmetric.

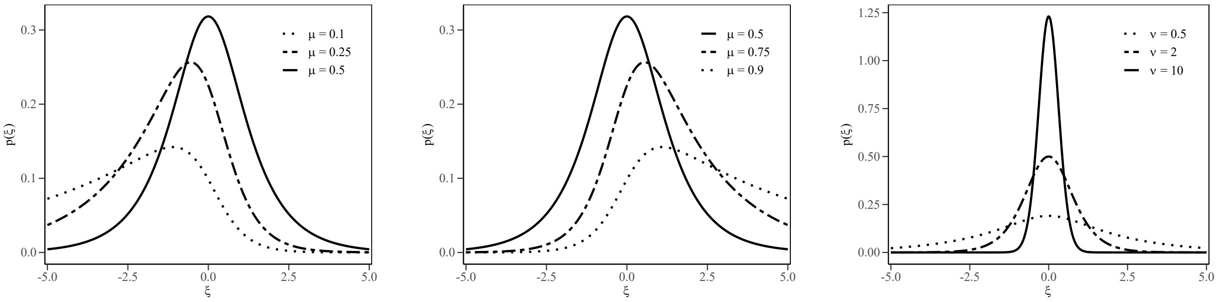
It is important to note that the Bayesian Lasso is not a special case of the NBP prior, and approximation between the two is challenging due to their distinct tail behaviors. The Lasso (exponential) prior vanishes at a log-super-linear rate in the right tail (i.e. exponentially decaying tails in the original parameterization) :

$$\lim_{\xi \to -\infty} \frac{-\log p_{\mathrm{L}}(\xi)}{\xi} = -2 \qquad \text{and} \qquad \lim_{\xi \to \infty} \frac{-\log p_{\mathrm{L}}(\xi)}{\xi} = \infty \tag{6.5}$$

The beta prime prior, on the other hand, has log-linear tails:

$$\lim_{\xi \to -\infty} \frac{-\log p_{\mathrm{G}}(\xi|a,b)}{\xi} = -2a \qquad \text{and} \qquad \lim_{\xi \to \infty} \frac{-\log p_{\mathrm{G}}(\xi|a,b)}{\xi} = 2b. \tag{6.6}$$

This suggests that the beta prime prior can only generalize to priors with heavy tails (log-linear decaying tails).



(A) $\nu = 1$, $\mu \leq 0.5$ (B) $\nu = 1$, $\mu \geq 0.5$ (C) $\mu = 0.5$

FIGURE 6.2: The behavior of the NBP prior density when varying (A)(B) μ and (C) ν in $\xi = \log \lambda$ space.

Let $x^2|\nu, b \sim \text{IG}(b, 1/\nu)$ and $\nu|a \sim \text{IG}(a, 1)$, then $p(x) \propto x^{2a-1}(1+x^2)^{-a-b}$ [155] whereby marginally x follows (6.3). Using this decomposition, we can express the NBP prior as a scale mixture representation:

$$\begin{aligned} \beta_j|\tau^2, \lambda_j^2, \sigma^2 &\sim N\left(0,\ \tau^2\lambda_j^2\sigma^2\right), \\ \lambda_j^2|\nu_j &\sim \text{IG}(b, 1/\nu_j), \\ \nu_{g1}, \cdots \nu_p &\sim \text{IG}(a, 1). \end{aligned} \tag{6.7}$$

This hierarchy can be viewed as a Student-t prior with an additional IG hyperprior. Upon integrating out the local shrinkage parameter $\boldsymbol{\lambda}$, it results in a Student-t prior distribution over the regression coefficient β_j of the form

$$\begin{aligned} p(\beta_j|\tau^2, \sigma^2, \nu, b) &= \int p(\beta_j|\tau^2, \sigma^2, \boldsymbol{\lambda}^2)p(\boldsymbol{\lambda}^2|\nu, b)\, d\boldsymbol{\lambda}^2 \\ &= \frac{\sqrt{\nu_j}\,\Gamma(b+\frac{1}{2})}{\sqrt{2\pi\sigma^2\tau^2}\,\Gamma(b)} \left(1 + \frac{\beta_j^2\nu_j}{2\sigma^2\tau^2}\right)^{-\left(b+\frac{1}{2}\right)} \end{aligned} \tag{6.8}$$

Consequently, the NBP model can be seen as the product of independent Student-t distributions over β_j.

6.3 Grouped Normal Beta Prime (GNBP) model

Using the hierarchical Bayesian grouped regression model (6.1) - (6.2), we assign the beta prime prior to both the group and local shrinkage parameters. We assume no prior knowledge on the degree of shrinkage of the regression coefficients and assign the recommended half-cauchy prior for

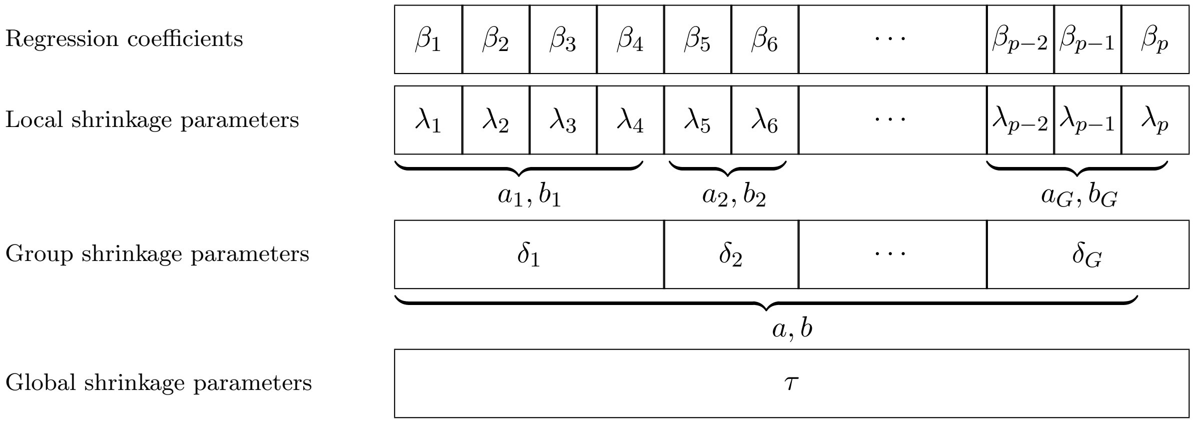

Regression coefficients
β_1 β_2 β_3 β_4 β_5 β_6 $\cdots$ β_{p-2} β_{p-1} β_p
Local shrinkage parameters
λ_1 λ_2 λ_3 λ_4 λ_5 λ_6 $\cdots$ λ_{p-2} λ_{p-1} λ_p
a_1, b_1
a_2, b_2
a_G, b_G
Group shrinkage parameters
δ_1 δ_2 $\cdots$ δ_G
a, b
Global shrinkage parameters
τ

Figure 6.3: An illustration of how the shrinkage parameters interact with each other. Each regression coefficient is associated with a local shrinkage parameter λ that controls individual shrinkage, while the global shrinkage parameter τ controls the overall shrinkage. Within each group of local shrinkage parameters, the additional group shrinkage parameter δ further amplifies or diminishes the shrinkage of λ values within that specific group. Each group of $\boldsymbol{\lambda}$ is also characterized by its own unique hyperparameters, denoted as a_g and b_g, which control the adaptivity and level of sparsity of the λs within group g. The additional pair of hyperparameters a and b controls the adaptivity of $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_G)$.

the global variance parameter [66, 141]:

$$\begin{aligned} \lambda_{gj}^2 &\sim B'(a_g, b_g) \\ \delta_g^2 &\sim B'(a, b) \\ \tau &\sim C^+(0,1). \end{aligned} \tag{6.9}$$

By selecting appropriate values for a_g and b_g, we can control the level of adaptivity within each individual group, while a and b govern the adaptivity of the overall group shrinkage parameter. The introduction of the group shrinkage parameter δ introduces apriori correlation among the local shrinkage parameters within the same group. Proposition 6.1 provides the exact formula for computing this correlation. It reveals that a higher variance in the sequence λ_g corresponds to a lower correlation, indicating that δ has a lesser impact on the local shrinkage parameter within group g. Conversely, a higher variance in δ results in a higher correlation, indicating that δ exerts a greater influence on the values of λ within the same group. With reference to Figure 6.3, this implies that if, for instance, group 1 contains more zero coefficients compared to group 2, we can enhance the shrinkage of the λ parameters in group 1 towards zero by adjusting the values of δ_1 without affecting the estimates in group 2.

Proposition 6.1. *The correlation between the log-transformed product of the group shrinkage parameter δ_g and the local shrinkage parameters λ_g within the same group, g, can be expressed as the ratio of the variance of $\log \delta_g$ to the sum of the variance of $\log \delta_g$ and $\log \lambda_g$:*

$$\operatorname{corr}(\log(\delta_g \lambda_{gi}), \log(\delta_g \lambda_{gj})) = \frac{\mathbb{V}\left[\log(\delta)\right]}{\mathbb{V}\left[\log(\delta)\right] + \mathbb{V}\left[\log(\lambda_g)\right]} \tag{6.10}$$

where $(\lambda_{gi}, \lambda_{gj}) \in \boldsymbol{\lambda}_g$.

