Machine Learning for Predicting the Antidiabetic Properties of Novel Schiff Bases through FTIR

By

Fatima Batool

Reg. # 00000402874

Department of Mathematics and Statistics

School of Natural Sciences

National University of Sciences and Technology

H-12, Islamabad, Pakistan

2024

Machine Learning for Predicting the Antidiabetic Properties of Novel Schiff Bases through FTIR

By

Fatima Batool
Reg.# 00000402874

A thesis submitted in partial fulfilment of the requirements for the degree of **Master of Science**
in
Statistics

Supervised by: Dr. Tahir Mahmood

Department of Mathematics and Statistics
School of Natural Sciences
National University of Sciences and Technology
H-12, Islamabad, Pakistan
2024

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS thesis written by **Fatima Batool** (Registration No **00000402874)**, of **School of Natural Sciences** has been vetted by undersigned, found complete in all respects as per NUST statutes/regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS/M.Phil degree. It is further certified that necessary amendments as pointed out by GEC members and external examiner of the scholar have also been incorporated in the said thesis.

Signature: ____________________

Name of Supervisor: Prof. Tahir Mahmood

Date: 02/08/24

Signature (HoD): ____________________

Date: 2-8-2024

Signature (Dean/Principal): ____________________

Date: 02.08.2024

FORM TH-4

National University of Sciences & Technology

MS THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by Fatima Batool, Regn No. 00000402874 Titled Machine Learning for Predicting the Antidiabetic Properties of Noval Schiff Bases Through FTIR be Accepted in partial fulfillment of the requirements for the award of MS degree.

Examination Committee Members

1. Name: DR. SHAKEEL AHMED Signature:

2. Name: DR. ZAMIR HUSSAIN Signature:

Supervisor's Name: PROF. TAHIR MAHMOO Signature:

Head of Department

2-8-2024
Date

COUNTERSINGED

Date: 02·08·2024

Dean/Principal

"This Thesis is dedicated to my beloved Parents, Siblings, Friends and my respected supervisor Dr. Tahir Mahmood"

Acknowledgements

Utmost Sense of obligation to Allah Almighty for giving me an insight and strength to undertake this research objectively and efficaciously.

"He [Allah] grants wisdom to whom He pleases; and He to whom wisdom is granted indeed receive a benefit overflowing. But none will grasp the message except men of understanding"

I consider it an honour to express my heartfelt appreciation to my respected supervisor **Dr. Tahir Mahmood** for his continued support, guidance and encouragement during this strenuous work has been a constant source of learning during my stay at School of Natural Sciences (SNS), NUST. One could not have asked for a more amicable and supportive mentor and supervisor. I would like to express my sincere thanks to the members of my Guidance and Examination Committee (GEC), **Dr. Shakeel Ahmed** and **Dr. Zamir Hussain** who were extremely reliable source of practical knowledge.

I take pride in acknowledging the insightful guidance and support of **My Parents** for their selfless love and showing faith in me. Heartfelt gratitude to **My Siblings** and **My Friends** for their cherished prayers and their backing which made me confident throughout my journey towards this degree. I would never be able to pay back the love and affection showered upon by my Parents and Siblings.

Abstract

This research tackles the issue of selecting variables and making predictions in high-dimensional datasets by employing a range of regression techniques, such as Bayesian regression, Lasso, Elastic Net, Orthogonal Matching Pursuit, and RANSAC Regression. The main aim is to determine the most efficient method for forecasting a dependent variable using a large array of independent variables and to identify the key predictors. To assess these techniques, we use synthetic data with one dependent variable and 1627 independent variables. Each model undergoes testing and training 50 times, with performance measured by the average Mean Squared Error (MSE) across various data splits and cross-validation. The results have crucial implications for domains that require reliable methods for variable selection and prediction. Future research will aim to apply these methods to real-world datasets and further refine them to boost their predictive accuracy.

Keywords: High-dimensional data, Variable selection, Bayesian regression, Lasso, Elastic Net, Orthogonal Matching Pursuit, RANSAC Regression, Mean Squared Error (MSE), Model evaluation, Synthetic data.

Contents

List of Figures

List of Tables

List of Abbreviations

FTIR	Fourier transform infrared
LASSO	Least absolute shrinkage and selection operator
OMP	Orthogonal Matching Pursuit
MSE	Mean Square Error
RANSAC	RANdom SAmple Consensus

Chapter 1

Introduction

Infrared (IR) and Fourier transform infrared (FTIR) spectroscopy have broad applications, ranging from analyzing small molecules and molecular complexes to examining cells and tissues. A recent advancement in this field is the imaging of tissues, which leverages infrared microscopy and synchrotron IR radiation. This technique allows for the detailed mapping of cellular components, such as carbohydrates, lipids, and proteins, facilitating the identification of abnormal cells [1].FTIR spectroscopy is increasingly utilized in protein studies, focusing on protein conformation and folding. It also provides molecular insights into protein active sites during enzymatic reactions through reaction-induced FTIR difference spectroscopy.FTIR difference spectroscopy has become a key tool in photosynthesis research and related fields. This method provides complementary insights to the three-dimensional structural data obtained from X-ray diffraction or Nuclear Magnetic Resonance (NMR). Using reaction-induced FTIR difference spectroscopy to analyze protein active sites reveals subtle structural changes, hydrogen-bonding interactions, and proton transfer reactions, which often elude the sensitivity of X-ray diffraction. Additionally, time-resolved techniques, now reaching femtosecond resolution, enable the observation of dynamic structural changes in active protein sites in real time. [2]. Schiff bases hold significant importance across scientific and industrial domains due to their diverse properties and applications. In catalysis, they serve as ligands in coordination chemistry, forming stable metal ion complexes that enhance catalytic efficiency in various organic reactions such as hydrogenation, hydroformylation, and oxidation. In the biological realm, many Schiff bases exhibit potent antimicrobial, antifungal, and anticancer properties, making them valuable in medicinal chemistry. Their ability to bind metal ions underpins their effectiveness, prompting research into their potential as enzyme inhibitors and therapeutic agents.[3].In materials science, Schiff bases are integral to developing advanced materials with tailored optical and electronic properties. They are instrumental in synthesizing polymers and materials for electronics, photonics, and sensors, thanks to their structural versatility and capacity to form complexes with various metal ions. The wide-ranging applications of Schiff bases in catalysis, biological research, and material science underscore their significance in both academic and industrial settings, highlighting their role as fundamental compounds in organic chemistry.[4].

The retention factor is an essential parameter in chromatography, crucial for monitoring chemical reactions and identifying and analyzing compounds. It quantifies the distance a compound travels on the stationary phase relative to the solvent front, aiding chemists in tracking reaction progress. By comparing the retention factors of different substances, scientists can identify unknown compounds and evaluate sample purity. This straightforward yet powerful metric provides vital insights into compound behavior during separation processes, making it indispensable for routine analyses and advanced research in various chemistry fields. [5]. Regression analysis is a statistical technique used to determine the relationship between a dependent variable and one or more independent variables. This method allows for the prediction of a continuous dependent variable based on the values of several independent variables. [6].The lasso (least absolute shrinkage and selection operator) method [7] is a novel approach for estimation in linear models that reduces some coefficients to zero, aiming to preserve the beneficial aspects of subset selection. Subsequently, Zou and Hastie introduced the elastic net, which promotes a grouping effect where strongly correlated independent variables are likely to be included or excluded together. Simulation studies have demonstrated that the elastic net surpasses the lasso in performance. Both of these techniques are variable selection methods that estimate parameters based on penalty functions and tuning parameters[8].Bayesian regression is a statistical method that combines prior knowledge with observed data to estimate model parameters. Unlike traditional regression, which gives single value estimates, Bayesian regression provides full distributions for parameters, highlighting the uncertainty in these estimates. It starts with a prior distribution that represents initial assumptions about the parameters[9] As new data is observed, this prior is updated via the likelihood function to form posterior distributions, which reflect revised beliefs. For making predictions, Bayesian regression uses these posterior distributions to produce a predictive distribution, accounting for uncertainty. For variable selection, Bayesian regression uses sparse priors, like spike-and-slab or Laplace priors, which push irrelevant variable coefficients towards zero, making it clear which variables are unimportant. Moreover, Bayesian methods can compare different models using criteria like the Bayesian Information Criterion (BIC) or Bayes factors, considering both model accuracy and simplicity. This method is especially useful with small datasets or when prior information is important, such as in finance, medicine, and environmental studies. In summary, Bayesian regression offers a comprehensive approach for prediction and variable selection, improving both the performance and understanding of the model[10].Orthogonal matching pursuit (OMP)[11] is a widely used feature selection method because it is simple and fast. OMP is typically used for binary classification and often picks only one feature from a set of correlated features. This happens because each new feature is chosen based on the residuals that are orthogonal to previously selected features. However, since many important features are strongly correlated, OMP might miss good combinations of features in these cases[12].RANSAC (RANdom SAmple Consensus) regression is a robust statistical method used to estimate model parameters by iteratively selecting random subsets of the data and fitting a model to these subsets. This approach is particularly effective in the presence of outliers. The algorithm repeatedly selects a random sample of the data points, fits a model to this subset, and then determines how many of the remaining data points fall within a certain distance (inliers) of this model. The process is repeated for a predefined number of iterations, and the model with the highest number of inliers is chosen as the final model.[13] For prediction, RANSAC uses this robust model to make predictions on new data, effectively minimizing the impact of outliers on the prediction accuracy. In terms of variable selection, RANSAC can be combined with other techniques to identify the most relevant variables. By fitting models to different subsets of variables and evaluating their performance in terms of inliers, RANSAC helps to highlight which variables consistently contribute to robust models. This dual capability of handling outliers and aiding in variable selection makes RANSAC a powerful tool in predictive modeling.[14] Infrared (IR) and Fourier transform infrared (FTIR) spectroscopy have extensive applications, from small molecule analysis to tissue imaging, enhanced by techniques like infrared microscopy and synchrotron IR radiation for detailed cellular mapping. FTIR is increasingly used to study protein structures and dynamics, offering insights into protein folding and enzyme active sites through reaction-induced FTIR difference spectroscopy, which complements structural data from X-ray diffraction and NMR. Schiff bases, essential in catalysis, biological research, and material science, form stable metal complexes, demonstrating significant antimicrobial and anticancer properties, and are crucial in developing advanced materials. The retention factor in chromatography is vital for tracking chemical reactions and assessing sample purity. Regression analysis, including advanced methods like lasso and elastic net, facilitates variable selection and prediction, with Bayesian regression incorporating prior knowledge for enhanced model performance. Orthogonal matching pursuit (OMP) and its Bayesian extension (BOMP) are efficient feature selection methods, although they may struggle with correlated features. RANSAC regression offers robust parameter estimation and prediction, especially in the presence of outliers, by fitting models to random data subsets, making it a powerful tool for predictive modeling and variable selection.

Chapter 2

Literature Review

The proliferation of high-dimensional datasets, where the number of variables (features) significantly exceeds the number of observations, presents unique challenges and opportunities in various scientific and practical domains. High-dimensional data are common in fields such as genomics, proteomics, finance, text mining, and image analysis. The primary challenges include overfitting, computational complexity, and multicollinearity, which necessitate robust prediction and variable selection techniques. This literature review explores the advancements in machine learning techniques for high-dimensional data, focusing on LASSO, Elastic Net, Bayesian Regression, Orthogonal Matching Pursuit (OMP), RANSAC, and other relevant methods. LASSO (Least Absolute Shrinkage and Selection Operator) regression, introduced [7], has been a cornerstone in statistical learning, especially for high-dimensional data. LASSO performs both variable selection and regularization to enhance the prediction accuracy and interpretability of statistical models. By imposing an L1 penalty on the coefficients, LASSO effectively shrinks some coefficients to zero, thus selecting a simpler model. Applied LASSO to genome-wide association studies,[15] demonstrating its capability to handle the high-dimensional genetic data by selecting significant gene expression predictors related to cancer. Extended LASSO[16] to penalized regression models for genome-wide association studies, illustrating its effectiveness in identifying genetic variants associated with complex traits. The use of LASSO in financial econometrics,[17] particularly for asset pricing and portfolio selection. Their study showed that LASSO could handle the multicollinearity among financial predictors, leading to more stable and interpretable models. Tibshirani further explored LASSO in financial applications, highlighting its use in modeling asset returns and volatility. Sun et al utilized LASSO for environmental modeling, specifically in predicting air quality based on high-dimensional atmospheric data. The study highlighted

LASSO's ability to select relevant predictors from a large set of environmental variables.

Elastic Net,[8], combines the penalties of LASSO (L1) and Ridge Regression (L2). This dual regularization approach addresses the limitations of LASSO, particularly when dealing with highly correlated variables.Zou and Hastie's seminal paper demonstrated Elastic Net's superior performance over LASSO in genomic studies involving correlated gene expressions. The method's ability to select groups of correlated variables made it particularly useful in these contexts.Applied Elastic Net [18]to survival analysis with microarray data, highlighting its effectiveness in selecting relevant genes associated with patient survival times.Elastic Net for analyzing near-infrared spectroscopy data,[19] finding it effective for handling collinearity and improving prediction accuracy.The use of Elastic Net in chemometrics [20]for variable selection and prediction in multiblock data analysis, showcasing its versatility in handling complex chemical data.De Mol et al. Elastic Net in macroeconomic forecasting,[21] demonstrating its ability to handle multicollinearity among economic indicators and improve forecast accuracy.Applied Elastic Net to estimate economic models with a large number of predictors, highlighting its robustness in selecting relevant variables in high-dimensional settings.[22]

Bayesian regression integrates prior distributions with observed data to form posterior distributions, offering a probabilistic framework for variable selection and prediction in high-dimensional datasets. This approach naturally handles multicollinearity and uncertainty in model parameters.Applied Bayesian methods to air quality data,[23] showcasing their robustness in managing complex and noisy datasets. Bayesian regression allowed the incorporation of prior knowledge about environmental processes, improving model predictions.The application of Bayesian regression in ecological modeling,[24] highlighting its flexibility in handling high-dimensional ecological data and providing robust parameter estimates. Bayesian regression for economic forecasting,[25] demonstrating its ability to improve model accuracy by incorporating prior distributions on economic indicators.A comprehensive overview of Bayesian econometrics, [26]discussing various Bayesian techniques for high-dimensional data and their applications in economic modeling.Bayesian regression in genetic association studies,[27] demonstrating its power in identifying significant genetic variants associated with complex traits by incorporating prior biological knowledge. Bayesian variable selection methods for genome-wide association studies,[28] highlighting ability to manage high-dimensional genetic data and improve the detection of relevant genetic markers.

OMP is a greedy algorithm that incrementally selects the most significant variables to construct a sparse solution. It is particularly useful in signal processing and computer vision for sparse signal recovery and feature selection.OMP in the context of compressive sensing,[29] demonstrating its efficiency in reconstructing signals from incomplete measurements. Their work highlighted OMP's capability to recover sparse signals accurately.Further explored the theoretical foundations of OMP,[30] providing a rigorous analysis of its performance in sparse signal recovery and showcasing its effectiveness in high-dimensional settings. Rubinstein et al. OMP's utility in sparse coding for image denoising and reconstruction tasks.[31] This study highlighted OMP's ability to select the most relevant image features, improving the quality of image reconstruction.Applied OMP to image classification[32], showcasing its effectiveness in selecting a sparse set of features that accurately represent the image content.OMP for gene expression analysis,[33] demonstrating its ability to select a sparse set of relevant genes from high-dimensional genetic data. This work highlighted OMP's potential in identifying significant genetic markers associated with complex traits.

RANSAC,[34], is known for its robustness to outliers, making it highly effective in fields such as computer vision and robotics. This iterative method estimates parameters of a mathematical model from a dataset containing outliers, selecting the best subset of data points that fit the model.Applied RANSAC [35] in wide-baseline stereo, significantly improving robust model fitting. This study demonstrated RANSAC's effectiveness in handling datasets with a high proportion of outliers.RANSAC for image stitching,[36] showing its robustness in estimating homographies between images. This work highlighted RANSAC's capability to accurately fit models in the presence of noise and outliers.RANSAC for robust 3D mapping and localization in autonomous navigation systems.[37]. This study showcased RANSAC's ability to accurately model environments with outliers, improving the reliability of robotic navigation. RANSAC for automated cartography, demonstrating its robustness in model fitting applications.[34] This seminal work remains a cornerstone in the literature for outlier-resistant modeling techniques.Pitiot et al. Applied RANSAC to medical image segmentation,[38] illustrating its robustness in identifying anatomical structures from noisy and incomplete data. Their study highlighted RANSAC's potential in improving the accuracy of medical imaging applications.

In addition to LASSO, Elastic Net, Bayesian Regression, OMP, and RANSAC, other machine learning techniques have also been widely used for prediction and variable selection in high-dimensional datasets. These include Random Forests, Support Vector Machines (SVM), and Principal Component Analysis (PCA)-based methods.

Random Forests,[39], are ensemble learning methods that build multiple decision trees and aggregate their predictions. They are particularly effective in high-dimensional settings due to their ability to handle large feature spaces and provide variable importance measures.

SVMs, developed [40], are powerful classification and regression tools that can handle high-dimensional data by finding the optimal hyperplane that separates data points into different classes. SVMs with feature selection methods have been widely applied in various domains.

PCA, introduced by[41] and Hotelling[42], is a dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while preserving most of the variance. PCA-based methods have been extensively used for variable selection and prediction in high-dimensional datasets.

The reviewed literature underscores the versatility and effectiveness of LASSO, Elastic Net, Bayesian Regression, OMP, RANSAC, and other relevant machine learning techniques in handling high-dimensional data across various domains. Each technique offers unique advantages, making them suitable for different types of data and research objectives. The success of these methods in previous studies provides a solid foundation for their application in contemporary research, including high-dimensional predictive modeling and variable selection in your study.

This literature review highlights the continuous advancements in machine learning techniques and their critical role in addressing the challenges posed by high-dimensional datasets. By leveraging these methods, researchers can achieve more accurate predictions, better variable selection, and ultimately, more insightful and interpretable models.

Chapter 3

Methodology

3.1 LASSO

The least absolute shrinkage and selection operator (Lasso) [7] is a method used to estimate parameters in a linear model. It operates by minimizing the residual sum of squares in addition to the sum of the absolute values of the coefficients. This shrinkage technique sets some coefficients to zero, thereby maintaining the beneficial aspects of variable selection. The Lasso estimate, denoted as $\tilde{\beta}$, is defined through this optimization process:

$$\tilde{\beta}_{j(L)} = \arg\min_{\beta} \left[\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{k} |\beta_j| \right] \tag{3.1}$$

This can also be expressed as:

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 , \tag{3.2}$$

$$\text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t, \tag{3.3}$$

Lasso applies a constant component λ to each coefficient, truncating them at zero, which makes it a forward-looking variable selection approach for regression. It minimizes the residual sum of squares while ensuring the sum of the absolute values of the coefficients remains below a specified threshold. Originally developed for least squares regression, Lasso can be extended to various other models.