Proof. Taking the logarithm of the product of two variables is equivalent to adding their logarithms. Therefore, we can write $\log(\delta_g \lambda_{gj}) = \log(\delta_g) + \log(\lambda_{gj})$. Let $d \equiv \log \delta$, $l_1 \equiv \log \lambda_{g1}$ and $l_2 \equiv \log \lambda_{g2}$, the correlation between the shrinkage factors within the same group is

$$\text{corr}(l_1 + d, l_2 + d) = \frac{\mathbb{V}\left[d\right]}{\sqrt{(\mathbb{V}\left[d\right] + \mathbb{V}\left[l_1\right])(\mathbb{V}\left[d\right] + \mathbb{V}\left[l_2\right])}} \tag{6.11}$$

Since l_1 and l_2 belong to the same group, they share the same hyperparameters, hence $\mathbb{V}\left[l_1\right] = \mathbb{V}\left[l_2\right] = \mathbb{V}\left[\log(\lambda_g)\right]$ and we have

$$\text{corr}(l_1 + d, l_2 + d) = \frac{\mathbb{V}\left[d\right]}{\mathbb{V}\left[d\right] + \mathbb{V}\left[\log(\lambda_g)\right]} \tag{6.12}$$

□

The variance of the local and group shrinkage parameters in logarithmic scales depends on the tail behavior of the prior distribution of these parameters in the original space. A lighter tail in the original space leads to lower variance on the log scale and conversely, a heavier tail leads to higher variance. Specifically, the hyperparameters a and b in the beta prime priors (6.9) control the tail behavior of both the local and global shrinkage parameters, λ and δ, and therefore, also control the extent of the grouping effect. Proposition 6.1 suggests that when the variance of $\log \delta$ is large relative to the variance of $\log \lambda$, the grouping effect becomes the dominant factor in the hyperparameter estimation (i.e., the correlation of the shrinkage parameters within the group is highest and the effect of δ becomes more significant). Conversely, if the variance of $\log \delta$ is small relative to the variance of $\log \lambda$ then the individual local shrinkage parameters will exhibit greater a priori variability. Therefore, by adjusting the beta prime tail behavior, we can control the *a priori* correlation among the shrinkage parameters. A significantly larger variance in $\log \delta$ (relative to the variance in $\log \lambda$) suggests strong grouping effect, whereas a smaller variance in $\log \delta$ (relative to the variance in $\log \lambda$) implies weaker impact.

6.4 Posterior Sampling

By adopting the scale mixture representation of the beta prime distribution in (6.7), we present the revised grouped NBP hierarchy as follows

$$\begin{aligned}
\mathbf{y}|\boldsymbol{X},\boldsymbol{\beta},\sigma^2 &\sim N_n\left(\boldsymbol{X}\boldsymbol{\beta},\ \sigma^2\boldsymbol{I}_n\right)\\
\sigma^2 &\sim \sigma^{-2}d\sigma^2\\
\beta_{gj}|\tau^2,\delta_g^2,\lambda_{gj}^2,\sigma^2 &\sim N\left(0,\ \tau^2\lambda_{gj}^2\delta_g^2\sigma^2\right)\\
\lambda_{gj}^2|\nu_{gj} \sim \mathrm{IG}(b_g,1/\nu_{gj}) \quad &, \quad \nu_{g1},\cdots\nu_{gp} \sim \mathrm{IG}(a_g,1)\\
\delta_g^2|\zeta_g \sim \mathrm{IG}(b,1/\zeta_g) \quad &, \quad \zeta_1,\cdots\zeta_G \sim \mathrm{IG}(a,1)\\
\tau &\sim C^+(0,1).
\end{aligned} \tag{6.13}$$

Using this hierarchy, we develop a Gibbs sampler that iteratively samples from the following full conditional densities:

$$\begin{aligned}
\boldsymbol{\beta}|\cdot &\sim N_p(\mathbf{A}^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y},\sigma^2\mathbf{A}^{-1})\\
\sigma^2|\cdot &\sim \mathrm{IG}\left(\frac{n+p}{2},\frac{||\mathbf{y}-\mathbf{X}\boldsymbol{\beta}||^2+\boldsymbol{\beta}^{\mathrm{T}}(\tau^2\boldsymbol{\Lambda})^{-1}\boldsymbol{\beta}}{2}\right)\\
\lambda_{gj}^2|\cdot \sim \mathrm{IG}\left(\frac{1}{2}+b_g,\ \frac{1}{\nu_{gj}}+\frac{\beta_{gj}^2}{2\tau^2\sigma^2}\right) \quad &, \quad \nu_{gj}|\cdot \sim \mathrm{IG}\left(a_g+b_g,\ \ 1+\frac{1}{\lambda_{gj}^2}\right)\\
\delta_g^2|\cdot \sim \mathrm{IG}\left(\frac{ng}{2}+b,\ \frac{1}{\zeta_g}+\sum_{g=1}^{ng}\frac{\beta_g^2}{2\tau^2\sigma^2\lambda_g^2}\right) \quad &, \quad \zeta_g|\cdot \sim \mathrm{IG}\left(a+b,\ \ 1+\frac{1}{\delta_g^2}\right)
\end{aligned} \tag{6.14}$$

where ng is the number of predictors in group g, $\mathbf{A} = \mathbf{X}^{\mathrm{T}}\mathbf{X} + (\tau^2\boldsymbol{\Lambda})^{-1}$ and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1^2,\cdots,\lambda_p^2)$. In the absence of a group structure, this Gibbs sampler can be readily transformed into a normal linear regression with a beta prime prior by setting $g = 1$ and not sampling the δ and ζ parameters.

6.4.1 Estimating the adaptive parameters a and b

To achieve adaptability in the generalized horseshoe prior for different levels of sparsity and signal sizes, we propose estimating the hyperparameters a and b using the mGrad [155] algorithm. The mGrad algorithm is a gradient-based sampling method that utilizes the first-order Taylor series expansion of the likelihood to construct a Metropolis-Hastings proposal density that is both likelihood and prior informed. In this section, we provide the necessary components to integrate the mGrad algorithm into our Gibbs sampling framework.

We assign hyper-priors to a and b such that they follow a good default non-informative prior, the half-Cauchy distribution:

$$a_1 \cdots a_G, b_1 \cdots b_G, a, b \;\sim\; C^+(0,1). \tag{6.15}$$

Given that the application of the mGrad algorithm is the same for all $G+1$ hyperparameters, we focus on the case of a single pair of hyperparameters, assuming the same application to each group. Thus, the hierarchical structure of interest becomes:

$$\begin{aligned} \boldsymbol{\kappa}|a,b \;&\sim\; \text{Beta}(a,b), & (6.16)\\ a \;&\sim\; C^+(0,1), & (6.17)\\ b \;&\sim\; C^+(0,1). & (6.18) \end{aligned}$$

Note that $a, b \in (0,\infty)$. Implementation of the mGrad algorithm is straightforward if the parameters range between $(-\infty,\infty)$, therefore, we apply a log transformation on both a and b such that:

$$m = \log a, \quad \phi = \log b \tag{6.19}$$

which yields the following transformed priors

$$p(m) = \frac{2e^m}{\pi(e^{2m}+1)}, \quad p(\phi) = \frac{2e^\phi}{\pi(e^{2\phi}+1)}. \tag{6.20}$$

These transformed priors exhibit unimodal, symmetric bell-shaped distributions with finite variances; in fact, $\mathbb{V}[m] = \mathbb{V}[\phi] = \frac{\pi^2}{4} \approx 3$.

The remaining component required for the mGrad algorithm are the negative log priors $-\log p(m)p(\phi)$, and the gradients of the negative log-likelihood $-\log p(\boldsymbol{\kappa}|m,\phi)$ with respect to both m and ϕ. The negative log priors can be presented straightforwardly as follows:

$$-\log p(m)p(\phi) \;=\; -m + \log(e^{2m}+1) - \phi + \log(e^{2\phi}+1). \tag{6.21}$$

To compute the gradients of $-\log p(\boldsymbol{\kappa}|m,\phi)$, we start with the negative log-likelihood of (6.16):

$$-\log p(\boldsymbol{\kappa}|a,b) \;=\; (1-a)S_1 + (1-b)S_2 + n\log \mathrm{B}(a,b) \tag{6.22}$$

where

$$S_1 = \sum_{j=1}^{p} \log \kappa_j, \quad S_2 = \sum_{j=1}^{p} \log(1-\kappa_j) \tag{6.23}$$

are the two sufficient statistics of the beta distribution. The gradients of the negative log-likelihood with respect to a and b are:

$$-\frac{\partial \log p(\boldsymbol{\kappa}|a,b)}{\partial a} = -S_1 + n[\psi(a) - \psi(a+b)], \tag{6.24}$$

$$-\frac{\partial \log p(\boldsymbol{\kappa}|a,b)}{\partial b} = -S_2 + n[\psi(b) - \psi(a+b)], \tag{6.25}$$

where $\psi(\cdot)$ is the digamma function. The gradients with respect to the transformed parameters m and ϕ will then just be (6.24) and (6.25) multiplied by their respective Jacobian term for the parameter change:

$$-\frac{\partial \log p(\boldsymbol{\kappa}|m,\phi)}{\partial m} = \left(-\frac{\partial \log p(\boldsymbol{\kappa}|a(m),b(\phi))}{\partial a}\right) e^m, \tag{6.26}$$

$$-\frac{\partial \log p(\boldsymbol{\kappa}|m,\phi)}{\partial \phi} = \left(-\frac{\partial \log p(\boldsymbol{\kappa}|a(m),b(\phi))}{\partial b}\right) e^\phi \tag{6.27}$$

where $a(m) = e^m$ and $b(\phi) = e^\phi$.

The mGrad algorithm utilizes Equations (6.21), (6.26), and (6.27) to generate proposals for the hyperparameters. By employing the recommended step size and acceptance conditions described in [155], we observe acceptance rates ranging from 50% to 70% for all the experiments in Section 6.5.

We also explored estimating the hyperparameters in terms of the μ and ν parameterization. However, we found that estimating either μ and ν or a and b did not yield a significant difference in predictive performance. Although one parameterization may outperform the other in certain scenarios, overall, we found that a and b provided slightly better performance. Therefore, we decided to present only the results for the a and b parameterization in Section 6.5. The reason behind this observation remains unclear and could be explored in future work.

6.4.2 Estimating the global shrinkage parameter τ

We estimate the global shrinkage parameter τ by sampling from the conditional posterior distribution with a standard half-Cauchy prior $C^+(0,1)$ placed on τ. We adopt the inverse gamma scale mixture representation of the half-Cauchy prior [110] such that:

$$\begin{aligned} \tau^2 \,|\, \omega &\sim \mathrm{IG}(1/2, 1/\omega), \\ \omega &\sim \mathrm{IG}(1/2, 1), \end{aligned} \tag{6.28}$$

and sampling τ simply requires sampling from the following conditional posterior distribution:

$$\begin{aligned} \tau^2 \,|\, \cdot \; &\sim \; \text{IG}\left(\frac{p+1}{2}, \frac{1}{\omega} + \frac{1}{2\sigma^2}\sum_{j=1}^{p}\frac{\beta_j^2}{\lambda_j^2}\right), \\ \omega \,|\, \cdot \; &\sim \; \text{IG}\left(1, 1 + \frac{1}{\tau^2}\right). \end{aligned} \tag{6.29}$$

The posterior mean estimates of τ are computed as the average of the τ samples drawn from this conditional posterior distribution.

6.5 Experimental Results

We performed a comparative evaluation of the proposed NBP estimator (P-NBP) against existing NBP estimators for linear regression and the GIGG model for grouped linear regression. All experiments were conducted using the R statistical platform. For the NBP model, we used the `nbp` function from the `NormalBetaPrime` R package, and for the GIGG model, we used the `gigg` R package. Unless specified otherwise, all function arguments were set to their default values. Without any loss of generality, all predictors and the target variable are standardized to have means zero and unit variance.

In terms of prior configuration, GIGG deviates slightly from our approach by adopting a different setting where the product of the group shrinkage parameter δ and the local shrinkage parameter λ follows a beta prime distribution. This is achieved through the scale mixture representation of the beta prime distribution, for which given two random variables x and s such that

$$x|\theta \;\sim\; \text{G}(a, z) \quad \text{and} \quad \theta \;\sim\; \text{IG}(b, z); \tag{6.30}$$

then $x|\theta \sim B'(a, b)$. Using this decomposition, we can directly compare the prior employed by GIGG (left) and our own (right):

$$\begin{aligned} \lambda_{gj}^2 \;&\sim\; \text{IG}(b_g, 1) \qquad & \lambda_{gj}^2 \;&\sim\; B'(a_g, b_g) \\ \delta_g^2 \;&\sim\; \text{G}(a_g, 1) \qquad & \delta_g^2 \;&\sim\; B'(a, b) \end{aligned} \tag{6.31}$$

In Boss et al. [37], they established $a_g = 1/n$ and learn b_g through marginal maximum likelihood estimation (MMLE). This implies a certain alignment with our approach, suggesting a link between our group shrinkage hyperparameter's a and their a_g, given that their a_g controls the degree of shrinkage in δ. Consequently, as demonstrated in Table 6.4, when conducting the group regression experiment to compare against GIGG, we provide two distinct configurations for our method. The first configuration involves setting $a = 1/n$ and only allowing the estimation of b. In the

Table 6.2: Details of the experimental settings. # A.Var represents the number of active variables, Coef is the true β coefficients assigned to each group and $\mathrm{U}(v)$ denotes uniform sampling from the vector v, where the size corresponds to the number of active variables within each group.

	$p = 50, n = 500$				$p = 100, n = 300$		$p = 80, n = 200$	
	Concentrated		Distributed		Dense		Half	
Group	# A.Var / Group Size	Coef	# A.Var / Group Size	Coef	# A.Var / Group Size	Coef	# A.Var / Group Size	Coef
1	1/10	0.5	10/10	0.5	17/30	3	22/25	0.8
2	1/10	1	0/10	-	8/10	$\mathrm{U}([2, 3, 4])$	0/10	-
3	1/10	1.5	0/10	-	16/10	0.5	3/10	2.5
4	1/10	2	0/10	-	3/10	$[9.5, 8, 7]$	8/10	1.5
5	1/10	2	0/10	-	14/15	1.5	0/5	-
6					18/20	0.5	4/15	$[1, 2, 3, 5]$
7							1/5	2

second scenario, both a and b can be estimated using the mgrad algorithm. In both cases, the hyperparameters for λ^2, (a_g, b_g) are subject to estimation.

We present four distinct yet related experimental settings, as summarized in Table 6.2. The first two experiments are adapted from Boss et al. [37]. In the first setting, the signal is concentrated in one of the regressors within each of the five groups, while in the second setting, the signal is more evenly distributed across all regressors but limited to only the first group. To further explore different scenarios, we introduce two additional variations: the third experiment involves dense signals (more than half of the regressors with signals) in all six groups, and the fourth experiment features a combination of dense signals in half of the groups and few to no signal regressors in the other half. Pairwise correlations within each group are set to 0.8 and the pairwise correlation across groups is 0.2. In all simulation settings, the residual error variance σ^2 is selected to achieve varying signal-to-noise ratios, $\mathrm{SNR} \in (0.2, 1, 5)$ with $\sigma^2 = \boldsymbol{\beta}^{\mathrm{T}}\Sigma\boldsymbol{\beta}(1 - \mathrm{SNR})/\mathrm{SNR}$. Following Boss et al. [37], we evaluate the estimation properties based on the empirical mean-squared error(MSE) stratified by null and non-null coefficients across 100 replicates. Specifically, we calculate the $\widehat{\mathrm{MSE}}$ as the average squared difference between the estimated $\hat{\boldsymbol{\beta}}$ and the true $\boldsymbol{\beta}$ for each of the 100 simulated datasets. The true coefficients for each of the experimental settings are given in Table 6.2. The results for both standard linear regression and grouped linear regression are presented in Table 6.3 and Table 6.4 respectively. The experimental settings for both scenarios remain consistent, except that in standard linear regression, grouping information is not passed into the estimator during model fitting.

Standard Linear Regression In this experiment, we compare the P-NBP estimator against the NBP estimator implemented in the `nbp` R function. For each estimator, 10,000 samples were

Table 6.3: Mean-squared errors (MSE) for the standard linear regression problem with no grouping information provided during model fitting. Z0 is the MSE for estimating the null coefficients, NZ0 is the MSE for estimating the null coefficients, and OA = Z0 + NZ0 is the overall MSE. Cells in bold highlight the estimator with the lowest overall MSE, one for each distinct signal-to-noise ratio (SNR) in the problem. The results for three estimators are presented - the ordinary least squares estimator (OLS), the normal-beta prime estimator using the `nbp` R function (NBP), and the proposed NBP estimator (P-NBP).