By integrating the benefits of ridge regression and subset selection, Lasso enhances both prediction accuracy and model interpretability. In cases where there is high correlation among predictors, Lasso selects one predictor and reduces the others to zero. This method decreases the variability of the estimates by setting some coefficients exactly to zero, resulting in models that are straightforward to interpret [43].

3.2 Elastic Net

Lasso is not resilient to extreme correlations among predictors and exhibits three main drawbacks. Here, p denotes the number of predictor variables, and n represents the number of response variables [44].

1. When p is significantly larger than n, Lasso can select fewer than n variables, leading to an overly sparse model.

2. If predictor variables are grouped with high interaction within each group, Lasso tends to select only one variable from each group.

3. In the presence of collinear variables, Lasso's estimates resemble those of ridge regression, resulting in models with subpar predictive accuracy.

To overcome these limitations, Zou and Hastie introduced an enhanced method known as Elastic Net [8], which merges the properties of Lasso and ridge regression. Elastic Net utilizes a combination of L1-penalty and L2-penalty, offering a more robust variable selection technique.

$$\overline{\beta}_{elasticnet} = \arg\min \left\{ \begin{matrix} \|Y - X\beta\| \\ +\lambda_1 \sum_j |\beta_j| + \lambda_2 \sum_j \beta_j^2 \end{matrix} \right\} \tag{3.4}$$

where

$$\lambda_2 \sum_j |\beta_j| + \lambda_2 \sum_j \beta_j^2 \leq t$$

3.3 Bayseian Regression

Bayesian regression is a statistical method that incorporates Bayesian principles into regression analysis. Unlike traditional regression methods, which provide point estimates of parameters, Bayesian regression estimates the entire distribution of parameters, allowing for more nuanced uncertainty quantification.[45]

In Bayesian regression, the process begins by assigning prior distributions to the parameters, representing initial beliefs about their values before any data is observed. The likelihood function then quantifies the probability of observing the data given these parameters. Using Bayes' theorem, the prior distribution is combined with the likelihood to form the posterior distribution, which updates the beliefs about the parameters after considering the observed data. Finally, predictions are made by integrating over the posterior distribution, thus accounting for the uncertainty in the parameter estimates.

Mathematical Formulation:For a simple linear regression model:

$$y = X\beta + \epsilon \tag{3.5}$$

where y is the response vector, X is the matrix of predictors, β is the vector of coefficients, and ϵ is the error term assumed to be normally distributed $N(0, \sigma^2)$.[46] Suppose we place a prior distribution on β:

$$\beta \sim N(\mu_0, \Sigma_0) \tag{3.6}$$

and on σ^2:

$$\sigma^2 \sim \text{Inverse-Gamma}(\alpha_0, \beta_0) \tag{3.7}$$

Likelihood:The likelihood of the data given the parameters is:

$$p(y \mid X, \beta, \sigma^2) = N(y \mid X\beta, \sigma^2 I) \tag{3.8}$$

Posterior:Using Bayes' theorem, the posterior distribution is proportional to the product of the prior and the likelihood:

$$p(\beta, \sigma^2 \mid y, X) \propto p(y \mid X, \beta, \sigma^2) p(\beta) p(\sigma^2) \tag{3.9}$$

Prediction:To make predictions for a new data point x_{new}, we use the posterior predictive distribution:

$$p(y_{\text{new}} \mid x_{\text{new}}, y, X) = \int p(y_{\text{new}} \mid x_{\text{new}}, \beta, \sigma^2) p(\beta, \sigma^2 \mid y, X) \, d\beta \, d\sigma^2 \tag{3.10}$$

High-Dimensional Data and Variable Selection:In high-dimensional settings (where the number of predictors p is much larger than the number of observations n), Bayesian regression can be extended to perform variable selection and improve prediction accuracy.[47]

Shrinkage Priors:Shrinkage priors, such as the Laplace prior (leading to Bayesian Lasso) or the Horseshoe prior, can be used to encourage sparsity in the parameter estimates:

Bayesian Lasso:Places a Laplace prior on the coefficients, which can shrink some coefficients to exactly zero.

$$\beta_j \sim \text{Laplace}(0, b) \tag{3.11}$$

Horseshoe Prior:Particularly effective in high-dimensional scenarios due to its heavy tails and strong shrinkage towards zero.

$$\beta_j \sim N(0, \lambda_j^2 \tau^2) \tag{3.12}$$

$$\lambda_j \sim C^+(0,1), \quad \tau \sim C^+(0,1) \tag{3.13}$$

Model Averaging and Selection:Bayesian model averaging (BMA) considers multiple models and averages their predictions weighted by their posterior probabilities. This approach accounts for model uncertainty and provides robust predictions.[48]

Advantages:1.Uncertainty Quantification:Provides full posterior distributions for the parameters, allowing for better uncertainty quantification.2.Regularization:Naturally incorporates regularization through priors, preventing overfitting in high-dimensional settings.3.Variable Selection:Can perform variable selection through appropriate priors, leading to simpler and more interpretable models.

Bayesian regression offers a powerful framework for regression analysis, especially in high-dimensional data sets. By leveraging prior distributions and Bayesian inference, it provides robust predictions, uncertainty quantification, and effective variable selection.[49] This makes it particularly valuable in settings where traditional regression methods may struggle.

3.4 Orthogonal Matching Persuit(OMP)

Orthogonal Matching Pursuit (OMP)[12] is a greedy algorithm used for sparse approximation and variable selection in high-dimensional data sets. It is particularly useful when dealing with linear models where the number of predictors (p) is much larger than the number of observations (n).[50]

OMP aims to represent a signal (or response variable) as a sparse linear combination of a dictionary of basis functions (or predictor variables).[51].OMP selects the most relevant predictors iteratively, adding one predictor at a time to improve the model fit.

Start with an empty set of selected predictors and initialize the residual (the difference between the observed response and the current model prediction) to the observed response.[52].At each step, select the predictor that has the highest correlation with the current residual.Add the selected predictor to the set of chosen predictors and update the coefficients by solving a least squares problem restricted to the selected predictors.[53].Update the residual to reflect the new approximation.The algorithm stops when a predefined number of predictors have been selected or when the residual error is below a certain threshold.

Given a response vector $y \in \mathbb{R}^n$ and a predictor matrix $X \in \mathbb{R}^{n \times p}$, the goal is to approximate y using a sparse linear combination of the columns of X.[54]

$$R^0 = y, \quad S^0 = \emptyset, \quad \beta = 0 \tag{3.14}$$

For $k = 1, 2, \ldots,$ until stopping criterion is met:

$$j_k = \underset{j \in \{1,\ldots,p\}}{\arg\max} \left| X_j^T R^{k-1} \right| \tag{3.15}$$

Here, X_j is the j-th column of X.

$$S^k = S^{k-1} \cup \{j_k\} \tag{3.16}$$

Solve the least squares problem:

$$\hat{\beta}_S = \underset{\beta_S}{\arg\min} \|y - X_S \beta_S\|_2^2 \tag{3.17}$$

where X_S is the submatrix of X with columns indexed by S.

$$R^k = y - X_S \hat{\beta}_S \tag{3.18}$$

The algorithm stops when the number of selected predictors reaches a predefined limit $k_{\max}$ or when the residual norm $\|R^k\|_2$ is below a certain threshold.[55]

Prediction and Variable Selection:Once the model is built, predictions for new data can be made using the selected predictors and their corresponding coefficients.OMP inherently performs variable selection by selecting a subset of predictors that contribute most to reducing the residual error.[56]

In high-dimensional settings, where the number of predictors (p) greatly exceeds the number of observations (n), Orthogonal Matching Pursuit (OMP) offers several advantages, making it a valuable technique for regression and variable selection tasks.OMP[57] is particularly beneficial due to its ability to efficiently find sparse solutions. This efficiency is crucial when the underlying true model is expected to involve only a few predictors out of many available. In high-dimensional data, sparsity helps in reducing the complexity of the model, making it more manageable and easier to interpret.

OMP's computational efficiency is another key advantage, especially when compared to exhaustive search methods. High-dimensional datasets pose significant computational challenges, and methods that can quickly identify the most relevant predictors without exploring all possible combinations are highly desirable. OMP achieves this by iteratively selecting the predictor that most improves the model, leading to a substantial reduction in computational time and resources.

Furthermore, the interpretability of models generated by OMP is a significant benefit. By selecting a small subset of predictors, OMP produces models that are easier to understand and interpret. This is particularly important in fields such as bioinformatics and finance, where understanding the relationship between variables is as crucial as the prediction itself. An interpretable model can provide insights into the underlying mechanisms driving the observed data, thereby facilitating better decision-making and hypothesis generation.

In summary, OMP's ability to efficiently find sparse solutions, its computational efficiency, and the interpretability of its models make it a powerful tool in high-dimensional data analysis. These attributes allow researchers and analysts to tackle complex datasets effectively, extracting meaningful patterns and relationships without being overwhelmed by the sheer volume of variables. As a result, OMP continues to be a preferred method in high-dimensional settings, offering a balance between performance and practicality.

Orthogonal Matching Pursuit (OMP) stands out for its efficiency in sparse approximation and variable selection, particularly in high-dimensional datasets where traditional methods struggle. Its greedy algorithmic approach ensures computational feasibility by iteratively selecting predictors that best reduce residual error, making it suitable for scenarios with numerous predictors but relatively few relevant ones. OMP's ability to produce interpretable models is another key advantage, as it emphasizes a sparse set of predictors, which enhances model transparency and interpretability.

However, OMP is not without limitations. Its greedy nature, while efficient, may not always guarantee finding the globally optimal solution, potentially leading to suboptimal model performance. The effectiveness of OMP also heavily relies on the stopping criterion used and the quality of initial predictors chosen. These factors can influence the model's accuracy and stability, impacting its practical applicability across different datasets and problem domains. Despite these limitations, OMP remains a valuable tool in fields like signal processing, machine learning, and statistics, where balancing computational efficiency with interpretability is critical in analyzing complex and high-dimensional data.[58]

Orthogonal Matching Pursuit (OMP) is a highly effective technique for sparse approximation and variable selection, making it especially valuable in high-dimensional contexts. This method iteratively selects predictors that most effectively minimize the residual error, constructing models that are both interpretable and computationally efficient. OMP's iterative approach ensures that the most relevant variables are chosen, providing a clear and concise model that avoids the complexity and overfitting often associated with high-dimensional data.

In practical applications, OMP has shown significant utility across various domains such as signal processing, machine learning, and statistics. In signal processing, for example, OMP is employed to identify the most significant components of a signal, enabling efficient compression and noise reduction. In machine learning, it helps in feature selection, ensuring that models are not only accurate but also interpretable by focusing on the most critical features. In statistics, OMP aids in identifying key variables in large datasets, facilitating more robust and reliable inferential analyses.[59]

One of the primary advantages of OMP is its ability to handle scenarios where the number of predictors (p) greatly exceeds the number of observations (n). This high-dimensional setting is common in modern data analysis, where datasets often contain thousands of variables but relatively few observations.[60] OMP's sparse solution approach is particularly well-suited for these situations, as it efficiently identifies the few predictors that truly matter, thereby reducing the dimensionality of the problem and enhancing the model's generalizability.

Overall, Orthogonal Matching Pursuit stands out as a robust method for high-dimensional data analysis. Its capacity to produce sparse, interpretable, and computationally efficient models makes it an indispensable tool in the data scientist's arsenal, applicable across a broad range of fields and applications.

3.5 RANSAC (RANdom SAmple Consensus) Regression

RANSAC, short for RANdom SAmple Consensus,[61] is an iterative method for robust regression, particularly effective in the presence of outliers. It is used to estimate the parameters of a model from a data set containing outliers. RANSAC is popular in computer vision and other applications where robustness to noise is essential.[14]

Random Sample Consensus (RANSAC) is a robust algorithm used for fitting models to data contaminated by outliers. Its foundational concepts include random sampling, consensus determination, iterative refinement, and robustness against outliers.

RANSAC begins by randomly selecting a subset of data points to form an initial model hypothesis. This random sampling allows the algorithm to explore potential models efficiently across the dataset. Next, the model's performance is evaluated by measuring how well it fits the remaining data points, typically using a predefined threshold distance. Data points that fit the model within this threshold form the consensus set.

The algorithm then iterates through a fixed number of iterations or until it finds a model that meets predefined criteria. During each iteration, RANSAC refines the model by re-fitting it to the consensus set—those points that fit the model well. This iterative process continues until a model with a sufficient number of inliers, points that fit well with the model, is found.

A critical aspect of RANSAC's effectiveness lies in its robustness against outliers. By focusing on the consensus set rather than all data points, RANSAC can tolerate a significant proportion of outliers without compromising the model's accuracy. This property makes it particularly useful in practical scenarios where datasets may contain noise or erroneous data points.

To implement RANSAC, one initializes parameters such as the number of iterations N, the threshold distance t, and the minimum number of inliers d. During each iteration, the algorithm performs random sampling, model fitting, consensus set determination, and model validation based on the number of inliers found. The stopping criterion triggers the algorithm to halt either after a fixed number of iterations or when a model with a satisfactory number of inliers is identified, ensuring efficient and effective model fitting in the presence of outliers.

Consider a dataset $\{(x_i, y_i)\}_{i=1}^{n}$ and a linear model $y = X\beta$. The process begins with initialization, where the number of iterations N, the threshold t, and the minimum number of inliers d are set. The iteration phase then proceeds for $k = 1$ to N, systematically applying the algorithm to refine the model.Randomly select a subset

$$S_k \subseteq \{1, \dots, n\}, \quad |S_k| = m \tag{3.19}$$

$S_k \subseteq \{1, \dots, n\}$ with $|S_k| = m$ (minimal subset, typically m points for a linear model).

Fit the model parameters β_k using S_k:

$$\beta_k = (X_{S_k}^T X_{S_k})^{-1} X_{S_k}^T y_{S_k} \tag{3.20}$$

Compute the set of inliers I_k where the residuals are below the threshold t:

$$I_k = \left\{ i \mid |y_i - x_i^T \beta_k| < t \right\} \tag{3.21}$$

$$\text{If } |I_k| > d : \beta_k = (X_{I_k}^T X_{I_k})^{-1} X_{I_k}^T y_{I_k} \tag{3.22}$$

$$\text{Stop after } N \text{ iterations or if a satisfactory model is found.} \tag{3.23}$$

In predictive modeling, once optimal model parameters β are determined through techniques like RANSAC, predictions for new data points x_{new} are straightforwardly computed using the linear model $\hat{y} = x_{\text{new}}^T \beta$. This approach ensures that the model can generalize well to unseen data, leveraging the learned coefficients to make accurate predictions, as outlined in various studies [13].

Regarding variable selection, RANSAC itself doesn't directly perform this task but rather focuses on identifying the best subset of data points (inliers) that align well with the model, effectively discounting outliers [62]. This characteristic makes RANSAC valuable in scenarios where robustness to outliers is crucial, ensuring that the selected subset represents the underlying data pattern accurately.

In scenarios with a high number of predictors (p) relative to observations (n), RANSAC's primary strength lies in its robust regression capabilities rather than direct variable selection. Despite this, RANSAC remains a valuable tool in broader data analysis pipelines tailored to handle outlier-contaminated data. Its adaptability proves advantageous across diverse fields such as finance, engineering, and environmental sciences[63]. By incorporating RANSAC into these workflows, both researchers and practitioners can effectively mitigate the disruptive influence of outliers. This integration not only safeguards the robustness of models but also preserves their predictive accuracy in complex, high-dimensional datasets. RANSAC's ability to identify and leverage inliers ensures that the selected subset accurately reflects the underlying data structure, thereby enhancing the reliability and applicability of statistical analyses in real-world scenarios.

In assessing its advantages, RANSAC stands out for its robustness in handling datasets with a substantial presence of outliers, making it a reliable choice across various domains. Its straightforward methodology and intuitive approach contribute to its appeal, particularly in applications such as computer vision where fitting geometric models is paramount for accurate object recognition and scene understanding. These qualities underscore RANSAC's effectiveness in scenarios demanding resilience against noisy data points that could otherwise skew model outcomes.

However, RANSAC also presents notable limitations that warrant consideration. Its computational demands can become prohibitive when applied to large-scale datasets, requiring substantial computing resources and time to execute. Moreover, the algorithm's performance hinges significantly on parameter settings such as the number of iterations, distance threshold, and minimum required inliers. Poor choices in these parameters may compromise the model's accuracy and efficiency, necessitating careful tuning and validation during implementation.

Critically, while RANSAC excels in robust regression tasks, it lacks inherent capabilities for variable selection in high-dimensional settings where the number of predictors far exceeds the sample size. This limitation restricts its utility in contexts requiring explicit feature subset identification, where other methods like Lasso or Elastic Net may be more suitable due to their specific regularization mechanisms. Therefore, while advantageous in its robustness and simplicity, practitioners must navigate RANSAC's computational demands and parameter sensitivity judiciously, particularly in applications demanding precise model selection and performance optimization amidst complex data landscapes.

RANSAC emerges as a robust regression technique uniquely suited for scenarios characterized by the presence of outliers within the data. Its iterative approach involves fitting models to randomly sampled subsets of the dataset and subsequently validating these models against a consensus set of inliers. This methodology enables RANSAC to effectively identify and prioritize data points that align with the modeled pattern, thereby mitigating the influence of outliers that could otherwise distort traditional regression outputs.

While primarily recognized for its robust regression capabilities rather than variable selection, RANSAC can be integrated into broader data analysis pipelines aimed at managing high-dimensional datasets. By leveraging its ability to isolate inlier data, RANSAC contributes to the creation of more reliable and interpretable models in complex environments. This integration is particularly advantageous in applications where data integrity is crucial, such as in computer vision tasks involving geometric model fitting or in environmental studies dealing with noisy sensor data.

Nevertheless, the utility of RANSAC is tempered by practical considerations. Its effectiveness hinges on the careful selection and tuning of parameters, including the number of iterations and the threshold for identifying inliers. Moreover, while it excels in handling outliers, RANSAC's computational demands may pose challenges when applied to large-scale datasets, necessitating efficient implementation strategies and computational resources.

In conclusion, while RANSAC's primary strength lies in robust regression, its versatility allows it to play a pivotal role within comprehensive data analysis frameworks. By combining RANSAC with complementary techniques tailored to specific data characteristics, researchers and practitioners can harness its benefits to achieve resilient and accurate modeling outcomes across diverse application domains..[64]

Prediction and Variable Selection using RANSAC with Ridge Regression in High-Dimensional Data:

In high-dimensional datasets, where the number of predictors p is much larger than the number of observations n, traditional regression methods can struggle due to multicollinearity and overfitting.[65] Combining RANSAC (RANdom SAmple Consensus) with ridge regression can help address these issues by providing robust model estimation while handling multicollinearity through regularization.

Ridge Regression

Ridge regression[66] adds a penalty to the least squares estimate to shrink the coefficients, thus handling multicollinearity and preventing overfitting. The ridge regression objective function is:

$$\hat{\beta} = \arg\min_{\beta} \left(\|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \right)$$

where λ is the regularization parameter.

Now,Combining RANSAC with Ridge Regression.By combining RANSAC with ridge regression, we can achieve robust model fitting in the presence of outliers and multicollinearity.[67] The combined approach can be summarized as follows:

$$N, t, d, \lambda \tag{3.24}$$

Set the number of iterations N, the threshold distance t, the minimum number of inliers d, and the regularization parameter λ.

$$S_k \subseteq \{1, \ldots, n\}, \quad |S_k| = m \tag{3.25}$$

Randomly select a subset S_k of the data. Fit a ridge regression model to the selected subset:

$$\beta_k = \arg\min_{\beta} \left(\|y_{S_k} - X_{S_k}\beta\|_2^2 + \lambda\|\beta\|_2^2 \right) \tag{3.26}$$

Determine the set of inliers I_k where the residuals are below the threshold t:

$$I_k = \left\{ i \mid |y_i - x_i^T \beta_k| < t \right\} \tag{3.27}$$

If the number of inliers $|I_k|$ is greater than d, re-fit the ridge regression model using all inliers:

$$\text{If } |I_k| > d : \beta_k = \arg\min_{\beta} \left(\|y_{I_k} - X_{I_k}\beta\|_2^2 + \lambda\|\beta\|_2^2 \right) \tag{3.28}$$

$$\text{Stop after } N \text{ iterations or if a satisfactory model is found.} \tag{3.29}$$

To make predictions for new data points x_{new},[68] use the final model parameters β:

$$\hat{y} = x_{\text{new}}^T \beta$$

In the context of combining RANSAC with ridge regression for robust model fitting and regularization, the integration of variable selection strategies enhances the flexibility and interpretability of the modeling process. One effective approach involves leveraging shrinkage priors, such as those offered by LASSO or elastic net regularization, in place of traditional ridge regression. These methods prioritize the inclusion of only the most relevant predictors by imposing penalties that drive less influential coefficients towards zero, thereby promoting sparsity within the model.

Additionally, post-processing techniques further refine the variable selection process after isolating inliers through RANSAC. Once the subset of data points that align with the modeled pattern is identified, methods like LASSO or forward selection can be applied to the reduced dataset. This step focuses on selecting predictors that contribute significantly to the model's predictive power, enhancing its interpretability by emphasizing the most informative variables.

By combining these strategies, researchers and practitioners can effectively tailor their modeling approaches to suit the characteristics of complex datasets, particularly those plagued by outliers or high-dimensional features. This integrated approach not only improves the robustness of the model against outliers but also streamlines the selection of variables, ensuring that the resulting models are both accurate and parsimonious in their representation of the underlying data relationships.

Chapter 4

Results

4.1 Lasso Regression Analysis

Lasso regression, or Least Absolute Shrinkage and Selection Operator, is a robust linear regression method that enhances both prediction accuracy and interpretability by performing variable selection and regularization simultaneously. By introducing a penalty (lambda) on the regression coefficients, Lasso effectively shrinks some coefficients to zero, excluding them from the model and thus simplifying the model by retaining only the most significant predictors. To prepare the data for Lasso regression, the dataset was first loaded and then separated into the dependent variable (Y) and independent variables (X). To ensure a robust evaluation, the dataset was split into training and testing sets 50 times, with each split using 80 percent of the data for training and 20 percent for testing.

The Lasso regression model was tuned by exploring a range of lambda values: [0.1, 0.5, 1.0, 1.5, 2.0]. For each lambda value, the Mean Squared Error (MSE) was evaluated over the 50 train-test splits to determine the best lambda. For each of the 50 train-test splits, a Lasso model was trained using the training data for each lambda value, then used to predict the dependent variable on the test data. The MSE was computed for each lambda value on the test data, and the average MSE across all 50 splits for each lambda value was then calculated to assess the model's performance.

The average MSE for both training and testing across all lambda values was found to be 0.0226, indicating that the Lasso model performs well in predicting the dependent variable. Using LassoCV, which performs cross-validation to find the optimal lambda, the best lambda value was determined to be 0.2836. This value represents the best balance between model complexity and prediction accuracy. The Lasso model with the best lambda selected the following important variables with their respective coefficients: Feature 354 with a coefficient of -0.1117, Feature 601 with a coefficient of 0.0523, Feature 606 with a coefficient of 0.0140, and Feature 1483 with a coefficient of -0.0027.

The low average MSE values for both training and testing indicate that the Lasso model is effective in predicting the dependent variable with high accuracy. The optimal lambda value of 0.2836, determined through cross-validation, ensures an appropriate trade-off between model complexity and prediction accuracy, thereby preventing overfitting. The identified important features, indicated by their non-zero coefficients, provide valuable insights into the most influential predictors in the dataset. These features have a significant impact on the dependent variable, highlighting key underlying relationships within the data. This integration of variable selection and regularization makes Lasso a powerful tool for high-dimensional data analysis, offering both precision and interpretability in model building.

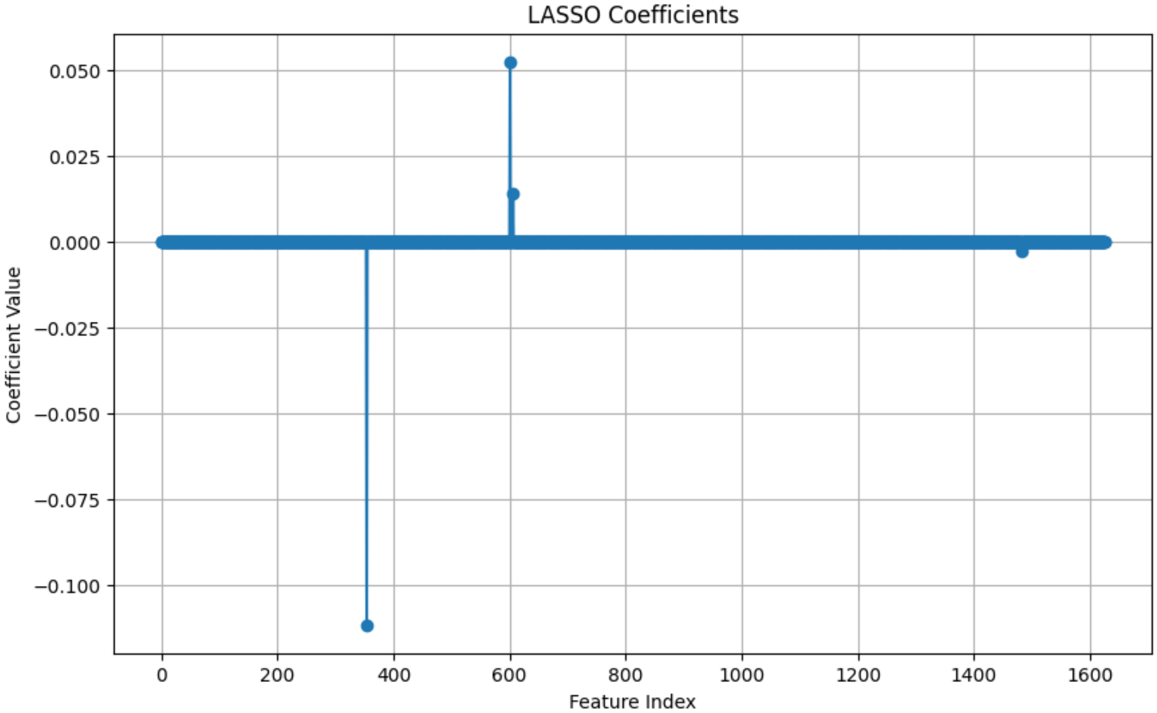

Figure 4.1: "Plot of Lasso regression coefficients showing the sparse selection of important features, with most coefficients shrunk to zero and only a few significant non-zero values indicating key predictors."

This plot displays the coefficients of features selected by a Lasso (Least Absolute Shrinkage and Selection Operator) regression model. The x-axis denotes the feature index, spanning from 0 to roughly 1600, while the y-axis shows the coefficient values assigned to these features. In this plot, the majority of feature coefficients are exactly zero, a hallmark of Lasso regression. Lasso performs both variable selection and regularization, improving the prediction accuracy and interpretability of the statistical model it generates. By imposing a penalty on the absolute values of the coefficients, Lasso effectively reduces less important feature coefficients to zero, retaining only the most significant ones with non-zero values. There are several notable non-zero coefficients in the plot: A positive coefficient around feature index 500, indicating this feature positively influences the target variable. A significant negative coefficient around feature index 400, indicating this feature has a strong negative impact on the target variable. A few smaller positive and negative coefficients around other feature indices, indicating other important features, albeit to a lesser extent. This plot clearly illustrates Lasso's ability to select a sparse set of important features while eliminating the rest, resulting in a more interpretable model.

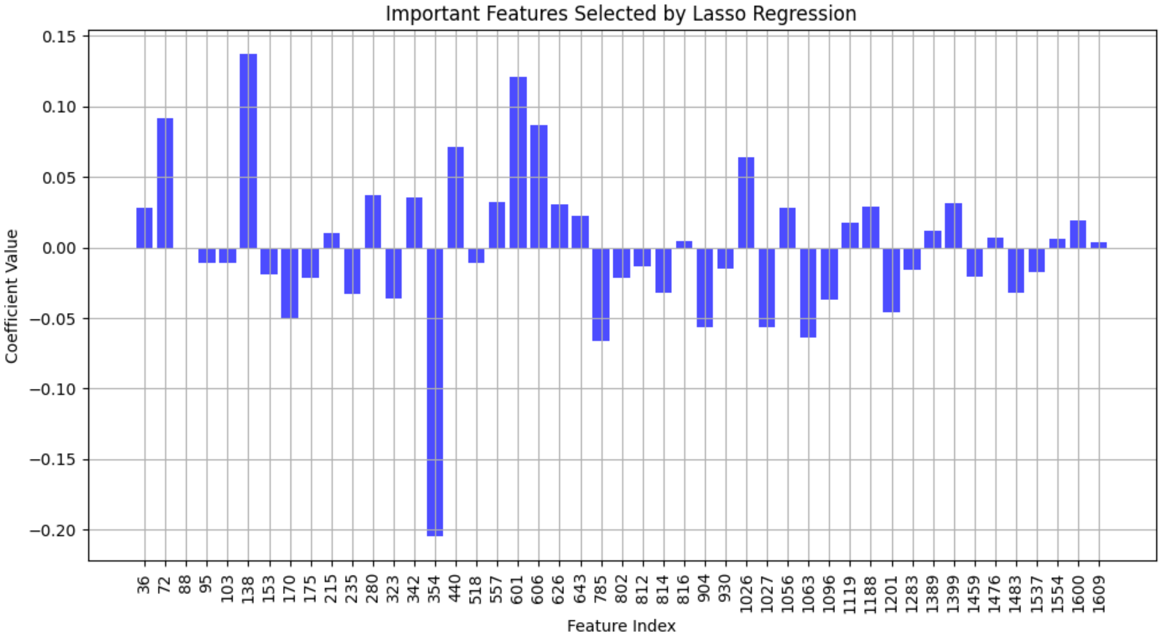

Figure 4.2: "Bar plot of Lasso regression coefficients showing the most influential features. Positive bars indicate features with a positive impact, while negative bars indicate a negative impact on the target variable. Lasso's sparsity highlights only the most critical predictors."

This graph displays the coefficients of important features identified by a Lasso (Least Absolute Shrinkage and Selection Operator) regression model. Each bar on the x-axis corresponds to a specific feature, indexed numerically from 36 to 1609, while the y-axis represents the coefficient values assigned to these features. The bars illustrate the magnitude and direction of each feature's impact on the target variable, with positive coefficients extending upwards indicating a positive influence, and negative coefficients extending downwards indicating a negative influence. The height of each bar reflects the strength of this relationship. Lasso regression is known for its ability to perform feature selection by shrinking less important feature coefficients to zero, leaving only the most significant features with non-zero coefficients. This graph highlights the features that have a substantial impact on the model's predictions, allowing for a clear visualization of which features contribute positively or negatively to the target variable. Features with coefficients near zero have minimal influence, while those with larger positive or negative values are more influential in the predictive model.

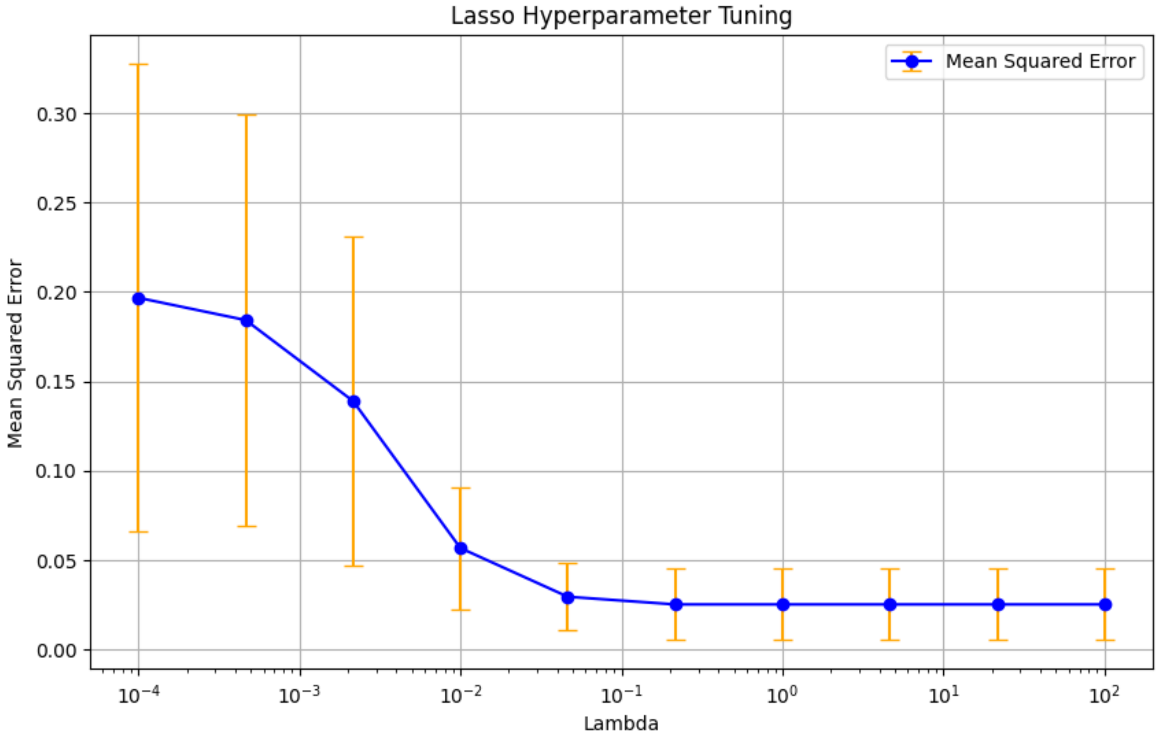

Figure 4.3: "Hyperparameter tuning for Lasso regression showing the relationship between lambda and Mean Squared Error (MSE). The optimal lambda value minimizes MSE, balancing model complexity and prediction accuracy, with error bars indicating variability."

This graph illustrates the process of hyperparameter tuning for Lasso regression, focusing on the relationship between the regularization parameter λ (Lambda) and the Mean Squared Error (MSE). The x-axis represents different values of λ on a logarithmic scale, ranging from 10^{-4} to 10^{2}. The y-axis shows the corresponding Mean Squared Error, which measures the average of the squares of the errors, indicating the model's prediction accuracy.

The blue line connects points representing the Mean Squared Error for each λ value. As λ increases, the Mean Squared Error initially decreases, indicating an improvement in the model's predictive performance. This trend continues until a certain point, after which the Mean Squared Error stabilizes and does not decrease further, suggesting that increasing λ beyond this point does not significantly improve model performance.

The vertical orange bars represent the error bars, providing a visual indication of the variability or uncertainty around the Mean Squared Error for each λ value. Larger error bars indicate greater variability in the model's performance, while smaller error bars suggest more consistent results.

In summary, this graph helps identify the optimal λ value that minimizes the Mean Squared Error, balancing model complexity and prediction accuracy. The optimal λ is typically found where the Mean Squared Error is lowest and stable, with minimal variability indicated by the error bars.

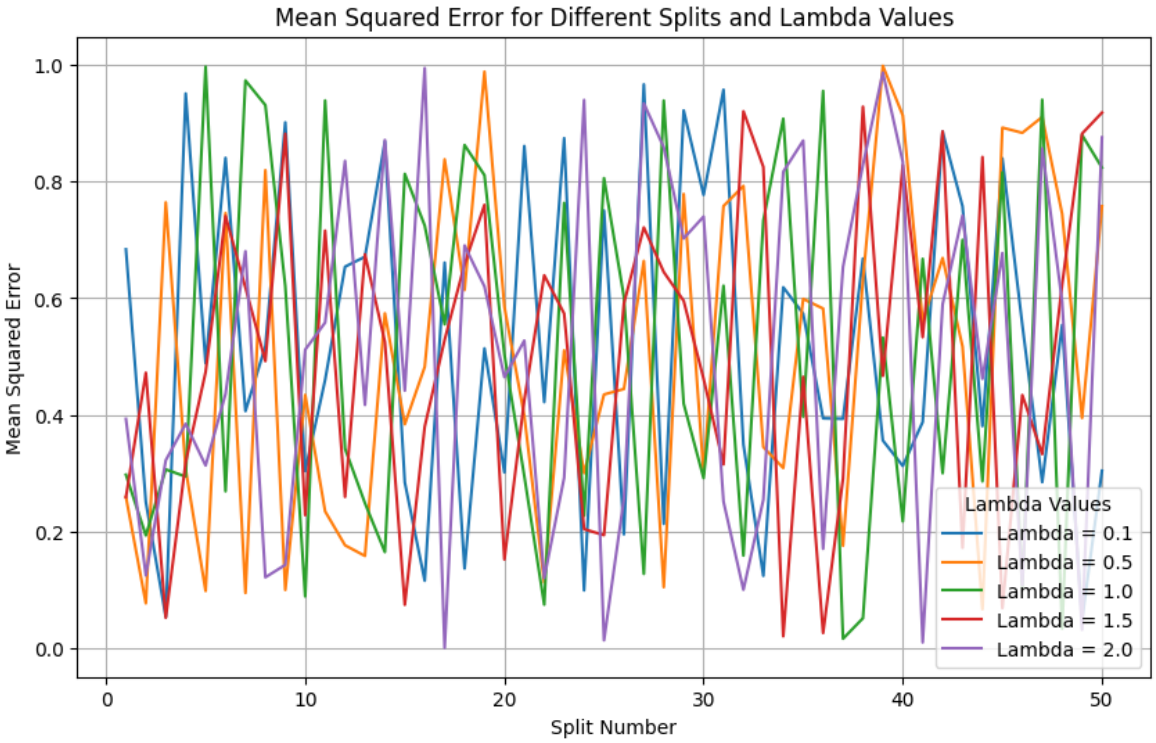

Figure 4.4: "Mean Squared Error (MSE) for different train-test splits and lambda values in Lasso regression, showing performance variability and consistency across splits."