		Concentrated				Distributed				Half				Dense			
SNR	Method	Z0	NZ0	OA	Time	Z0	NZ0	OA	Time	Z0	NZ0	OA	Time	Z0	NZ0	OA	Time
	OLS	212.8	25.5	238.3	0.01	467.2	112.9	580.2	0.01	35466	33141	68608	0.01	91740	285864	377604	0.01
0.2	NBP	18.92	8.92	27.84	32.6	432.9	105.1	538.0	32.4	34878	32445	67323	62.6	93328	260839	354167	104
	P-NBP	2.79	8.81	**11.6**	8.82	5.96	5.61	**11.6**	8.81	88.7	99.9	**188.6**	10.4	163.2	765.5	**928.8**	13.6
	OLS	8.52	1.02	9.53	0.01	18.69	4.52	23.21	0.01	1418	1325	2744	0.01	3669	11434	15104	0.01
1	NBP	1.06	1.13	2.19	32.1	3.13	3.58	6.71	32.2	1070	1000	2070	67.3	3561	11113	14675	99.7
	P-NBP	0.89	1.26	**2.16**	8.83	1.65	3.08	**4.73**	8.85	44.1	74.3	**118.4**	10.5	182.7	501.3	**683.9**	13.1
	OLS	0.34	0.04	0.38	0.01	0.75	0.18	0.93	0.01	56.75	53.03	109.8	0.01	146.8	457.4	604.2	0.01
5	NBP	0.04	0.03	0.07	32.1	0.12	0.20	0.32	35.2	12.15	25.12	37.27	66.1	41.84	174.5	216.3	95.1
	P-NBP	0.02	0.02	**0.04**	8.89	0.09	0.22	**0.31**	9.64	12.81	22.95	**35.76**	10.3	42.14	153.2	**195.3**	12.4

drawn from the corresponding posterior distribution with a burnin period of 10,000 samples. Given the absence of a thinning option in the `nbp` R function, and to ensure fair time complexity comparisons, we set the thinning level of the P-NBP estimator to be 1 (i.e. no thinning). The results in Table 6.3 suggest that the NBP estimator consistently performs worse than the P-NBP estimator, particularly when the SNR is low. Its performance deteriorates further when the true regression coefficients are not sparse, offering only marginal improvements in mean squared error (MSE) over the least squares method. Nevertheless, in settings with high SNR, the NBP estimator becomes comparable to the P-NBP estimator, although P-NBP maintains a slight advantage in terms of prediction accuracy. Overall, the P-NBP estimator consistently provides superior prediction accuracy when compared to NBP, with the additional advantage of being up to six times faster.

Grouped Linear Regression In this experiment, we compare the proposed grouped NBP (GNBP) estimator against the GIGG estimator from the `gigg` R package. For each estimator, 10,000 samples were drawn from the corresponding posterior distribution with a burnin period of 10,000 samples and a thinning level of 5. The results presented in Table 6.4 suggest that with increasing signal-to-noise ratio (SNR), the predictive performance of all estimators (excluding OLS) tends to level out. However, in scenarios with low SNR, the proposed GNBP estimator consistently demonstrates superior overall MSE performance. When SNR is set to 1, there are instances where GIGG outperforms our approach. Nonetheless, we observed that by aligning certain settings with their implementation, particularly by setting a_δ to $1/n$, GNBP's performance aligns more closely with GIGG. From a broader perspective, the average predictive performance of both the GIGG estimator and the proposed GNBP estimator appears comparable. However,

Table 6.4: Mean-squared errors (MSE) for the grouped linear regression problem. Z0 is the MSE for estimating the null coefficients, NZ0 is the MSE for estimating the null coefficients, and OA = Z0 + NZ0 is the overall MSE. Cells in bold highlight the estimator with the lowest overall MSE, one for each distinct signal-to-noise ratio (SNR) in the problem. OLS records the results for the ordinary least squares estimator; GIGG records the results using the `gigg` R function, and the remaining two methods record the results of the proposed GNBP estimator under two different settings: the first with a fixed group level hyperparameter $a_\delta = 1/n$, and only b_δ is learned, and the second setting where both a_δ, b_δ are estimated.

		Concentrated			Distributed			Half			Dense		
SNR	Method	Z0	NZ0	OA	Z0	NZ0	OA	Z0	NZ0	OA	Z0	NZ0	OA
0.2	OLS	212.8	25.5	238.3	467.2	112.9	580.2	35466	33141	68608	91740	285864	377604
	GIGG($a_g = 1/n, b_g =?$)	4.26	9.37	13.63	2.32	11.59	13.91	152.1	136.2	288.3	459.4	1198	1657
	(a_g, b_g)												
	$a_\delta = 1/n$, $b_\delta =?$	3.45	9.57	13.02	1.26	9.83	**11.09**	105.6	120.6	226.2	390.1	1046	1435
	$a_\delta =?$, $b_\delta =?$	2.38	10.38	**12.76**	0.02	13.01	13.03	32.6	101.3	**133.9**	212.7	807.1	**1019**
1	OLS	8.52	1.02	9.53	18.69	4.52	23.21	1418	1325	2744	3669	11434	15104
	GIGG($a_g = 1/n, b_g =?$)	0.67	1.13	**1.79**	0.12	2.43	**2.55**	42.80	114.5	157.3	229.2	580.6	809.8
	(a_g, b_g)												
	$a_\delta = 1/n$, $b_\delta =?$	0.80	1.27	2.07	0.07	2.61	2.68	38.97	100.0	**139.0**	205.2	516.4	721.6
	$a_\delta =?$, $b_\delta =?$	0.66	1.24	1.90	0.02	4.97	4.99	35.39	109.1	144.5	209.4	511.9	**721.3**
5	OLS	0.34	0.04	0.38	0.75	0.18	0.93	56.7	53.1	109.8	146.8	457.4	604.2
	GIGG($a_g = 1/n, b_g =?$)	0.02	0.02	0.03	0.01	0.16	0.17	9.07	23.35	32.41	43.87	121.0	164.9
	(a_g, b_g)												
	$a_\delta = 1/n$, $b_\delta =?$	0.03	0.02	0.05	0.01	0.16	0.17	8.94	23.48	32.42	39.85	126.8	166.6
	$a_\delta =?$, $b_\delta =?$	0.01	0.01	**0.02**	0.01	0.16	0.17	8.68	23.61	**32.28**	39.49	125.2	**164.7**

the results do suggest that the GNBP estimator has an advantage over the GIGG particularly when estimating coefficient vectors with lower levels of sparsity.

6.6 Discussion

In this chapter, we make two main contributions. Firstly, in standard linear regression (i.e. the group structure information is absent), experimental results strongly suggest that the P-NBP estimator outperforms existing adaptive normal-beta prime estimators, specifically the one implemented within the NBP package [19] in R. This superiority is observed both in terms of time complexity and predictive performance. Secondly, in the context of grouped regression, the proposed GNBP estimator offers a straightforward conceptual framework while upholding predictive accuracy. Notably, in scenarios involving coefficients of lower sparsity, our approach appears to be highly competitive with state-of-the-art GIGG [37] prior. This suggests that introducing adaptability into the shrinkage parameters by assigning an adaptive prior is the key to achieving good performance in grouped regression. We believe that any prior exhibiting behavior similar to the NBP prior can serve this purpose.

To elaborate further, the GIGG hierarchy presented by Boss et al. [37] is a cleverly designed hierarchical structure in which the regression coefficients marginally follow a horseshoe prior when there is only one group level (i.e., $G = 1$). Their work demonstrates the superior performance of this hierarchy compared to grouped half-Cauchy hierarchy presented in Xu et al. [185]. However, we believe that this comparison may not be entirely fair, as Xu et al. [185] does not incorporate adaptivity in their shrinkage parameters. In this chapter, we demonstrate that once adaptivity is introduced in both the local and grouped shrinkage parameters by assigning the beta prime prior to both, the conceptually simple hierarchy presented by Xu et al. [185] can perform as effectively as the GIGG estimator. Additionally, the proposed GNBP estimator has the potential to effectively handle multiple and overlapping group structures. It is not immediately clear how one would extend the GIGG hierarchy to handle this setting.

Chapter 7

Concluding Remarks

This thesis explores various global-local shrinkage priors in the context of Bayesian penalized linear regression; specifically, two main research contributions are made: an EM procedure for posterior mode estimation (Chap. 4 - 5) and an MCMC sampler for estimating hyperparameters of the normal-beta prime prior with a specific focus on application to grouped predictors (Chap. 6). These methods are adaptable and can be applied to a wide range of global-local shrinkage priors with minimal modifications. Chapter 4 explores the application of the EM algorithm to Bayesian ridge regression, followed by Chapter 5 that discusses the application of the EM algorithm to sparsity-inducing priors, specifically the horseshoe and Laplace priors. Having examined the application of the EM algorithm to both non-sparse ridge priors and (two) sparsity-inducing priors, Chapter 6 introduces the adaptive normal beta prime prior and an MCMC sampler for hyperparameter estimation, enabling the model to adapt and learn the problem's sparsity level directly from the data. This approach eliminates the need for manual (sparse) prior selection, allowing the data itself to dictate the appropriate level of sparsity. A more detailed summary of the main contributions of this thesis is as follows:

1. Developed a novel expectation-maximization (EM) procedure to solve for the *exact* posterior mode of Gaussian linear regression models that is applicable to a wide range of priors, including those with no closed-from density. This procedure was also extended to generalized linear models.