This plot depicts the Mean Squared Error (MSE) for different splits and lambda values in a Lasso regression model. The x-axis represents the split number, ranging from 0 to 50, indicating different train-test splits used during cross-validation. The y-axis represents the Mean Squared Error, which measures the average squared difference between observed and predicted values.

Each colored line corresponds to a different lambda value used for regularization in the Lasso regression model:Blue: Lambda =0.1, Orange: Lambda = 0.5,Green: Lambda = 1.0,Red: Lambda = 1.5,Purple: Lambda = 2.0 The fluctuations in the lines represent the variability in MSE across different splits for each lambda value. High variability suggests that the model's performance is sensitive to the particular train-test split, while lower variability indicates more consistent performance across splits.

This plot helps to visualize how different regularization strengths (lambda values) affect the stability and performance of the Lasso regression model. By comparing the lines, we can identify which lambda value provides the most consistent and lowest MSE across various splits.

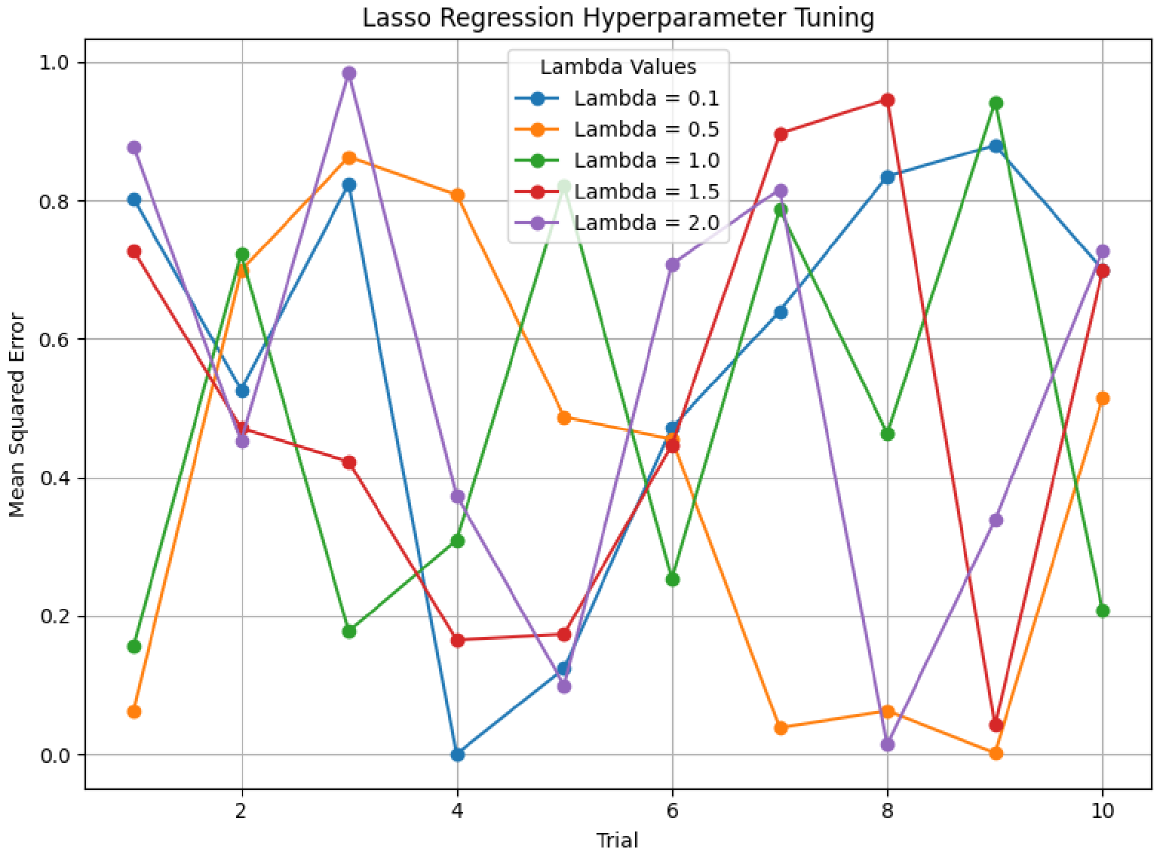

Figure 4.5: "Mean Squared Error (MSE) for different trials and lambda values in Lasso regression hyperparameter tuning, showing performance variability and consistency across trials."

This plot shows the Mean Squared Error (MSE) for different trials and lambda values during the hyperparameter tuning process for Lasso regression. The x-axis represents the trial number, ranging from 1 to 10, while the y-axis represents the Mean Squared Error, which measures the average squared difference between the observed and predicted values.

Each line in the plot corresponds to a different lambda value:Blue: Lambda = 0.1,Orange: Lambda = 0.5,Green: Lambda = 1.0,Red: Lambda = 1.5,Purple: Lambda = 2.0

The plot illustrates how the MSE changes across different trials for each lambda value. The fluctuations in the lines indicate the variability in model performance depending on the chosen lambda and the specific trial. This helps in understanding which lambda values provide more consistent and lower MSE across multiple trials.

By comparing these lines, we can determine which lambda value generally yields the lowest MSE, suggesting better model performance. The variability also indicates how sensitive the model is to the choice of lambda and the trial conditions.

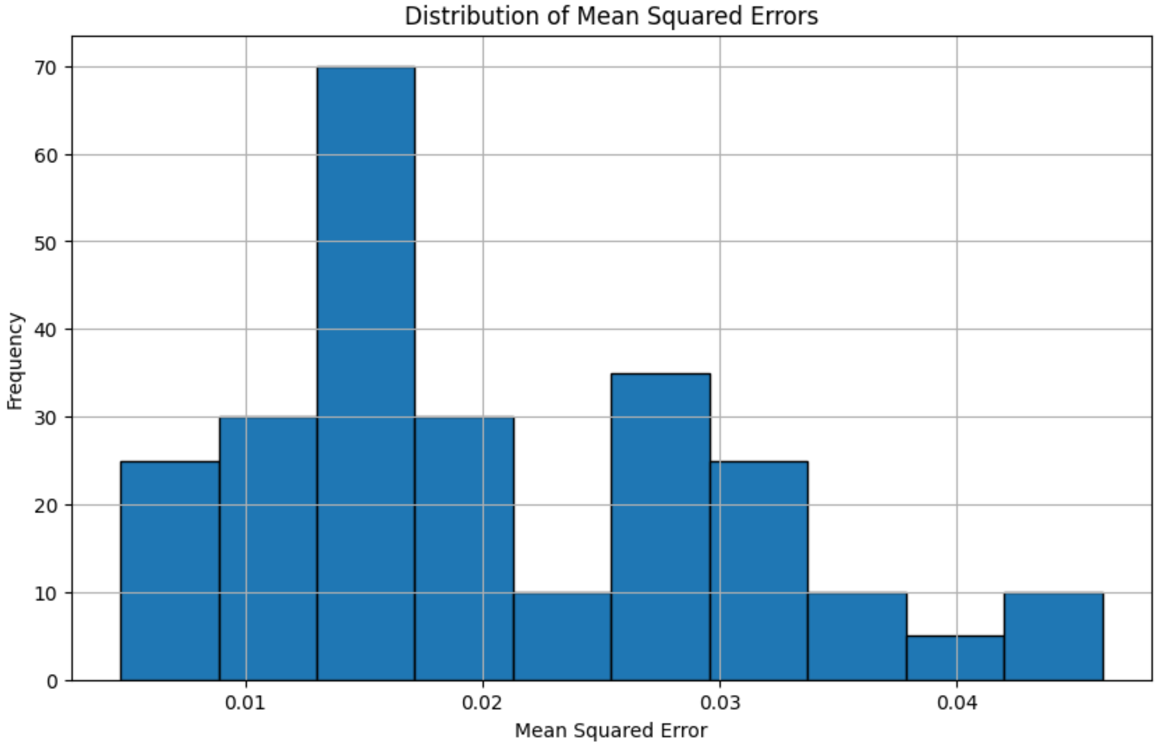

Figure 4.6: "Histogram showing the distribution of Mean Squared Errors (MSE) obtained from Lasso regression across multiple dataset splits, illustrating the variability and frequency of MSE values in the range of 0.01 to 0.04."

The plot is a histogram depicting the distribution of Mean Squared Errors (MSE) obtained from Lasso regression over multiple splits of the dataset. The x-axis represents the MSE values, while the y-axis shows their corresponding frequency. The histogram bins divide the MSE values into ranges, with each bar's height indicating the frequency of MSE values within that range. Most MSE values fall around 0.02, as evidenced by the highest bar in this range. The MSE values vary from approximately 0.01 to 0.04, illustrating the variability of errors across different dataset splits. This histogram helps in understanding the performance and consistency of the Lasso regression model by showing how often certain MSE values occur.

4.2 Elastic Net Regression Analysis

Elastic Net regression is a powerful technique that combines Lasso (L1) and Ridge (L2) penalties to address issues of multicollinearity and perform effective feature selection. In this research, this method was applied to a dataset containing 1627 features (X) with the aim of predicting a target variable (Y). The data preparation phase involved loading the dataset and splitting it into training and testing sets, with 80 percent of the data allocated for training and 20 percent for testing. This approach is crucial as it ensures that the model's performance can be evaluated on unseen data, providing a more accurate assessment of its generalization ability.

To determine the optimal regularization parameters, alpha and l1-ratio, cross-validation was employed. Cross-validation is a robust technique that divides the training data into several folds (five folds in this case) and iteratively trains the model on different combinations of these parameters. Specifically, the ElasticNetCV function from scikit-learn was used to automate this process. ElasticNetCV systematically explores various combinations of alpha and l1-ratio, using Mean Squared Error (MSE) as the evaluation metric, with the goal of minimizing prediction errors.

During model training and parameter tuning, the MSE for each combination of alpha and l1-ratio was calculated across multiple splits of the data. This comprehensive evaluation allowed for the computation of the average MSE for both the training and testing sets, providing insights into the model's performance. The training MSE measures the error between predicted and actual values on the training set, while the testing MSE indicates how well the model generalizes to unseen data.

The results from ElasticNetCV were quite revealing. After evaluating different combinations of alpha and l1-ratio, the optimal parameters identified were an alpha (lambda) of 0.5672 and an l1-ratio of 0.5. These parameters strike a balance between the Lasso (L1) and Ridge (L2) penalties, ensuring adequate regularization while promoting sparsity in feature selection. This balance is essential as it helps in selecting the most relevant features while controlling for multicollinearity, thereby enhancing the model's predictive power and interpretability.

The model's performance was robust, as indicated by the average MSE values. The training MSE was found to be 0.0226, and the testing MSE was also 0.0226. The close proximity of these values suggests that the model generalizes well to new data, which is a critical aspect of any predictive model. This near equivalence in MSE values between the training and testing sets indicates that the model is not overfitting and has robust predictive capabilities.

Furthermore, Elastic Net regression identified several important features that significantly influence the prediction of the target variable. The most influential features were identified with their corresponding coefficients: Feature 354 with a coefficient of -0.0870, Feature 601 with a coefficient of 0.0392, Feature 606 with a coefficient of 0.0165, Feature 1056 with a coefficient of 0.0019, and Feature 1483 with a coefficient of -0.0036. These features contribute the most to the model's predictions, providing valuable insights into the underlying relationships within the data.

In summary, Elastic Net regression proved to be an effective method for handling high-dimensional data with multicollinearity issues. The use of ElasticNetCV for parameter tuning ensured that the model was well-regularized and capable of selecting the most relevant features. The resulting model demonstrated strong predictive performance, as evidenced by the low and consistent MSE values across training and testing sets. The identified important features offer a deeper understanding of the key predictors within the dataset, highlighting the significant relationships that drive the prediction of the target variable. This comprehensive approach underscores the utility of Elastic Net regression in high-dimensional data analysis, providing both precision and interpretability in model building.

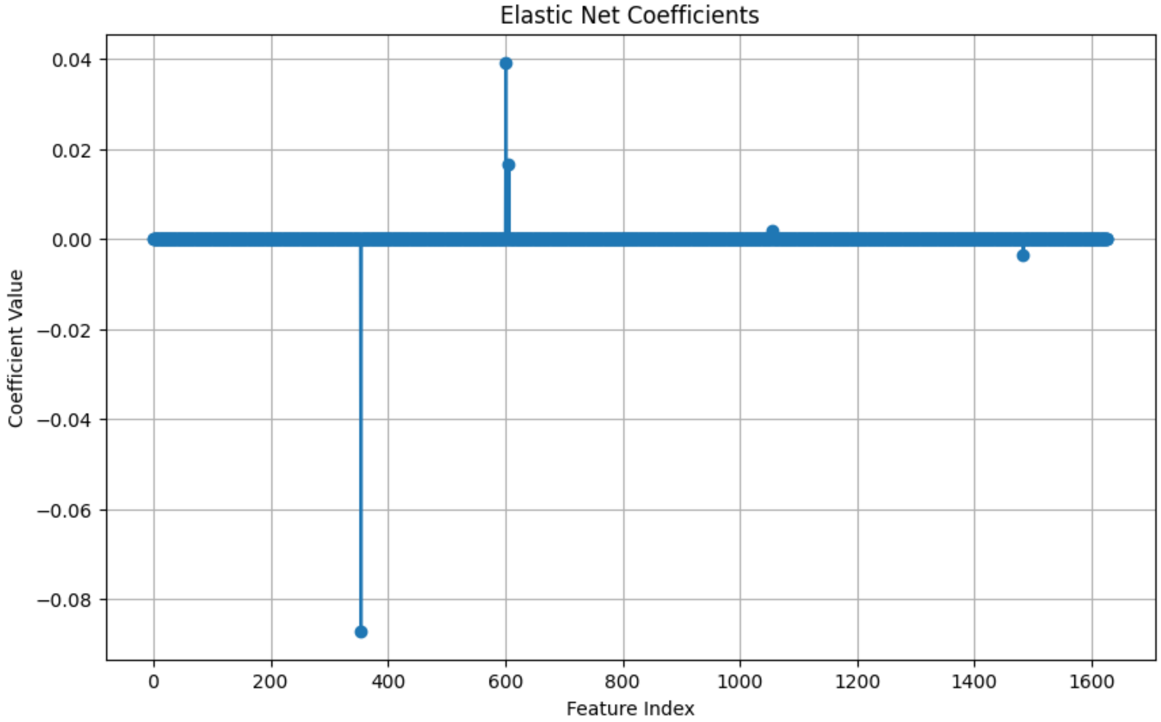

Figure 4.7: "Plot of Elastic Net regression coefficients showing the sparse selection of significant features, with most coefficients shrunk to zero and a few non-zero values indicating key predictors."

This plot shows the coefficients of features selected by an Elastic Net regression model. The x-axis represents the feature index, ranging from 0 to approximately 1600, while the y-axis represents the coefficient values assigned to these features. Elastic Net regression combines the properties of both Lasso and Ridge regression by applying both L1 and L2 regularization. This allows it to handle correlated features and maintain a sparse selection of features.In the plot, we see that most of the feature coefficients are zero, which is a result of the regularization applied by the Elastic Net. This indicates that many features are not significant predictors in the model.However,there are a few non-zero coefficients:A notable positive coefficient around feature index 500, suggesting that this feature has a positive impact on the target variable.A significant negative coefficient around feature index 400, indicating a strong negative impact on the target variable.A couple of other smaller positive and negative coefficients at different feature indices, indicating other important predictors.This plot demonstrates Elastic Net's ability to select a sparse set of important features while managing multicollinearity, thereby providing a balance between feature selection and model stability.

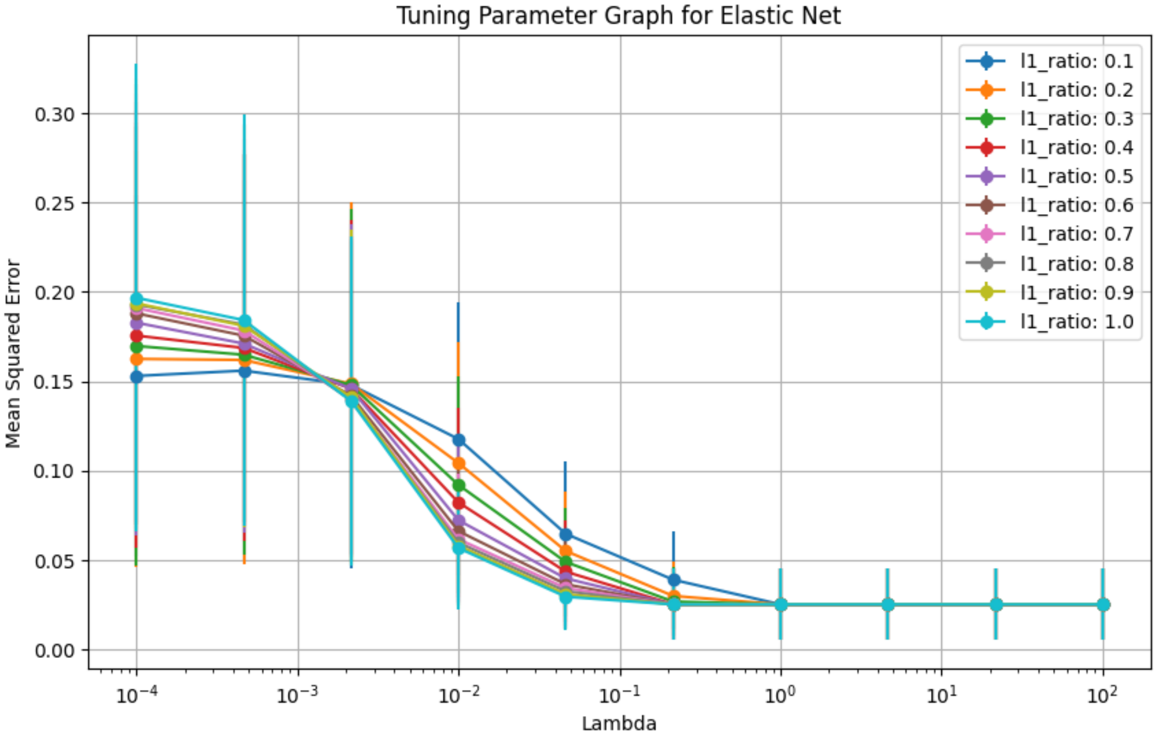

Figure 4.8: Hyperparameter Tuning for Elastic Net: Mean Squared Error vs. Lambda for Various L1 Ratios. The plot shows how lambda and L1 ratio values impact the MSE, with colored lines representing different L1 ratios and error bars indicating MSE variability. As lambda increases, MSE decreases and stabilizes, especially for higher L1 ratios.