2. Applied the proposed EM procedure to Bayesian ridge regression, and improved the computational efficiency using the singular value decomposition (SVD) representation of the predictor matrix **X** for faster EM updates. This results in a novel method for tuning the regularization parameter of ridge regression that is faster than leave-one-out cross-validation (LOOCV). Importantly, this proposed ridge EM method yields ridge estimates of the regression parameters of equal, or particularly in the setting of sparse covariates, superior quality to those obtained by minimizing the LOOCV risk. As a supplementary outcome, a

faster implementation of the LOOCV ridge regression risk is presented using the same SVD technique, outperforming existing implementations, including the popular `scikit-learn` library, by approximately a factor of two.

3. Provided a finite sample bound to guarantee the unimodality of the posterior distribution within the Bayesian ridge regression model. This guarantees the convergence of iterative posterior optimization procedures like the proposed EM algorithm to a unique optimal solution, for a large enough sample size, under relatively mild conditions.

4. Applied the proposed EM procedure to sparsity inducing prior, namely the horseshoe and Laplace prior, resulting in the novel introduction of the Bayesian sparse horseshoe estimator and the Bayesian sparse lasso estimator respectively. Our findings demonstrated the strong performance of these proposed estimators when compared to state-of-the-art competitors in terms of predictive accuracy and computational efficiency. This application is also extended to grouped regression.

5. Demonstrated that different parameterizations of the shrinkage parameter when applying the EM algorithm result in posterior mode estimates with different behaviors. For instance, in the normal means model with the Laplace prior, maximizing for λ^2 leads to regression estimates resembling the Bayesian lasso posterior mode, while maximizing for λ or $\log \lambda$ yields estimates resembling the Bayesian lasso posterior mean.

6. Derived an efficient MCMC sampler for linear regression with the normal beta-prime (NBP) prior. This sampler estimates the hyperparameters for automatic adaptivity to the sparsity of the problem and offers a promising framework for potential extension to accommodate beta regression models.

7. Extended the NBP model to handle group structure, in addition to providing an alternative and concise interpretation of the interaction between the local and grouped shrinkage parameters using log-scale representation of the shrinkage prior. Empirical experiments consistently demonstrate the strong performance of the proposed adaptive regression method across regression problems with varying sparsity levels and signal-to-noise ratios.

7.1 Limitations

While this thesis has made significant contributions, there are several limitations that need to be acknowledged:

1. An important open problem is the theoretical analysis of the expected number of EM iterations that are required for convergence. While the empirical evidence in Chapter 4 and

our intuition based on the convergence of the posterior to a multivariate normal distribution suggest that the number of iterations converges to a constant, a rigorous worst-case analysis is necessary.

2. The proposed stochastic EM procedure, while promising, is not yet well-established and is sensitive to the smoothing factor. It also did not perform well in some empirical experiments in Chapter 5. Improving this procedure is important, as it could potentially be extended to non-linear models such as neural networks.

3. The efficiency of the proposed Metropolis-Hasting sampler could potentially be further improved in terms of effective sample size per second. One possible enhancement is to consider the utilization of second-order information to aid in the convergence process.

7.2 Future Work

This section outlines the potential future research directions that can be used to extend the work presented in this thesis.

1. **Enhance scalability and speed:** Improve the proposed procedures to handle larger datasets and enhance computational speed. Specifically, a form of coordinate-wise-descent could potentially allow application of the sparse horseshoe EM algorithm to very large problems.

2. **Handling non-Gaussian regression models:** Explore alternative methods for handling non-Gaussian linear regression models. This includes conducting thorough testing of the iteratively reweighted least squares algorithm in various regression scenarios. While it has demonstrated promise in logistic regression, there is potential for exploring its applicability in other contexts, such as Poisson regression, gamma regression, and survival analysis.

3. **Investigation of sparse group EM estimators.** In Chapter 5.4, we presented update formulae for applying our EM procedure to sparse group estimation. A detailed empirical comparison of these sparse group estimators against state-of-the-art sparse group procedures, as well as the MCMC methods developed in Chapter 6 is important future work.

4. **Extension to non-linear models:** A promising avenue for future research involves extending the EM procedure to accommodate non-linear models such as trend filtering and additive models.

Appendix A

Supplementary Results Material

Table A.1 provides the details about the real datasets used in the experiments in Section 4.6.2, Section 5.1.2.2 and Section 5.2.2. This includes the source of the data and a brief description of the target variables for each dataset.

Table A.1: Real datasets details

Datasets	Abbreviation	n	p	Target Variable	Source
Buzz in social media (Twitter)	Twitter	583250	77	mean number of active discussion	UCI
Blog Feedback	Blog	60021	281	number of comments in the next 24 hours	UCI
Relative location of CT slices on axial axis	CT slices	53500	386	reference: Relative image location on axial axis	UCI
Buzz in social media (Tom's Hardware)	TomsHw	28179	97	Mean Number of display	UCI
Condition-based maintenance of naval propulsion plants	NPD - com	11934	16	GT Compressor decay state coefficient	UCI
Condition-based maintenance of naval propulsion plants	NPD - tur	11934	16	GT Turbine decay state coefficient	UCI
Parkinson's Telemonitoring	PT - motor	5875	26	motor UPDRS score	UCI
Parkinson's Telemonitoring	PT - total	5875	26	total UPDRS score	UCI
Abalone	Abalone	4177	8	Rings (age in years)	UCI
Communities and Crime	Crime	1994	128	ViolentCrimesPerPop	UCI
Airfoil Self-Noise	Airfoil	1503	6	Scaled sound pressure level (decibels)	UCI
Student Performance	Student	649	33	final grade (with G1 & G2 removed)	UCI
Concrete Compressive Strength	Concrete	1030	9	Concrete compressive strength (MPa)	UCI
Forest Fires	F.Fires	517	13	forest burned area (in ha)	UCI
Boston Housing	B.Housing	506	13	Median value of owner-occupied homes in $1000's	[80]
Facebook metrics	Facebook	500	19	Total Interactions (with comment, like, and share columns removed)	UCI
Diabetes	Diabetes	442	10	quantitative measure of disease progression one year after baseline	[58]
Real estate valuation	R.Estate	414	7	house price of unit area	UCI
Auto mpg	A.MPG	398	8	city-cycle fuel consumption in miles per gallon	UCI
Yacht hydrodynamics	Yacht	308	7	residuary resistance per unit weight of displacement	UCI
Automobile	A.mobile	205	26	price	UCI
Rat eye tissues	Eye	120	200	the expression level of TRIM32 gene	[152]
Prostate	Prostate	97	8	Log PSA	[160]
Riboflavin	Ribo	71	4088	Log-transformed riboflavin production rate	[40]
Crop	Crop	24000	3072	24 crop classes	UCR
Electric Devices	ElecD	16637	4096	7 electric devices	UCR
StarLight Curves	StarL	9236	7168	3 starlight curves	UCR

Bibliography

[1] Salaheddine El Adlouni, Garba Salaou, and André St-Hilaire. Regularized Bayesian quantile regression. *Communications in Statistics-Simulation and Computation*, 47(1):277–293, 2018.

[2] Mohammad Ahsanullah, BM Golam Kibria, and Mohammad Shakil. *Normal and student's t distributions and their applications*, volume 4. Springer, 2014.

[3] Hirotugu Akaike. A new look at the statistical model identification. In *Selected Papers of Hirotugu Akaike*, pages 215–222. Springer, 1974.

[4] Carlos M Alaíz, Alvaro Barbero, and José R Dorronsoro. Group fused lasso. In *Artificial Neural Networks and Machine Learning–ICANN 2013: 23rd International Conference on Artificial Neural Networks Sofia, Bulgaria, September 10-13, 2013. Proceedings 23*, pages 66–73. Springer, 2013.

[5] Rahim Alhamzawi and Haithem Taha Mohammad Ali. The Bayesian adaptive lasso regression. *Mathematical biosciences*, 303:75–82, 2018.

[6] Rahim Alhamzawi and Haithem Taha Mohammad Ali. The Bayesian elastic net regression. *Communications in Statistics-Simulation and Computation*, 47(4):1168–1178, 2018.

[7] Rahim Alhamzawi and Haithem Taha Mohammad Ali. A new gibbs sampler for Bayesian lasso. *Communications in Statistics-Simulation and Computation*, 49(7):1855–1871, 2020.

[8] Mahdi Alkhamisi, Ghadban Khalaf, and Ghazi Shukur. Some modifications for choosing ridge parameters. *Communications in Statistics-Theory and Methods*, 35(11):2005–2020, 2006.

[9] David M Allen. The relationship between variable selection and data agumentation and a method for prediction. *technometrics*, 16(1):125–127, 1974.

[10] David F Andrews and Colin L Mallows. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(1):99–102, 1974.

[11] Davide Anguita, Luca Ghelardoni, Alessandro Ghio, Luca Oneto, Sandro Ridella, et al. The'k'in k-fold cross validation. In *ESANN*, pages 441–446, 2012.

[12] Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. 2010.

[13] A Armagan, D Dunson, and J Lee. Bayesian generalized double pareto shrinkage. *Biometrika*, 2010.

[14] Artin Armagan, Merlise Clyde, and David Dunson. Generalized beta mixtures of gaussians. *Advances in neural information processing systems*, 24, 2011.

[15] Artin Armagan, David B Dunson, and Jaeyong Lee. Generalized double pareto shrinkage. *Statistica Sinica*, 23(1):119, 2013.

[16] Arthur Asuncion and David Newman. UCI machine learning repository, 2007.

[17] Ray Bai and Malay Ghosh. The inverse gamma-gamma prior for optimal posterior contraction and multiple hypothesis testing. *arXiv preprint arXiv:1710.04369*, 2017.

[18] Ray Bai and Malay Ghosh. Large-scale multiple hypothesis testing with the normal-beta prime prior. *Statistics*, 53(6):1210–1233, 2019.

[19] Ray Bai and Malay Ghosh. On the beta prime prior for scale parameters in high-dimensional Bayesian regression models. *Statistica Sinica*, 31(2):843–865, 2021.

[20] V Balakrishnan. All about the dirac delta function (?). *Resonance*, 8(8):48–58, 2003.

[21] David Barber and Wim Wiegerinck. Tractable variational structures for approximating graphical models. *Advances in Neural Information Processing Systems*, 11, 1998.

[22] Ole Barndorff-Nielsen, John Kent, and Michael Sørensen. Normal variance-mean mixtures and z distributions. *International Statistical Review/Revue Internationale de Statistique*, pages 145–159, 1982.

[23] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

[24] J Bernardo, J Berger, APAFMS Dawid, A Smith, et al. Regression and classification using gaussian process priors. *Bayesian statistics*, 6:475, 1998.

[25] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, and Brandon Willard. Default Bayesian analysis with global-local shrinkage priors. *Biometrika*, 103(4):955–969, 2016.

[26] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, Brandon Willard, et al. The horseshoe+ estimator of ultra-sparse signals. *Bayesian Analysis*, 12(4):1105–1131, 2017.

[27] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, Brandon Willard, et al. Lasso meets horseshoe: A survey. *Statistical Science*, 34(3):405–427, 2019.

[28] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, and Brandon T Willard. The horseshoe-like regularization for feature subset selection. *Sankhya B*, pages 1–30, 2019.

[29] Anindya Bhadra, Jyotishka Datta, Yunfan Li, and Nicholas Polson. Horseshoe regularisation for machine learning in complex and deep models 1. *International Statistical Review*, 88(2): 302–320, 2020.

[30] Satish Bhat and Vidya Raju. A class of generalized ridge estimators. *Communications in Statistics - Simulation and Computation*, 46(7):5105–5112, 2017. doi: 10.1080/03610918.2016.1144765. URL https://doi.org/10.1080/03610918.2016.1144765.

[31] Anirban Bhattacharya, Debdeep Pati, Natesh S Pillai, and David B Dunson. Dirichlet–laplace priors for optimal shrinkage. *Journal of the American Statistical Association*, 110 (512):1479–1490, 2015.

[32] Anirban Bhattacharya, Antik Chakraborty, and Bani K Mallick. Fast sampling with gaussian scale mixture priors in high-dimensional regression. *Biometrika*, page asw042, 2016.

[33] Christopher Bishop, Neil Lawrence, Tommi Jaakkola, and Michael Jordan. Approximating posterior distributions in belief networks using mixtures. *Advances in neural information processing systems*, 10, 1997.

[34] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[35] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[36] Dominique Bontemps. Bernstein–von Mises theorems for Gaussian regression with increasing number of regressors. 2011.

[37] Jonathan Boss, Jyotishka Datta, Xin Wang, Sung Kyun Park, Jian Kang, and Bhramar Mukherjee. Group inverse-gamma gamma shrinkage for sparse linear models with block-correlated regressors. *Bayesian Analysis*, 1(1):1–30, 2023.

[38] Leo Breiman. Heuristics of instability and stabilization in model selection. *The annals of statistics*, 24(6):2350–2383, 1996.

[39] Philip J Brown and Jim E Griffin. Inference with normal-gamma prior distributions in regression problems. *Bayesian analysis*, 5(1):171–188, 2010.

[40] Peter Bühlmann, Markus Kalisch, and Lukas Meier. High-dimensional statistics with a view toward applications in biology. *Annual Review of Statistics and Its Application*, 1:255–278, 2014.

[41] Carlos M Carvalho, Nicholas G Polson, and James G Scott. Handling sparsity via the horseshoe. In *Artificial Intelligence and Statistics*, pages 73–80, 2009.

[42] Carlos M Carvalho, Nicholas G Polson, and James G Scott. The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480, 2010.

[43] George Casella, Malay Ghosh, Jeff Gill, and Minjung Kyung. Penalized regression, standard errors, and Bayesian lassos. *Bayesian analysis*, 5(2):369–411, 2010.

[44] Ismaël Castillo and Romain Mismer. Empirical Bayes analysis of spike and slab posterior distributions. 2018.

[45] Michael Celentano, Andrea Montanari, and Yuting Wei. The lasso with general gaussian designs with applications to hypothesis testing. *arXiv preprint arXiv:2007.13716*, 2020.

[46] Gilles Celeux. The SEM algorithm: a probabilistic teacher algorithm derived from the em algorithm for the mixture problem. *Comput Stat quarterly*, 2:73–82, 1985.

[47] S Grace Chang, Bin Yu, and Martin Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE transactions on image processing*, 9(9):1532–1546, 2000.

[48] Ray-Bing Chen, Chi-Hsiang Chu, Shinsheng Yuan, and Ying Nian Wu. Bayesian sparse group selection. *Journal of Computational and Graphical Statistics*, 25(3):665–683, 2016.

[49] Simon L Cotter, Gareth O Roberts, Andrew M Stuart, and David White. Mcmc methods for functions: modifying old algorithms to make them faster. 2013.

[50] Jyotishka Datta and Jayanta K Ghosh. Asymptotic properties of Bayes risk for the horseshoe prior. 2013.

[51] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The UCR time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305, 2019.

[52] Angus Dempster, François Petitjean, and Geoffrey I Webb. ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5):1454–1495, 2020.

[53] Angus Dempster, Daniel F Schmidt, and Geoffrey I Webb. HYDRA: Competing convolutional kernels for fast and accurate time series classification. *Data Mining and Knowledge Discovery*, 2023.

[54] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *J R Stat Soc Series B Stat Methodol*, 39(1):1–22, 1977.

[55] David L Donoho and Jain M Johnstone. Ideal spatial adaptation by wavelet shrinkage. *biometrika*, 81(3):425–455, 1994.

[56] Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

[57] Bradley Efron. Microarrays, empirical Bayes and the two-groups model. *Statistical science*, pages 1–22, 2008.

[58] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[59] AK Md Ehsanes Saleh and BM Golam Kibria. Performance of some new preliminary test ridge regression estimators and their properties. *Communications in Statistics-Theory and Methods*, 22(10):2747–2764, 1993.

[60] Frank Emmert-Streib and Matthias Dehmer. High-dimensional lasso-based computational regression models: regularization, shrinkage, and selection. *Machine Learning and Knowledge Extraction*, 1(1):359–383, 2019.

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

[62] Mário AT Figueiredo. Adaptive sparseness for supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 25(9):1150–1159, 2003.

[63] LLdiko E Frank and Jerome H Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

[64] Shuai Fu. Hierarchical Bayesian lasso for a negative binomial regression. *Journal of Statistical Computation and Simulation*, 86(11):2182–2203, 2016.

[65] Alan E Gelfand and Adrian FM Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409, 1990.

[66] Andrew Gelman. Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper). *Bayesian analysis*, 1(3):515–534, 2006.

[67] Andrew Gelman, Daniel Lee, and Jiqiang Guo. Stan: A probabilistic programming language for Bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, 40(5):530–543, 2015.

[68] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984.

[69] Subhashis Ghosal. Asymptotic normality of posterior distributions in high-dimensional linear models. *Bernoulli*, pages 315–331, 1999.

[70] Subhashis Ghosal, Jayanta K Ghosh, and Aad W Van Der Vaart. Convergence rates of posterior distributions. *Annals of Statistics*, pages 500–531, 2000.

[71] Mark Girolami. A variational method for learning sparse and overcomplete representations. *Neural computation*, 13(11):2517–2532, 2001.

[72] Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73 (2):123–214, 2011.

[73] Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[74] Izrail Solomonovich Gradshteyn and Iosif Moiseevich Ryzhik. *Table of integrals, series, and products*. Academic press, 2014.