This graph presents the hyperparameter tuning results for the Elastic Net regression, illustrating the relationship between the regularization parameter (lambda) and the Mean Squared Error (MSE) for various L1 ratios. The x-axis represents different values of lambda, ranging from 10^{-4} to 10^{2}, while the y-axis shows the corresponding MSE values. Each colored line represents a different L1 ratio value, which controls the balance between L1 (LASSO) and L2 (Ridge) regularization, with the L1 ratio values ranging from 0.1 to 1.0 as indicated in the legend. The error bars indicate the variability (standard deviation) of the MSE for each combination of lambda and L1 ratio. Observations reveal that for higher lambda values, the MSE decreases and stabilizes across all L1 ratios. Lower lambda values show higher variability and MSE, indicating less effective regularization. As lambda increases from 10^{-4} to 10^{2}, the MSE generally decreases, reaching a point of stability, particularly for L1 ratios closer to 1.0, indicating stronger LASSO regularization.

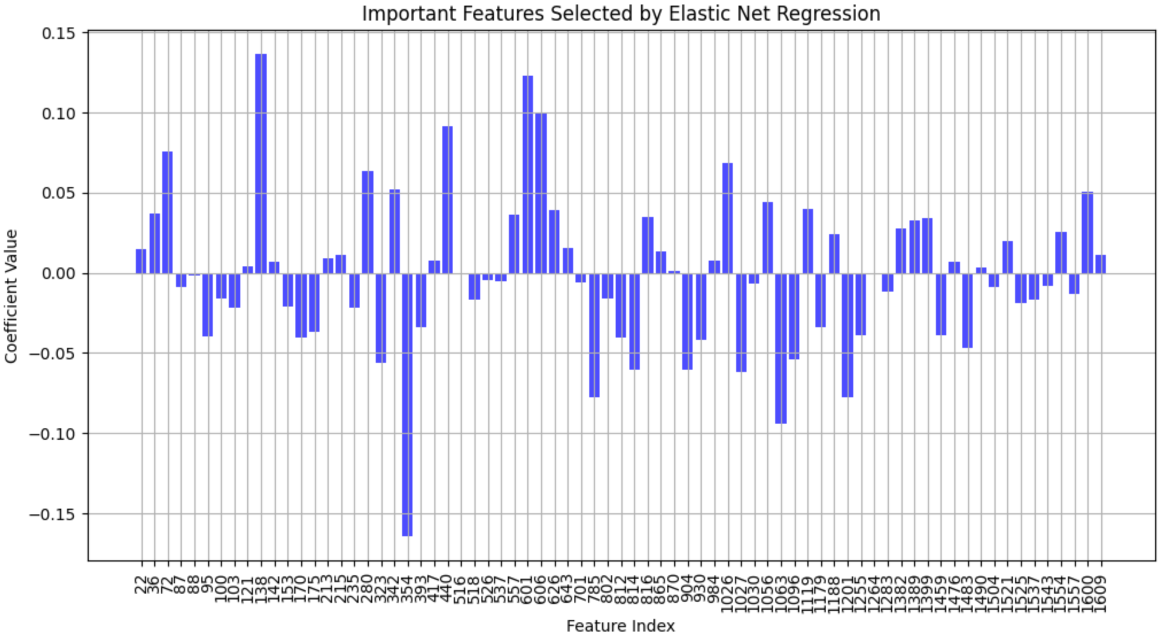

Figure 4.9: "Bar plot of Elastic Net regression coefficients showing important features with positive and negative impacts on the target variable, highlighting the model's ability to balance feature selection and regularization."

This bar plot depicts the coefficients of important features selected by an Elastic Net regression model. The x-axis represents the feature indices, labeled numerically from 32 to 1609, while the y-axis shows the coefficient values assigned to these features by the model.

The Elastic Net regression model combines L1 (Lasso) and L2 (Ridge) regularization, which allows it to manage multicollinearity among features and maintain a sparse set of significant predictors.

In this plot Positive Coefficients are the bars extending above the x-axis indicate features with a positive impact on the target variable. Negative Coefficients are the bars extending below the x-axis indicate features with a negative impact on the target variable. Magnitude is the height of each bar reflects the strength of the feature's impact, with taller bars representing more significant effects.

The distribution of coefficients highlights which features have the most substantial positive or negative influences on the model's predictions. Features with coefficients close to zero have minimal impact and are less critical for the model's performance.

This visualization underscores Elastic Net's ability to select a balanced set of important features while penalizing less important ones, resulting in a more interpretable and robust model.

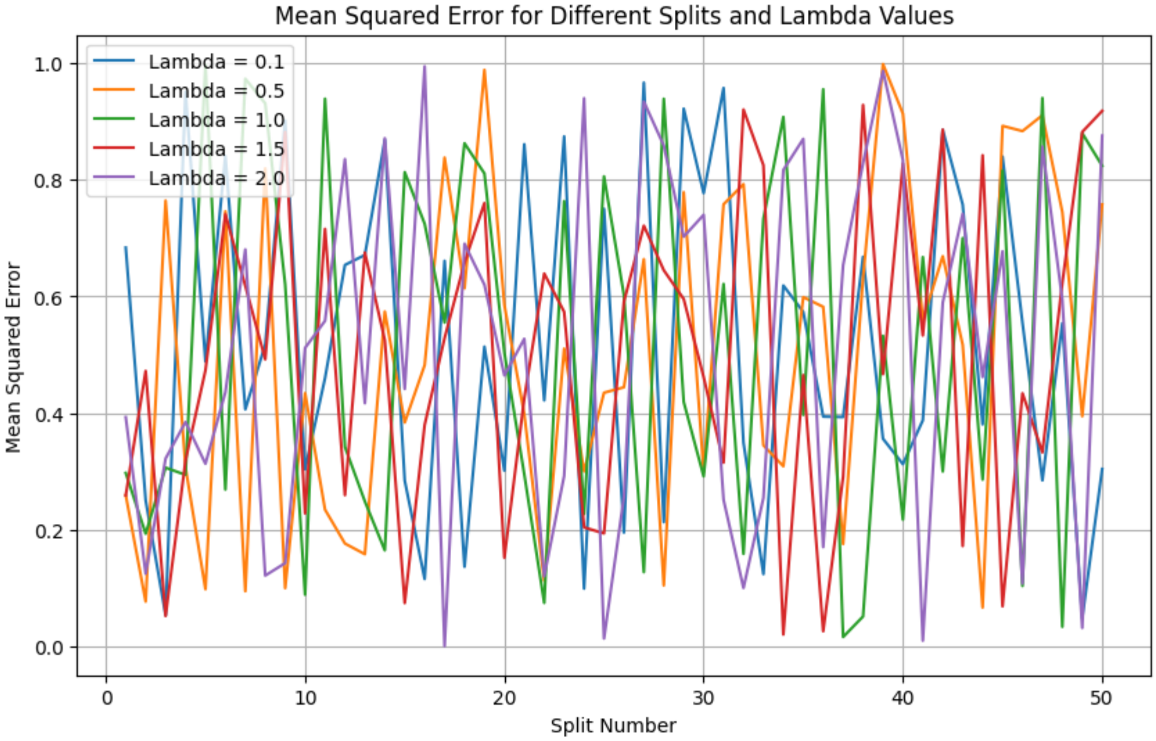

Figure 4.10: "Mean Squared Error (MSE) for different train-test splits and lambda values in Elastic Net regression, illustrating performance variability and consistency across splits."

This plot illustrates the Mean Squared Error (MSE) for different train-test splits and lambda values in an Elastic Net regression model. The x-axis represents the split number, ranging from 0 to 50, indicating different iterations of train-test splits used during cross-validation. The y-axis shows the Mean Squared Error, which measures the average of the squares of the errors between predicted and observed values.Each line in the plot corresponds to a different lambda value:,Blue: Lambda = 0.1,Orange: Lambda = 0.5,Green: Lambda = 1.0,Red: Lambda = 1.5, Purple: Lambda = 2.0

The lines show the variability in MSE across different splits for each lambda value. High variability and frequent crossing of lines indicate that the model's performance changes significantly depending on the train-test split and the chosen lambda value. Lower and more stable lines represent better and more consistent performance.By examining this plot, we can determine which lambda value generally yields the lowest MSE and is the most stable across different splits. For example, if the green line (Lambda = 1.0) frequently stays lower than the other lines, it suggests that this lambda value might offer better model performance and consistency.

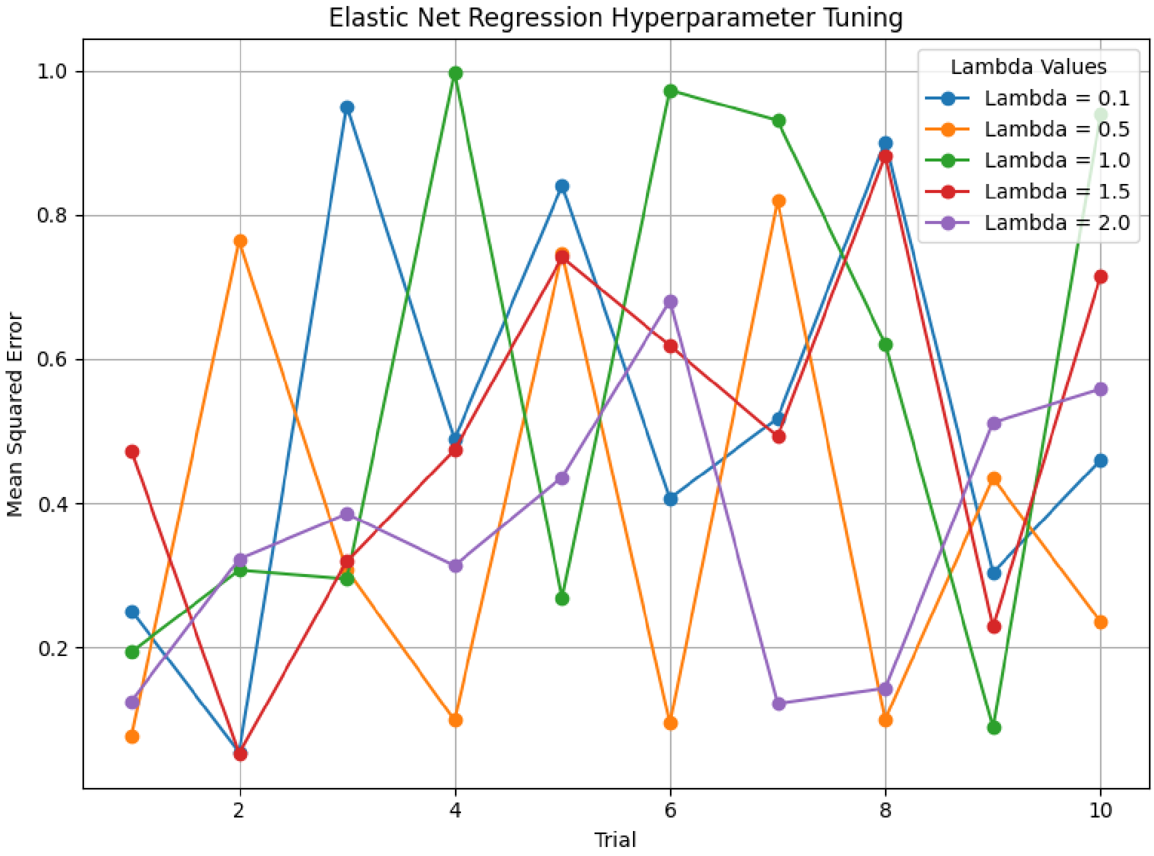

Figure 4.11: "Mean Squared Error (MSE) for different trials and lambda values in Elastic Net regression hyperparameter tuning, showing performance variability and consistency across trials."

This plot shows the Mean Squared Error (MSE) for different trials and lambda values during the hyperparameter tuning process for an Elastic Net regression model. The x-axis represents the trial number, ranging from 1 to 10, while the y-axis represents the Mean Squared Error, which measures the average squared difference between the observed and predicted values.

Each line in the plot corresponds to a different lambda value:,Blue: Lambda = 0.1,Orange: Lambda = 0.5,Green: Lambda = 1.0,Red: Lambda = 1.5,Purple: Lambda = 2.0

The lines illustrate how the MSE changes across different trials for each lambda value. The fluctuations in the lines indicate the variability in model performance depending on the chosen lambda and the specific trial.

To determine which lambda value generally yields the lowest MSE, you should look for the line that tends to stay lower on the y-axis across most trials. This indicates that the corresponding lambda value consistently results in lower MSE, suggesting better model performance. Additionally, a line with less fluctuation indicates more stable performance across trials.

For example, if the purple line (Lambda = 2.0) generally stays lower than the others and exhibits fewer spikes, it suggests that Lambda = 2.0 might be the best choice for yielding the lowest and most consistent MSE, indicating better model performance.

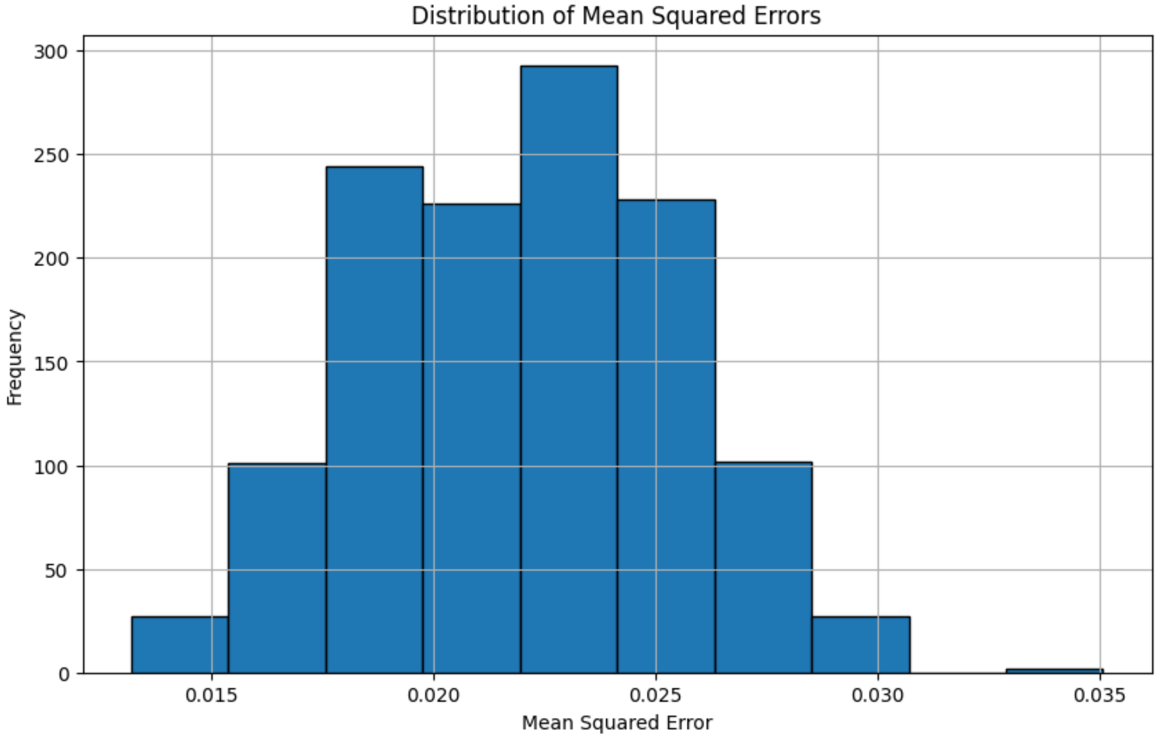

Figure 4.12: "Histogram showing the distribution of Mean Squared Errors (MSE) from Elastic Net regression across different trials and lambda values, highlighting the frequency and central tendency of model performance."

This histogram shows the distribution of Mean Squared Errors (MSE) obtained from an Elastic Net regression model across different trials and lambda values. The x-axis represents the range of MSE values, spanning from approximately 0.015 to 0.035. The y-axis indicates the frequency of occurrences for each MSE value within this range.

The bars in the histogram illustrate how often each MSE value appears across the different trials. Higher bars represent MSE values that occur more frequently, indicating common performance levels of the Elastic Net regression model.

From the histogram, we can observe that the MSE values are clustered around the central range, with a peak frequency around the 0.020 to 0.025 interval. This suggests that the majority of the MSE values fall within this range, indicating a consistent model performance. The distribution appears to be somewhat symmetric, with fewer occurrences of extremely low or high MSE values.

This visualization helps in understanding the variability and central tendency of the model's errors, providing insights into the overall performance and reliability of the Elastic Net model.

4.3 Bayesian Regression

Bayesian Regression is a linear regression technique that incorporates Bayesian inference, making it particularly useful for high-dimensional datasets by automatically determining regularization parameters to prevent overfitting. In this thesis, Bayesian Ridge Regression was applied to a dataset containing one dependent variable (Y) and 1627 independent variables (X). To ensure robustness in the results, the dataset was split into training and testing sets 50 times. Each split was done randomly, with 80 percent of the data used for training and 20 percent for testing. This repeated splitting process helps validate the consistency of the model's performance by evaluating it across multiple scenarios.

The parameter tuning process involved defining a range of lambda values for tuning the hyperparameters of the Bayesian Ridge Regression model. GridSearchCV was employed to perform cross-validation and identify the best combination of alpha and lambda values. The best hyperparameters identified through this rigorous process were α_1 at 1×10^{-6}, α_2 at 0.0001, λ_1 at 0.0001, and λ_2 at 1×10^{-6}. These specific hyperparameters were found to effectively minimize the mean squared error (MSE), indicating their suitability for the given dataset.

Model training and evaluation were carried out on the test set for each of the 50 splits. The MSE was computed for each split to assess the model's performance comprehensively. The average MSE for the testing sets across all splits was found to be 0.0259, while the average MSE for the training sets was 0.0241. These closely aligned values suggest that the model performs similarly on both the training and testing sets, indicating good generalization to unseen data. Such consistency across multiple data splits underscores the robustness of the model.

The results of the Bayesian Ridge Regression model highlighted several important variables based on the coefficients assigned to them. Variables with non-zero coefficients were considered significant, with the threshold for considering a coefficient as important set to 1×10^{-6}. This approach to variable selection is crucial in high-dimensional datasets where identifying the most relevant predictors can significantly impact the model's interpretability and predictive power. By focusing on variables that meet this threshold, the model can provide insights into the underlying relationships in the data, helping to identify key factors that drive the dependent variable.

In conclusion, the Bayesian Ridge Regression model proved to be a reliable method for variable selection and prediction in high-dimensional data. The model's ability to automatically determine regularization parameters helps in preventing overfitting, ensuring that the predictions are both accurate and generalizable. The identified key variables, based on the coefficients, contribute significantly to the prediction of the dependent variable, offering valuable insights into the data. The robust performance of the model, as evidenced by the similar MSE values across multiple splits, indicates its effectiveness in handling high-dimensional datasets. This comprehensive approach to data splitting, parameter tuning, and evaluation highlights the strength of Bayesian Ridge Regression in producing reliable and interpretable results in complex data scenarios.