[75] Jim E Griffin and Philip J Brown. Alternative prior distributions for variable selection with very many more variables than observations. 2005.

[76] Jim E Griffin and Philip J Brown. Bayesian hyper-lassos with non-convex penalization. *Australian & New Zealand Journal of Statistics*, 53(4):423–442, 2011.

[77] Ramadan Hamed, Ali EL Hefnawy, and Aya Farag. Selection of the ridge parameter using mathematical programming. *Communications in Statistics-Simulation and Computation*, 42 (6):1409–1432, 2013.

[78] Chris Hans. Bayesian lasso regression. *Biometrika*, 96(4):835–845, 2009.

[79] Richard J Hanson. A numerical method for solving Fredholm integral equations of the first kind using singular values. *SIAM Journal on Numerical Analysis*, 8(3):616–622, 1971.

[80] David Harrison Jr and Daniel L Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81–102, 1978.

[81] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC press, 2015.

[82] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *Annals of statistics*, 50(2):949, 2022.

[83] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[84] Ali El Hefnawy and Aya Farag. A combined nonlinear programming model and Kibria method for choosing ridge parameter regression. *Communications in Statistics-Simulation and Computation*, 43(6):1442–1470, 2014.

[85] John W Hilgers. On the equivalence of regularization and certain reproducing kernel Hilbert space approaches for solving first kind problems. *SIAM Journal on Numerical Analysis*, 13 (2):172–184, 1976.

[86] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[87] Matthew D Hoffman, Andrew Gelman, et al. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.

[88] Clive J Hoggart, John C Whittaker, Maria De Iorio, and David J Balding. Simultaneous analysis of all SNPs in genome-wide and re-sequencing association studies. *PLoS genetics*, 4 (7):e1000130, 2008.

[89] TC Hsiang. A Bayesian view on ridge regression. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 24(4):267–268, 1975.

[90] Daniel Hsu, Sham M. Kakade, and Tong Zhang. Random design analysis of ridge regression. In Shie Mannor, Nathan Srebro, and Robert C. Williamson, editors, *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23 of *Proceedings of Machine Learning Research*, pages 9.1–9.24, Edinburgh, Scotland, 25–27 Jun 2012. PMLR.

[91] Tommi S Jaakkola and Michael I Jordan. Improving the mean field approximation via the use of mixture distributions. In *Learning in graphical models*, pages 163–173. Springer, 1998.

[92] W JAMES. Estimation with quadratic loss. In *Proc. 4th Berkeley Symp. on Math. Statist. and Prob., 1961*, 1961.

[93] Lawless JF. A simulation study of ridge and other regression estimators. *Communications in Statistics-theory and Methods*, 5(4):307–323, 1976.

[94] James Johndrow, Paulo Orenstein, and Anirban Bhattacharya. Scalable approximate mcmc algorithms for the horseshoe prior. *Journal of Machine Learning Research*, 21(73), 2020.

[95] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37:183–233, 1999.

[96] Ghadban Khalaf and Ghazi Shukur. Choosing ridge parameter for regression problems. 2005.

[97] BM Golam Kibria. Performance of some new ridge regression estimators. *Communications in Statistics-Simulation and Computation*, 32(2):419–435, 2003.

[98] Harri T Kiiveri. A Bayesian approach to variable selection when the number of variables is very large. *Lecture Notes-Monograph Series*, pages 127–143, 2003.

[99] Dmitry Kobak, Jonathan Lomond, and Benoit Sanchez. The optimal ridge penalty for real-world high-dimensional data can be zero or negative due to the implicit ridge regularization. *The Journal of Machine Learning Research*, 21(1):6863–6878, 2020.

[100] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

[101] Dimitris Korobilis, Kenichi Shimizu, et al. Bayesian approaches to shrinkage and sparse estimation. *Foundations and Trends® in Econometrics*, 11(4):230–354, 2022.

[102] Ravi Kumar, Daniel Lokshtanov, Sergei Vassilvitskii, and Andrea Vattani. Near-optimal bounds for cross-validation via loss stability. In *International Conference on Machine Learning*, pages 27–35. PMLR, 2013.

[103] Chenlei Leng, Minh-Ngoc Tran, and David Nott. Bayesian adaptive lasso. *Annals of the Institute of Statistical Mathematics*, 66:221–244, 2014.

[104] Hanning Li and Debdeep Pati. Variable selection using shrinkage priors. *Computational Statistics & Data Analysis*, 107:107–119, 2017.

[105] Jiahan Li. *The Bayesian lasso, Bayesian SCAD and Bayesian group lasso with applications to genome-wide association studies.* The Pennsylvania State University, 2011.

[106] Jiahan Li, Zhong Wang, Runze Li, and Rongling Wu. Bayesian group lasso for nonparametric varying-coefficient models with application to functional genome-wide association studies. *The annals of applied statistics*, 9(2):640, 2015.

[107] Ker-Chau Li. Asymptotic optimality for cp, cl, cross-validation and generalized cross-validation: discrete index set. *The Annals of Statistics*, pages 958–975, 1987.

[108] Qing Li and Nan Lin. The Bayesian elastic net. 2010.

[109] D. V. Lindley and A. F. M. Smith. Bayes estimates for the linear model. *Journal of the Royal Statistical Society (Series B)*, 34(1):1–41, 1972.

[110] Enes Makalic and Daniel F Schmidt. A simple sampler for the horseshoe estimator. *IEEE Signal Processing Letters*, 23(1):179–182, 2015.

[111] Enes Makalic and Daniel F Schmidt. High-dimensional Bayesian regularised regression with the BayesReg package. *arXiv preprint arXiv:1611.06649*, 2016.

[112] Enes Makalic and Daniel F Schmidt. A simple sampler for the horseshoe estimator. *IEEE Signal Processing Letters*, 23(1):179–182, 2016.

[113] Himel Mallick and Nengjun Yi. A new Bayesian lasso. *Statistics and its interface*, 7(4):571–582, 2014.

[114] Himel Mallick and Nengjun Yi. Bayesian group bridge for bi-level variable selection. *Computational statistics & data analysis*, 110:115–133, 2017.

[115] Himel Mallick and Nengjun Yi. Bayesian bridge regression. *Journal of applied statistics*, 45 (6):988–1008, 2018.

[116] Colin L Mallows. Some comments on cp. *Technometrics*, 42(1):87–94, 2000.

[117] Gertraud Malsiner-Walli and Helga Wagner. Comparing spike and slab priors for Bayesian variable selection. *arXiv preprint arXiv:1812.07259*, 2018.

[118] Geoffrey McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[119] Rosa J Meijer and Jelle J Goeman. Efficient approximate k-fold and leave-one-out cross-validation for ridge regression. *Biometrical Journal*, 55(2):141–155, 2013.

[120] Nicolai Meinshausen. Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1): 374–393, 2007.

[121] Nicolai Meinshausen, Peter Bühlmann, et al. High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, 34(3):1436–1462, 2006.

[122] Xiao-Li Meng and Donald B Rubin. Maximum likelihood estimation via the ecm algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

[123] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[124] Thomas P Minka. Expectation propagation for approximate Bayesian inference. *arXiv preprint arXiv:1301.2294*, 2013.

[125] Léo Miolane and Andrea Montanari. The distribution of the lasso: Uniform control over sparse balls and adaptive parameter tuning. *The Annals of Statistics*, 49(4):2313–2335, 2021.

[126] Toby J Mitchell and John J Beauchamp. Bayesian variable selection in linear regression. *Journal of the american statistical association*, 83(404):1023–1032, 1988.

[127] Gisela Muniz and BM Golam Kibria. On some ridge regression estimators: An empirical comparisons. *Communications in Statistics—Simulation and Computation®*, 38(3):621–630, 2009.

[128] Kevin P. Murphy. *Machine learning : a probabilistic perspective.* MIT Press, Cambridge, Mass. [u.a.], 2013. ISBN 9780262018029 0262018020. URL https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr_1_2?ie=UTF8&qid=1336857747&sr=8-2.

[129] Balas Kausik Natarajan. Sparse approximate solutions to linear systems. *SIAM journal on computing*, 24(2):227–234, 1995.

[130] Radford M Neal et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

[131] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[132] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972.

[133] Tomoyuki Obuchi and Yoshiyuki Kabashima. Cross validation in lasso and its acceleration. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(5):053304, 2016.

[134] Trevor Park and George Casella. The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

[135] Pratik Patil, Yuting Wei, Alessandro Rinaldo, and Ryan Tibshirani. Uniform consistency of cross-validation estimators for high-dimensional ridge regression. In *International Conference on Artificial Intelligence and Statistics*, pages 3178–3186. PMLR, 2021.

[136] François Petitjean, Jordi Inglada, and Pierre Gançarski. Satellite image time series analysis under time warping. *IEEE transactions on geoscience and remote sensing*, 50(8):3081–3095, 2012.