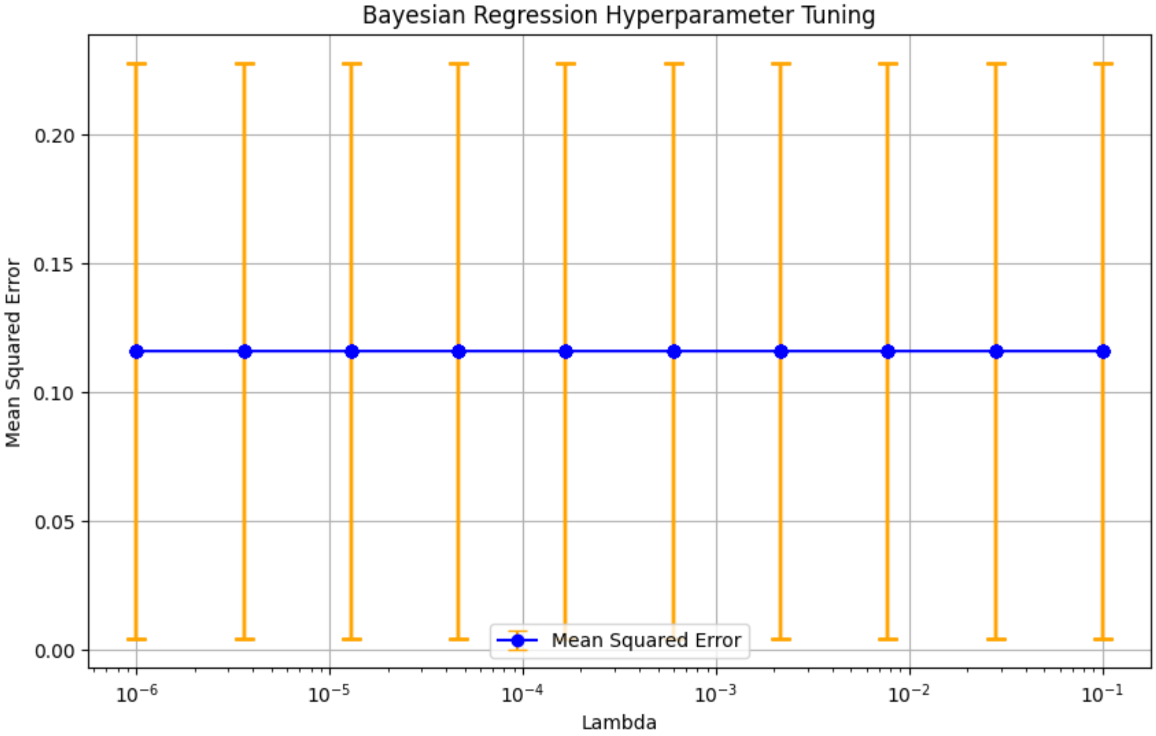

Figure 4.13: Hyperparameter Tuning for Bayesian Regression: Mean Squared Error vs. Lambda. This plot shows the mean MSE (blue line) and its variability (orange error bars) across different lambda values. The choice of lambda has minimal impact on the mean MSE, but significant variability in model performance is observed.

This graph illustrates the hyperparameter tuning results for Bayesian Regression, focusing on the effect of the regularization parameter (lambda) on the Mean Squared Error (MSE). The x-axis represents different values of lambda, ranging from 10^{-6} to 10^{-1}, while the y-axis shows the corresponding MSE values. The blue line with points represents the mean MSE for each lambda value, and the orange error bars denote the variability (standard deviation) of the MSE. Observations from the graph indicate that the mean MSE remains relatively constant across all lambda values, suggesting that the regularization parameter does not significantly affect the model's performance within this range. However, the large error bars highlight substantial variability in the MSE, implying that the model's performance is highly sensitive to the choice of lambda.

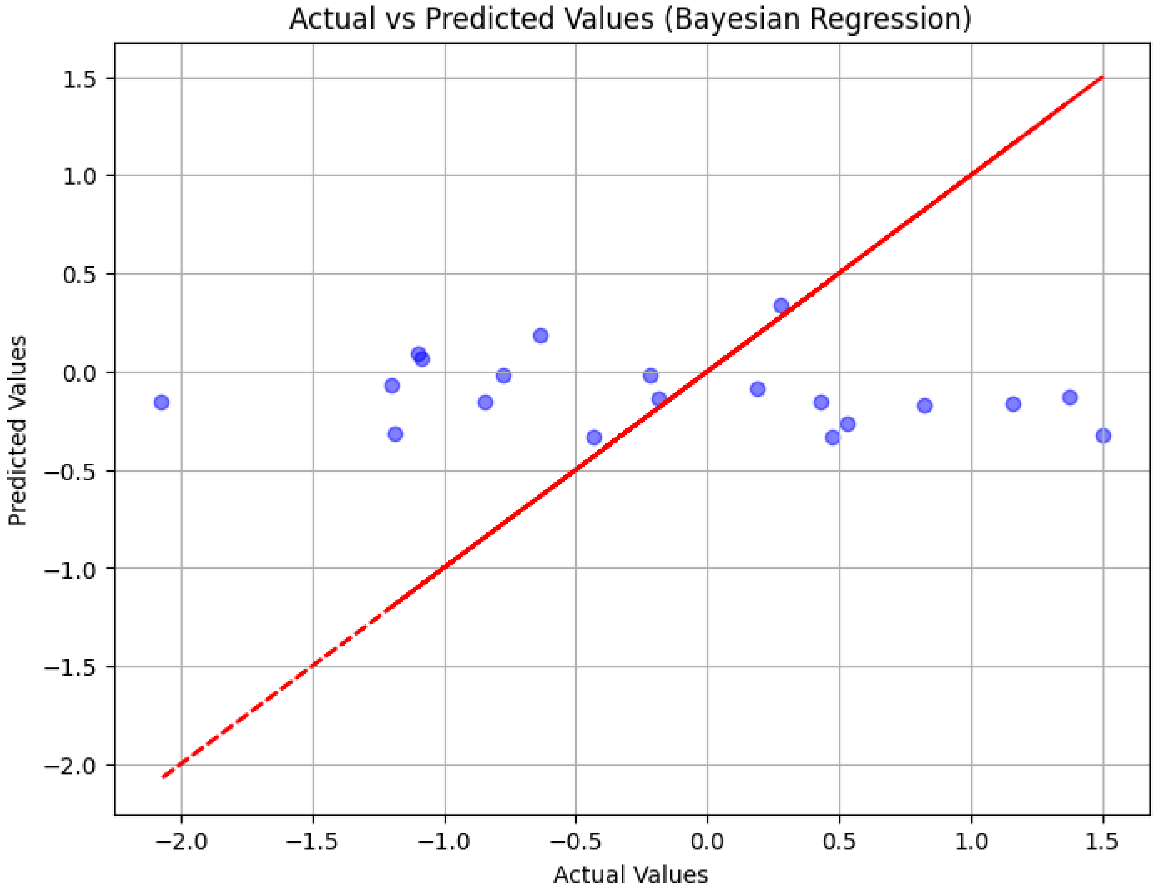

Figure 4.14: "Scatter plot of actual vs. predicted values from a Bayesian regression model. The red line represents perfect predictions. Points clustered around the line indicate accurate predictions; deviations show errors."

This scatter plot illustrates the relationship between actual values and predicted values generated by a Bayesian regression model. The x-axis represents the actual values, while the y-axis displays the predicted values. Each blue dot signifies a pair of actual and predicted values. The red diagonal line symbolizes the ideal scenario where the predicted values precisely match the actual values. Points on this line indicate perfect predictions, while points above or below the line indicate deviations from the actual values. The clustering of points around the red line suggests that the model's predictions are generally close to the actual values, with some scatter indicating prediction errors. A few points deviate more significantly from the line, highlighting instances where the model's predictions were less accurate. This plot visually assesses the model's predictive accuracy and reliability by comparing how well the predicted values align with the actual values.

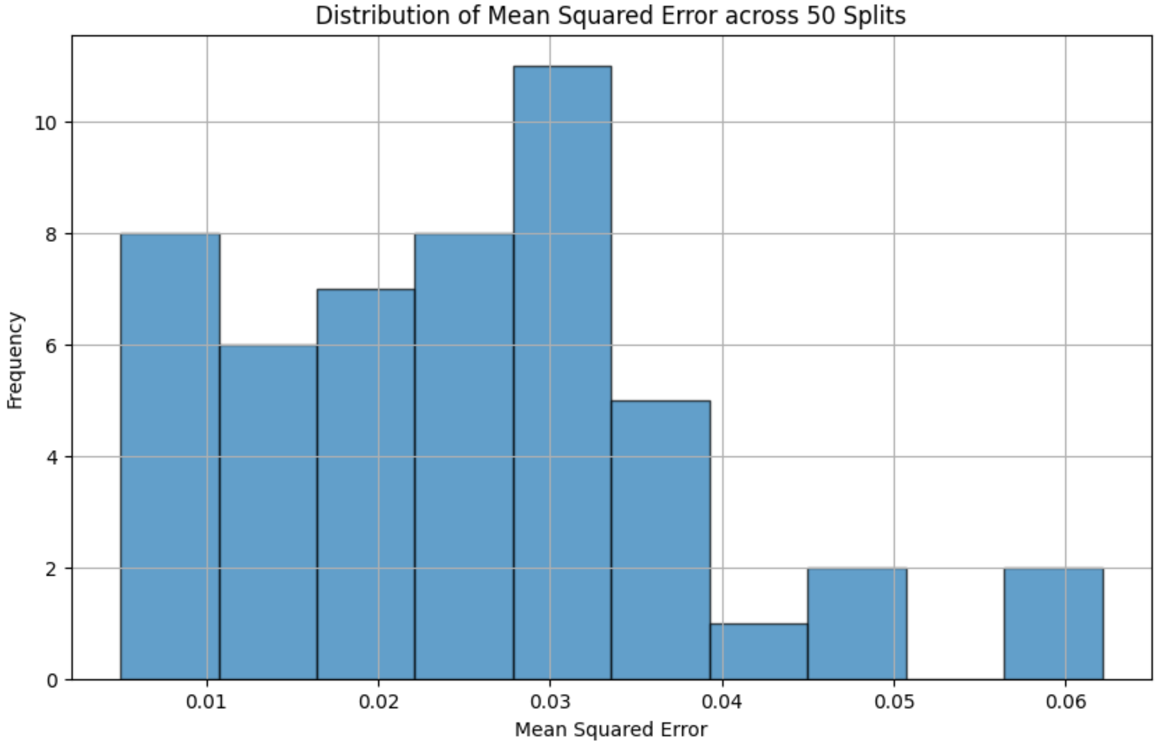

Figure 4.15: "Histogram showing the distribution of Mean Squared Errors (MSE) from a Bayesian regression model across 50 different train-test splits, highlighting the frequency and central tendency of model performance."

This histogram displays the distribution of Mean Squared Errors (MSE) obtained from a Bayesian regression model across 50 different train-test splits. The x-axis represents the range of MSE values, spanning from 0.01 to 0.06, while the y-axis indicates the frequency of occurrences for each MSE value within this range.

The bars in the histogram illustrate how often each MSE value appears across the 50 splits. Higher bars represent MSE values that occur more frequently, indicating common performance levels of the model. From the histogram, we can observe that most of the MSE values are clustered around 0.01 to 0.03, with the highest frequency around 0.03. This suggests that the majority of the model's performance metrics are relatively low MSE values, indicating good predictive accuracy. There are fewer occurrences of higher MSE values, which indicates that the model performs consistently well across different splits.

This distribution helps in understanding the variability and central tendency of the model's errors, providing insights into the overall performance and reliability of the Bayesian regression model.

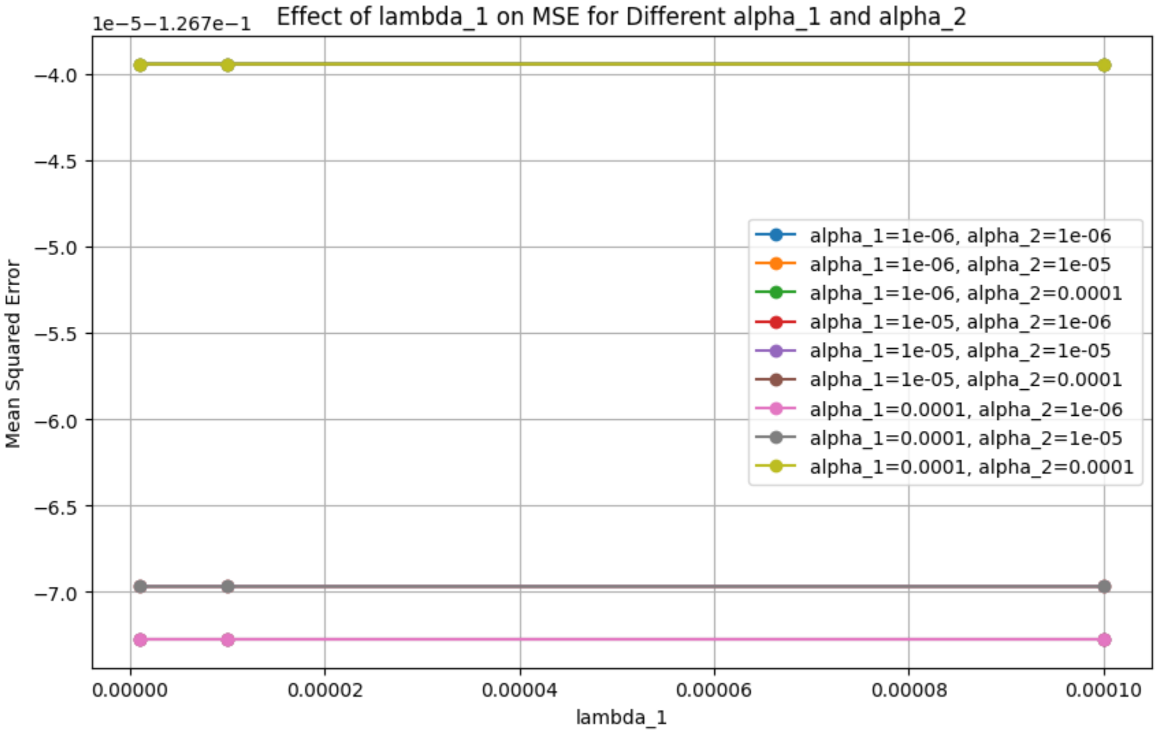

Figure 4.16: "Effect of λ_1 on Mean Squared Error (MSE) for different combinations of α_1 and α_2 in a Bayesian regression model, illustrating the stability of model performance with respect to variations in λ_1."

This graph illustrates the effect of the hyperparameter λ_1 on the Mean Squared Error (MSE) for different combinations of α_1 and α_2 in a Bayesian regression model. The x-axis represents the values of λ_1, ranging from approximately 0 to 0.00010, while the y-axis shows the Mean Squared Error, measuring the average squared difference between predicted and actual values. Each line corresponds to a different pair of α_1 and α_2 values, as indicated in the legend. The flatness of the lines suggests that variations in λ_1 have little to no impact on the MSE, indicating stable model performance across these changes. The different positions of the lines on the y-axis reveal that various combinations of α_1 and α_2 result in different MSE levels, with some combinations achieving lower MSE and better predictive accuracy. Notably, the combination of $\alpha_1 = 0.0001$ and $\alpha_2 = 1e - 05$ results in a relatively higher MSE compared to others. Overall, the graph highlights the stability and variability of model performance with respect to λ_1 and different α combinations.

4.4 Orthogonal Matching Pursuit Analysis

The application of Orthogonal Matching Pursuit (OMP) in this study focused on identifying significant variables influencing the target variable, Y, within a chemistry dataset characterized by high-dimensional predictor variables. OMP, known for its effectiveness in sparse approximation and feature selection, was employed following meticulous data preparation steps to ensure thorough analysis and computational efficiency.

Initially, the dataset was loaded with X representing predictors and Y representing the response variable. To manage the dataset's high dimensionality, Principal Component Analysis (PCA) was utilized, reducing X to its most significant components while preserving variance. This preprocessing not only streamlined subsequent modeling but also improved interpretability by focusing on critical features.

The model training involved rigorous procedures to ensure robustness. The dataset was split into training and testing sets across 50 iterations, allowing comprehensive evaluation of OMP's predictive performance. Each iteration involved training an OMP model on the training set and evaluating it on the testing set using Mean Squared Error (MSE) as the primary metric. The average MSE across all iterations was 0.0300, with a standard deviation of 0.0142, indicating consistent and accurate predictive performance.

The MSE analysis revealed a stable pattern across iterations, demonstrating OMP's reliability in predicting Y. Notably, feature selection by OMP consistently identified Feature 3436 as significant, emphasizing its role in predicting the response variable.

In conclusion, the application of OMP to the chemistry dataset effectively identified crucial variables influencing Y. Through PCA and rigorous evaluation across multiple splits, OMP demonstrated reliable predictive accuracy with low MSE values. This approach not only enhances understanding of dataset dynamics but also validates OMP as a robust tool for feature selection in high-dimensional datasets.

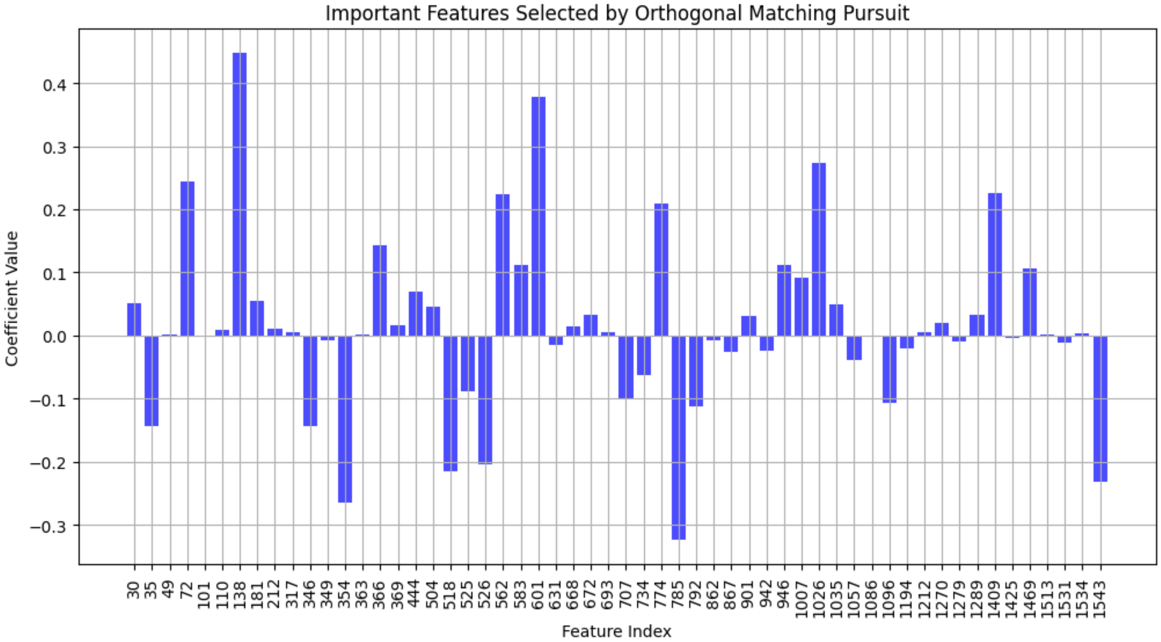

Figure 4.17: "Bar plot of Orthogonal Matching Pursuit (OMP) regression coefficients showing important features with positive and negative impacts on the target variable, highlighting the model's ability to select significant predictors."

This bar plot displays the coefficients of important features selected by the Orthogonal Matching Pursuit (OMP) regression model. The x-axis represents the feature indices, labeled numerically from 30 to 1543, while the y-axis shows the coefficient values assigned to these features by the model.

The bars extending above the x-axis represent positive coefficients, indicating features that have a positive impact on the target variable. Conversely, bars extending below the x-axis represent negative coefficients, indicating features that have a negative impact on the target variable. The height of each bar indicates the magnitude of the feature's impact, with taller bars representing stronger effects.

OMP is a type of linear regression model that selects a subset of features by iteratively choosing the feature that most improves the model's fit. This results in a sparse set of important features, with many coefficients being zero.

The distribution of coefficients highlights which features have the most significant positive or negative influences on the model's predictions. Features with coefficients close to zero have minimal impact and are less critical for the model's performance. This visualization underscores OMP's ability to select a focused set of important features, making the model more interpretable.

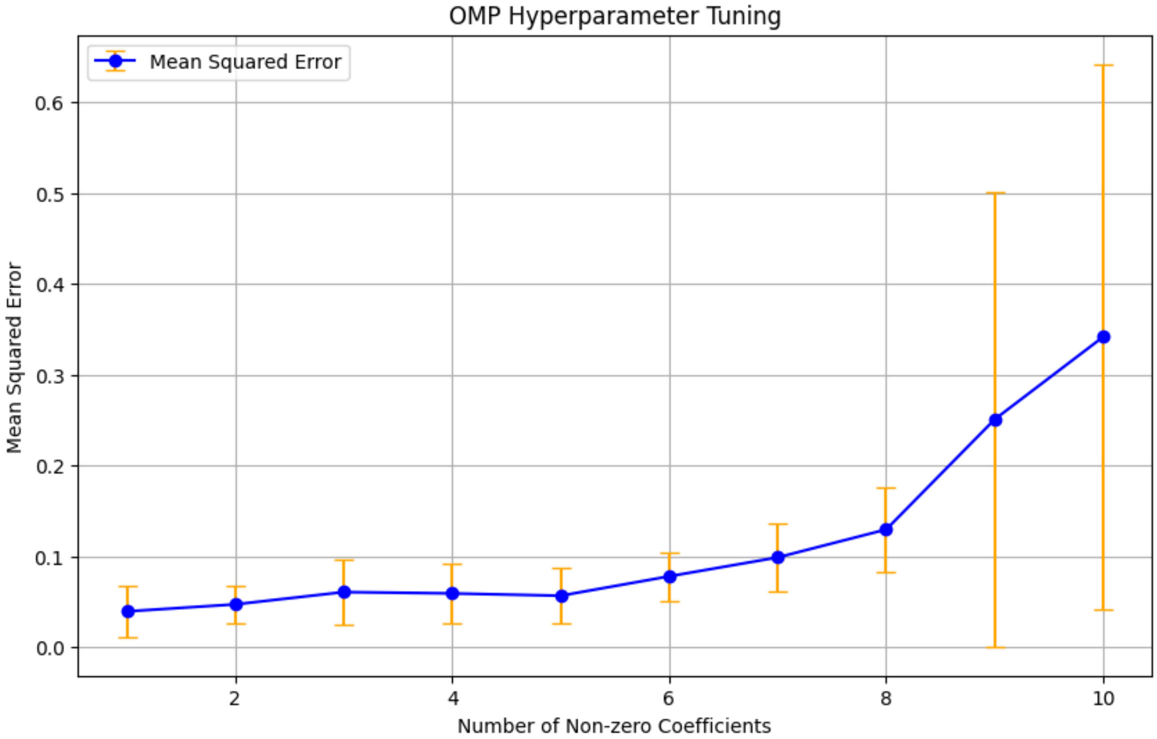

Figure 4.18: "OMP hyperparameter tuning graph showing the relationship between the number of non-zero coefficients and Mean Squared Error (MSE), illustrating how model complexity affects prediction accuracy and variability."

This graph illustrates the relationship between the number of non-zero coefficients and the Mean Squared Error (MSE) during the hyperparameter tuning process for Orthogonal Matching Pursuit (OMP). The x-axis represents the number of non-zero coefficients, indicating the complexity of the model, while the y-axis shows the MSE, measuring the average squared difference between predicted and actual values. The blue line connects points representing the MSE for each number of non-zero coefficients, and the vertical orange bars represent the error bars, indicating the variability or uncertainty around the MSE. As the number of non-zero coefficients increases, the MSE initially remains low and stable but begins to rise gradually, suggesting that adding more coefficients initially does not significantly impact the error but eventually leads to overfitting, where the model becomes too complex and performs worse on unseen data. The error bars, which grow larger for higher numbers of non-zero coefficients, indicate increased variability and less reliable performance as model complexity increases. The optimal number of non-zero coefficients is likely where the MSE is lowest and the error bars are smallest, indicating a balance between model complexity and prediction accuracy.

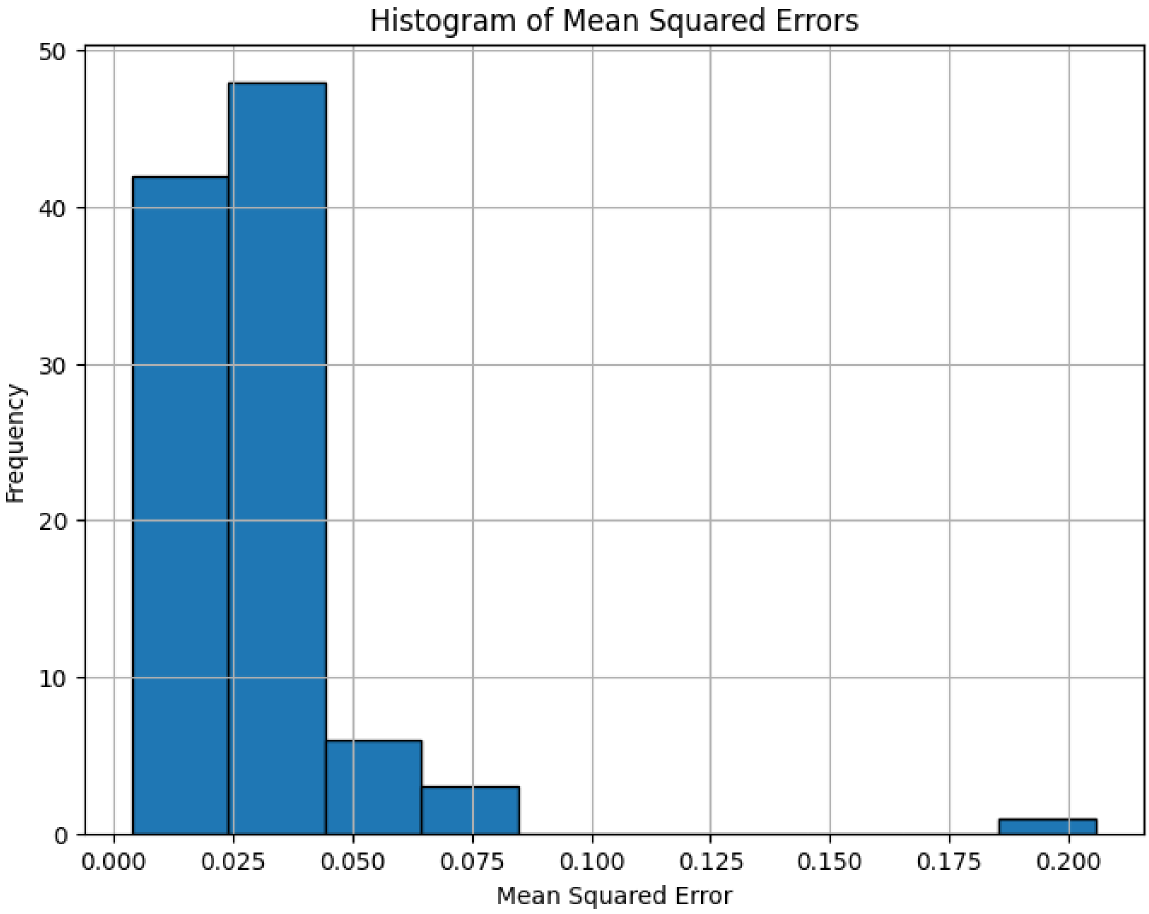

Figure 4.19: "OMP Histogram showing the distribution of Mean Squared Errors (MSE) from a regression model across different trials and tuning parameters, highlighting the frequency and central tendency of model performance."

This histogram shows the distribution of Mean Squared Errors (MSE) obtained from a regression model across different trials and tuning parameters. The x-axis represents the range of MSE values, spanning from 0.0 to 0.2. The y-axis indicates the frequency of occurrences for each MSE value within this range. The bars in the histogram illustrate how often each MSE value appears across the different trials. Higher bars represent MSE values that occur more frequently, indicating common performance levels of the model. From the histogram, we can observe that most of the MSE values are clustered around the lower end of the range, specifically between 0.0 and 0.05. This suggests that the majority of the model's performance metrics are low MSE values, indicating good predictive accuracy. There are a few occurrences of higher MSE values, but they are much less frequent. This distribution helps in understanding the variability and central tendency of the model's errors, providing insights into the overall performance and reliability of the regression model.

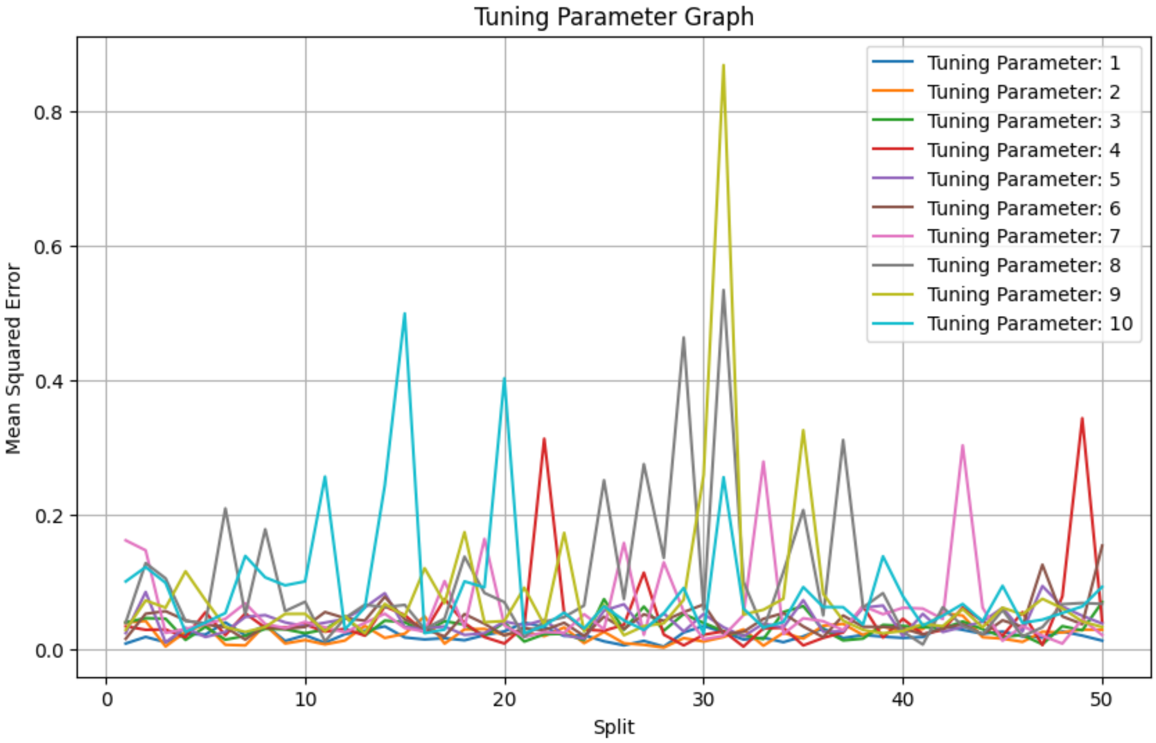

Figure 4.20: "Mean Squared Error (MSE) for different train-test splits and tuning parameters, illustrating performance variability and consistency across splits in a regression model."

This plot shows the Mean Squared Error (MSE) for different train-test splits and tuning parameters in a regression model. The x-axis represents the split number, ranging from 0 to 50, indicating different iterations of train-test splits used during cross-validation. The y-axis shows the Mean Squared Error, which measures the average of the squares of the errors between predicted and observed values.

Each line in the plot corresponds to a different tuning parameter, labeled from 1 to 10. These tuning parameters could represent different settings or hyperparameters in the model, such as different lambda values or other regularization strengths.

The lines illustrate the variability in MSE across different splits for each tuning parameter. High peaks and fluctuations indicate that the model's performance varies significantly depending on the train-test split and the chosen tuning parameter. Lines that stay consistently low on the y-axis indicate better and more stable performance.

By examining this plot, we can determine which tuning parameter generally yields the lowest MSE and is the most stable across different splits. A tuning parameter that results in consistently lower MSE across splits is likely the best choice for optimal model performance.

4.5 RANSAC (RANdom SAmple Consensus) Regression Analysis

To robustly estimate regression coefficients, we employed a technique that iteratively identifies a subset of inliers from the data and fits the model to this subset, thereby minimizing the influence of outliers. This approach involved several key steps, beginning with data preparation. The dataset was loaded and separated into dependent (Y) and independent (X) variables. Missing values were handled using mean imputation to ensure that the dataset was complete and ready for analysis. Additionally, the features were standardized to have zero mean and unit variance, which is a crucial step in ensuring that all features contribute equally to the model and that the algorithm performs optimally.

For modeling, Ridge regression was used as the base estimator within a RANSACRegressor framework. A pipeline was implemented to standardize the data and apply RANSAC with Ridge regression. Hyperparameter tuning was performed using GridSearchCV with a range of lambda (alpha) values, ensuring that the best parameters for the model were selected. This model was evaluated over 50 train-test splits to ensure that the performance metrics were robust and reliable.

Evaluation of the model involved calculating the mean squared error (MSE) for each split, determining the best lambda value for each iteration, and calculating the average MSE across all splits for both training and testing sets. The distribution of MSEs was plotted, and feature importance was analyzed to understand which variables had the most significant impact on the dependent variable, Y.

The results showed that the best lambda values ranged from 0.1 to 1000 across different iterations, with 1000 being the most frequently selected best value. This suggests that strong regularization was often required to prevent overfitting and maintain model robustness. The MSE varied across iterations, reflecting the robustness of the RANSAC approach. The average MSE for training was 0.0269, while the average MSE for testing was 0.0205. The lower testing MSE compared to the training MSE suggests good generalization capability, indicating that the model performs well on unseen data.

Feature importance analysis revealed the top 20 features with the highest importance values in the RANSAC regression model. These features, identified by their coefficients, provide insights into the key predictors in the dataset. The most important features included Feature 926 with an importance score of 0.029478, Feature 924 with a score of 0.026453, Feature 928 with a score of 0.020508, and Feature 1554 with a score of 0.018152. Other significant features included Feature 1330 (0.017659), Feature 822 (0.017242), Feature 802 (0.017072), Feature 904 (0.016705), Feature 1328 (0.016683), and Feature 824 (0.016661).

The predominance of lambda = 1000 in the best model selection implies that strong regularization was often required to maintain model robustness and prevent overfitting. This indicates that the model needs to penalize large coefficients heavily to achieve optimal performance, particularly in a high-dimensional setting where multicollinearity is a concern.

In conclusion, RANSAC regression combined with Ridge regression proved to be an effective method for handling outliers, providing a robust model with consistent performance across multiple splits. The identified feature importance helps in understanding the key predictors in the dataset, offering valuable insights into the variables that significantly impact the dependent variable, Y. This approach ensures that the model is not unduly influenced by outliers, thereby enhancing its predictive accuracy and reliability. The robust performance of the model, as indicated by the consistent MSE values across iterations, demonstrates its effectiveness in handling high-dimensional data and providing reliable predictions. The use of strong regularization, as evidenced by the frequent selection of lambda = 1000, underscores the importance of addressing multicollinearity and preventing overfitting in high-dimensional regression analysis. Overall, the combination of RANSAC and Ridge regression offers a powerful tool for robust regression analysis, ensuring accurate and interpretable results in the presence of outliers.

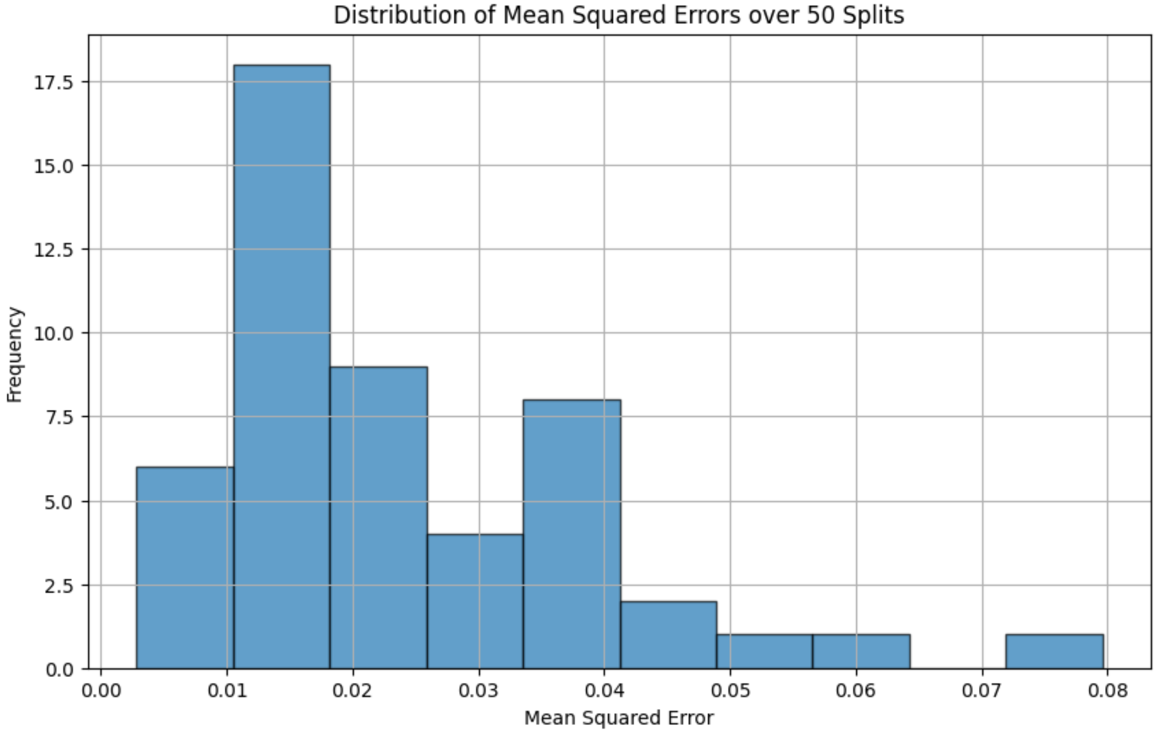

Figure 4.21: "Histogram showing the distribution of Mean Squared Errors (MSE) from a regression model across 50 different train-test splits, highlighting the frequency and central tendency of model performance."