[137] Juho Piironen and Aki Vehtari. On the hyperprior choice for the global shrinkage parameter in the horseshoe prior. In *Artificial Intelligence and Statistics*, pages 905–913. PMLR, 2017.

[138] Juho Piironen, Aki Vehtari, et al. Sparsity information and regularization in the horseshoe and other shrinkage priors. *Electronic Journal of Statistics*, 11(2):5018–5051, 2017.

[139] Martyn Plummer. Jags: Just another gibbs sampler. 2004.

[140] Nicholas G Polson and James G Scott. Shrink globally, act locally: Sparse Bayesian regularization and prediction. *Bayesian statistics*, 9:501–538, 2010.

[141] Nicholas G Polson, James G Scott, et al. On the half-cauchy prior for a global scale parameter. *Bayesian Analysis*, 7(4):887–902, 2012.

[142] Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108 (504):1339–1349, 2013.

[143] Nicholas G Polson, James G Scott, and Jesse Windle. The Bayesian bridge. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(4):713–733, 2014.

[144] Michael JD Powell. *A direct search optimization method that models the objective and constraint functions by linear interpolation.* Springer, 1994.

[145] Sudhir Raman, Thomas J Fuchs, Peter J Wild, Edgar Dahl, and Volker Roth. The Bayesian group-lasso for analyzing contingency tables. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 881–888, 2009.

[146] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

[147] Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363, 1996.

[148] SW Roberts. Control chart tests based on geometric moving averages. *Technometrics*, 42 (1):97–101, 2000.

[149] Veronika Ročková. Bayesian estimation of sparse signals with a continuous spike-and-slab prior. 2018.

[150] Håvard Rue. Fast sampling of gaussian markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):325–338, 2001.

[151] Lawrence Saul and Michael Jordan. Exploiting tractable substructures in intractable networks. *Advances in neural information processing systems*, 8, 1995.

[152] Todd E Scheetz, Kwang-Youn A Kim, Ruth E Swiderski, Alisdair R Philp, Terry A Braun, Kevin L Knudtson, Anne M Dorrance, Gerald F DiBona, Jian Huang, Thomas L Casavant, et al. Regulation of gene expression in the mammalian eye and its relevance to eye disease. *Proceedings of the National Academy of Sciences*, 103(39):14429–14434, 2006.

[153] Daniel F Schmidt and Enes Makalic. Estimation of stationary autoregressive models with the Bayesian lasso. *Journal of Time Series Analysis*, 34(5):517–531, 2013.

[154] Daniel F Schmidt and Enes Makalic. Log-scale shrinkage priors and adaptive Bayesian global-local shrinkage estimation. *arXiv preprint arXiv:1801.02321*, 2018.

[155] Daniel F Schmidt and Enes Makalic. Bayesian generalized horseshoe estimation of generalized linear models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 598–613. Springer, 2019.

[156] Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2): 461–464, 1978.

[157] Jun Shao. Linear model selection by cross-validation. *Journal of the American statistical Association*, 88(422):486–494, 1993.

[158] Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of computational and graphical statistics*, 22(2):231–245, 2013.

[159] David Spiegelhalter, Andrew Thomas, Nicky Best, and Wally Gilks. BUGS 0.5: Bayesian inference using gibbs sampling manual (version ii). *MRC Biostatistics Unit, Institute of Public Health, Cambridge, UK*, pages 1–59, 1996.

[160] Thomas A Stamey, John N Kabalin, John E McNeal, Iain M Johnstone, Fuad Freiha, Elise A Redwine, and Norman Yang. Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. ii. radical prostatectomy treated patients. *The Journal of urology*, 141(5):1076–1083, 1989.

[161] Will Stephenson, Zachary Frangella, Madeleine Udell, and Tamara Broderick. Can we globally optimize cross-validation loss? Quasiconvexity in ridge regression. *Advances in Neural Information Processing Systems*, 34:24352–24364, 2021.

[162] William E Strawderman. Proper Bayes minimax estimators of the multivariate normal mean. *The Annals of Mathematical Statistics*, 42(1):385–388, 1971.

[163] Shu Yu Tew, Daniel F Schmidt, and Enes Makalic. Sparse horseshoe estimation via expectation-maximisation. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, September 19–23, 2022, Proceedings, Part V*, pages 123–139. Springer, 2023.

[164] Shu Yu Tew, Mario Boley, and Daniel F Schmidt. Bayes beats Cross Validation: Efficient and Accurate Ridge Regression via Expectation Maximization. *Advances in Neural Information Processing Systems (to appear)*, 2024.

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

[166] Robert Tibshirani. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(3):273–282, 2011. doi: https://doi.org/10.1111/j.1467-9868.2011.00771.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.00771.x.

[167] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(1):91–108, 2005.

[168] Robert Tibshirani, Martin Wainwright, and Trevor Hastie. *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC, 2015.

[169] Michalis K Titsias and Omiros Papaspiliopoulos. Auxiliary gradient-based sampling algorithms. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 80(4): 749–767, 2018.

[170] Stéphanie van der Pas, Botond Szabó, and Aad van der Vaart. Adaptive posterior contraction rates for the horseshoe. *Electronic Journal of Statistics*, 11(2):3196–3225, 2017.

[171] Stéphanie L Van Der Pas, Bas JK Kleijn, Aad W Van Der Vaart, et al. The horseshoe estimator: Posterior concentration around nearly black vectors. *Electronic Journal of Statistics*, 8(2):2585–2618, 2014.

[172] Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

[173] Sara Van Erp, Daniel L Oberski, and Joris Mulder. Shrinkage priors for Bayesian penalized regression. *Journal of Mathematical Psychology*, 89:31–50, 2019.

[174] M Varah, James. On the numerical solution of ill-conditioned linear systems with applications to ill-posed problems. *SIAM Journal on Numerical Analysis*, 10(2):257–267, 1973.

[175] Sergei Vassilvitskii, Satyen Kale, and Ravi Kumar. Cross-validation and mean-square stability.

[176] Matthew P Wand, John T Ormerod, Simone A Padoan, and Rudolf Frühwirth. Mean field variational Bayes for elaborate distributions. 2011.

[177] Boxiang Wang and Hui Zou. Honest leave-one-out cross-validation for estimating post-tuning generalization error. *Stat*, 10(1):e413, 2021.

[178] Shuaiwen Wang, Wenda Zhou, Haihao Lu, Arian Maleki, and Vahab Mirrokni. Approximate leave-one-out for fast parameter tuning in high dimensions. In *International Conference on Machine Learning*, pages 5228–5237. PMLR, 2018.

[179] Greg CG Wei and Martin A Tanner. A monte carlo implementation of the em algorithm and the poor man's data augmentation algorithms. *Journal of the American statistical Association*, 85(411):699–704, 1990.

[180] Thomas Weise. Global optimization algorithms-theory and application. *Self-Published Thomas Weise*, 361, 2009.

[181] Ashia Wilson, Maximilian Kasy, and Lester Mackey. Approximate cross-validation: Guarantees for model assessment and selection. In *International Conference on Artificial Intelligence and Statistics*, pages 4530–4540. PMLR, 2020.

[182] Ji Xu, Arian Maleki, Kamiar Rahnama Rad, and Daniel Hsu. Consistent risk estimation in moderately high-dimensional linear regression. *IEEE Transactions on Information Theory*, 67(9):5997–6030, 2021.

[183] S Xu. An expectation–maximization algorithm for the lasso estimation of quantitative trait locus effects. *Heredity*, 105(5):483–494, 2010.

[184] Xiaofan Xu and Malay Ghosh. Bayesian variable selection and estimation for group lasso. 2015.

[185] Zemei Xu, Daniel F Schmidt, Enes Makalic, Guoqi Qian, and John L Hopper. Bayesian grouped horseshoe regression with application to additive models. In *Australasian Joint Conference on Artificial Intelligence*, pages 229–240. Springer, 2016.

[186] Zemei Xu, Daniel F Schmidt, Enes Makalic, Guoqi Qian, and John L Hopper. Bayesian sparse global-local shrinkage regression for selection of grouped variables. *arXiv preprint arXiv:1709.04333*, 2017.

[187] ASAR Yasin, Adnan Karaibrahimoğlu, and GENÇ Aşır. Modified ridge regression parameters: A comparative monte carlo study. *Hacettepe Journal of Mathematics and Statistics*, 43(5): 827–841, 2013.

[188] Hanjun Yu, Xinyi Xu, and Di Cao. The adaptive normal-hypergeometric-inverted-beta priors for sparse signals. *Journal of Statistical Computation and Simulation*, 91(2):396–419, 2021.

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

[190] Cun-Hui Zhang et al. Nearly unbiased variable selection under minimax concave penalty. *The Annals of statistics*, 38(2):894–942, 2010.

[191] Yongli Zhang and Yuhong Yang. Cross-validation for selecting a model selection procedure. *Journal of Econometrics*, 187(1):95–112, 2015.

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

[193] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.