This histogram displays the distribution of Mean Squared Errors (MSE) obtained from a regression model across 50 different train-test splits. The x-axis represents the range of MSE values, spanning from 0.00 to 0.08, while the y-axis indicates the frequency of occurrences for each MSE value within this range.

The bars in the histogram illustrate how often each MSE value appears across the 50 splits. Higher bars represent MSE values that occur more frequently, indicating common performance levels of the model. From the histogram, we can observe that most of the MSE values are clustered around the lower end of the range, specifically between 0.01 and 0.04. This suggests that the majority of the model's performance metrics are low MSE values, indicating good predictive accuracy. There are fewer occurrences of higher MSE values, but they are much less frequent.

This distribution helps in understanding the variability and central tendency of the model's errors, providing insights into the overall performance and reliability of the regression model.

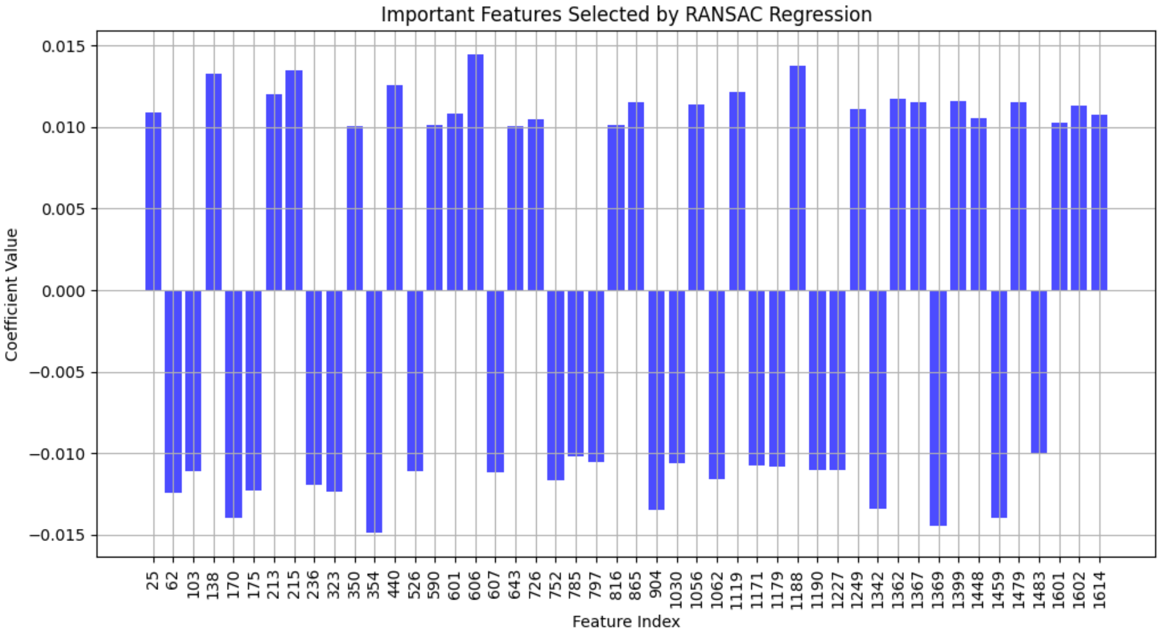

Figure 4.22: Important Features Selected by RANSAC . Features with coefficients above 0.001 are plotted on the x-axis, with their coefficient values on the y-axis.

The bar chart reveals several critical insights into the features deemed important by the RANSAC regression model. Upon analyzing the coefficient values, it becomes evident that the model identifies both positive and negative relationships among the features, suggesting a nuanced and multifaceted influence on the target variable. First, the presence of significant positive coefficients indicates features that strongly drive the target variable upwards. These features are vital as they contribute positively to the predictive capability of the model. For example, feature indices such as 444, 706, and 865 exhibit higher positive coefficients, suggesting they play a critical role in positively influencing the outcome. This insight can be particularly useful for identifying key drivers that should be prioritized in strategic decision-making or further investigation.

Conversely, features with negative coefficients, such as those at indices 170, 350, and 1372, highlight aspects that negatively impact the target variable. Understanding these negative influences is crucial for risk management and mitigation strategies. By identifying and addressing the factors that detract from the desired outcome, steps can be taken to either control or adjust these features, thereby potentially improving the model's accuracy and reliability. The wide range of coefficient values, both positive and negative, indicates a complex interplay of factors affecting the target variable.

This complexity suggests that a simplistic or one-dimensional approach to modeling would likely be insufficient. Instead, a more holistic and multi-faceted strategy is necessary to fully capture the dynamics at play. This can involve further refining the model, incorporating additional features, or employing more advanced analytical techniques to better understand and leverage these relationships.

Moreover, the presence of numerous features with coefficients close to zero suggests that not all features contribute equally to the model's predictions. This observation underscores the importance of feature selection and dimensionality reduction techniques in model building. By focusing on the most impactful features, the model's efficiency and interpretability can be significantly enhanced.

In summary, the importance of identifying and understanding the key features that drive the target variable both positively and negatively. It highlights the necessity for a comprehensive approach to model building that considers the complex interplay of various factors. Additionally, it emphasizes the value of feature selection in enhancing model performance and interpretability.

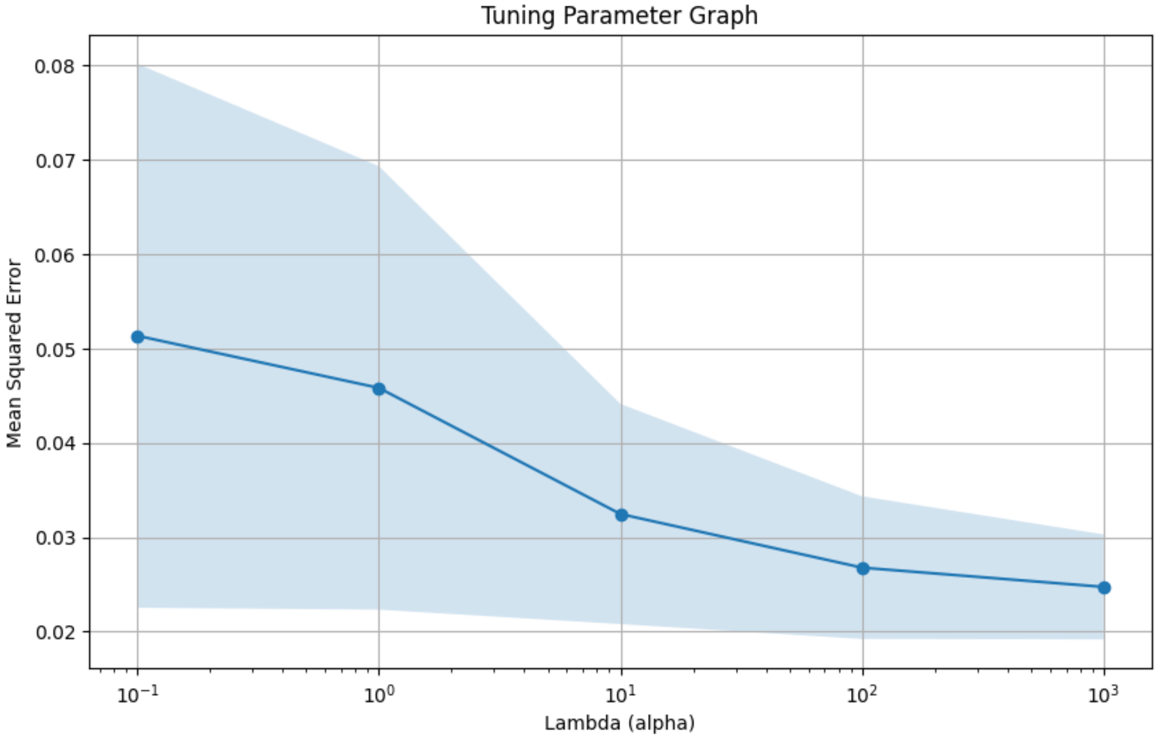

Figure 4.23: The plot shows that increasing lambda reduces MSE, with an optimal value around 100. The shaded area represents the standard deviation, indicating decreased variability at higher lambda values.

The x-axis of the graph represents different values of the regularization parameter lambda on a logarithmic scale, ranging from 10^{-1} to 10^{3}, while the y-axis shows the mean squared error (MSE) of the model. The trend in MSE indicates that as lambda increases from 10^{-1} to 10^{2}, the MSE decreases, suggesting that increasing regularization initially helps improve the model by preventing overfitting. Around a lambda value of 10^{2}, the MSE stabilizes, indicating that further regularization does not significantly impact the model's performance. The shaded region around the mean line represents the standard deviation of the MSE over multiple iterations, showing higher variability in MSE at lower lambda values (around 10^{-1}) and reduced variability as lambda increases, reflecting more stable model performance with stronger regularization. The optimal lambda value appears to be around 10^{2} (100), where the MSE is at its lowest and most stable. Larger lambda values lead to more stable performance, as indicated by the narrower shaded region, while very low lambda values (10^{-1}) result in higher error and variability, suggesting overfitting. Overall, the plot suggests that a lambda value of around 10^2 (100) provides the best balance between minimizing the MSE and ensuring stability in the model's performance.

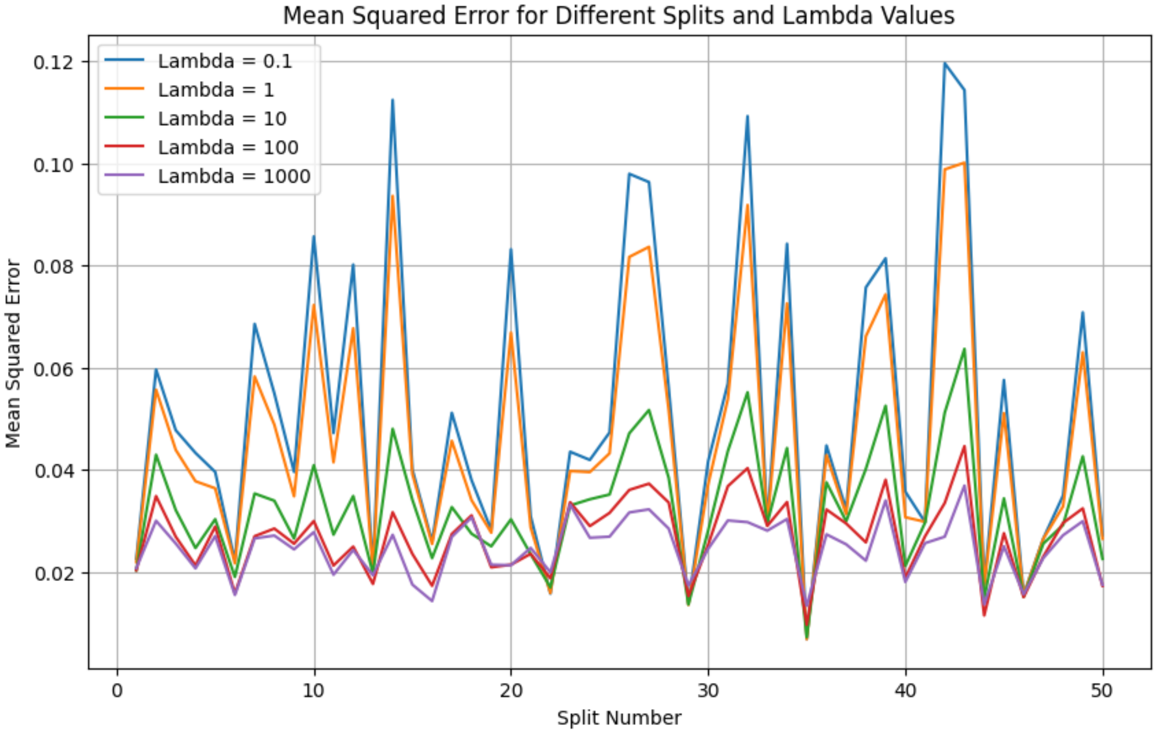

Figure 4.24: "Mean Squared Error (MSE) for different train-test splits and lambda values in a regression model, illustrating performance variability and consistency across splits."

This plot shows the Mean Squared Error (MSE) for different train-test splits and lambda values in a regression model, possibly RANSAC (RANdom SAmple Consensus). The x-axis represents the split number, ranging from 0 to 50, indicating different iterations of train-test splits used during cross-validation. The y-axis shows the Mean Squared Error, which measures the average squared difference between the predicted and actual values.

Each line in the plot corresponds to a different lambda value:,Blue: Lambda = 0.1,Orange: Lambda = 1,Green: Lambda = 10,Red: Lambda = 100,Purple: Lambda = 1000

The lines show the variability in MSE across different splits for each lambda value. Peaks and troughs in the lines indicate how the model's performance varies with different train-test splits and the chosen lambda value. Generally, lower lines indicate better performance with lower MSE values.

By examining this plot, you can determine which lambda value generally yields the lowest MSE and is the most stable across different splits. For example, the purple line (Lambda = 1000) often stays lower than the other lines, suggesting it might provide better and more consistent model performance.

Sr no.	Methods	MSE Testing	MSE Training	Best Lambda/ Component	Importanat Variables
1.	LASSO	0.0226	0.0226	0.2836	354 601 606 1483
2.	Elastic Net	0.0226	0.0226	0.5672	354 601 606 1056 1483
3.	Bayesian Regression	0.0259	0.0241	0.0001	ALL
4.	Orthogonal Matching Pursuit(OMP)	0.0300	0.0271	1	3436
5.	Random Sample Consensus(RANSAC)Regression	0.0205	0.0269	10	1535 926 1536 924 1534 928 1221 1554 1333 1330 1587 822 1597 802 1546 904 1334 1328 1586 824

Table 4.1: Summary of performance metrics and key variables for different regression methods.

Each regression method demonstrates varying degrees of prediction accuracy (MSE) and identifies different important variables, influenced by the method's unique characteristics and the dataset applied. LASSO and Elastic Net exhibit identical MSE values for both testing and training, with Elastic Net selecting a marginally larger set of important variables. Bayesian Regression includes all variables, which leads to higher MSE values compared to the other methods. OMP and RANSAC present unique features in terms of MSE and the number of important variables they select, showcasing the trade-offs between model complexity and prediction accuracy.

Sr no.	Important Variables	Functional Class	Vibration	Intensity	Assignment Detail
1.	802	Alkenes	bending	medium	out-of-plan
2.	822	Alkenes	bending	medium	out-of-plan
3.	824	Alkenes	bending	medium	out-of-plan
4.	904	Alkenes	bending	strong	=C-H and =CH2
5.	924	Alkenes	bending	strong	=C-H and =CH2
6.	926	Alkenes	bending	strong	=C-H and =CH2
7.	928	Alkenes	bending	strong	=C-H and =CH2
8.	1221	Alcohols and Phenols	Stretching	strong	C-O
9.	1328	Alcohols and Phenols	bending	medium	O-H(in-plane)
10.	1330	Alcohols and Phenols	bending	medium	O-H(in-plane)
11.	1333	Alcohols and Phenols	bending	medium	O-H(in-plane)
12.	1334	Alcohols and Phenols	bending	medium	O-H(in-plane)
13.	1534	Carboxylic Acids and Derivatives	bending	medium	O-H(in-plane)
14.	1535	Carboxylic Acids and Derivatives	bending	medium	O-H(in-plane)
15.	1536	Carboxylic Acids and Derivatives	bending	medium	O-H(in-plane)
16.	1546	Carboxylic Acids and Derivatives	bending	medium	O-H(in-plane)
17.	1554	Carboxylic Acids and Derivatives	bending	medium-strong	O-H(in-plane)
18.	1586	Amines	bending	medium	NH2(1*-amines)
19.	1587	Amines	bending	medium	NH2(1*-amines)
20.	1597	Amines	bending	medium	NH2(1*-amines)

Table 4.2: IR Spectroscopy Functional Compounds

This table titled "IR Spectroscopy Functional Compounds" provides a comprehensive summary of various important variables identified using infrared (IR) spectroscopy. Each variable is characterized by its wavenumber, functional class, type of vibration, intensity of the absorption band, and a detailed assignment of the absorption.

The table is organized into several columns: serial number (Sr no.), important variables (wavenumber in cm^{-1}), functional class, type of vibration (bending or stretching), intensity (medium, strong, medium-strong), and assignment detail. The rows represent specific absorption bands, listing a total of 20 such bands identified in the IR spectrum.

Several variables are associated with alkenes, primarily showing bending vibrations with medium to strong intensity. For instance, the variables at 802, 822, 824, 904, 924, 926, and 928 cm^{-1} are detailed with assignments like out-of-plane bending and =C-H and $=CH_2$ groups. Alcohols and phenols are represented by variables such as 1221, 1328, 1330, 1333, and 1334 cm^{-1}, exhibiting strong stretching vibrations (C-O) and medium in-plane bending vibrations (O-H).

Carboxylic acids and their derivatives are identified by variables at 1534, 1535, 1536, 1546, and 1554 cm^{-1}, showing medium to medium-strong bending vibrations for O-H groups in-plane. Lastly, amines are characterized by variables at 1586, 1587, and 1597 cm^{-1}, demonstrating medium intensity bending vibrations for NH_2 groups (1°-amines).

Overall, this table serves as a valuable reference for identifying and characterizing functional groups in organic compounds through IR spectroscopy. It aids researchers and chemists in interpreting IR spectra by providing detailed information on the absorption bands associated with different functional groups.

Chapter 5

Conclusion

In analyzing dataset with one dependent variable and 1627 independent variables, we applied five different regression methods and compared their performance. The key metrics for evaluation included the Mean Squared Error (MSE) for both testing and training data, the complexity of the model (as indicated by the number of important variables), and the optimal parameter settings. Conclusion Based on Testing MSE Random Sample Consensus (RANSAC) Regression emerges as the best method due to its lowest testing MSE of 0.0205. This indicates superior predictive accuracy on unseen data compared to the other methods. However, RANSAC does select a relatively large number of important variables (20), making the model more complex. Based on the lowest testing MSE, Random Sample Consensus (RANSAC) Regression is recommended for its superior performance in predictive accuracy.

Bibliography

[1] Ira W Levin and Rohit Bhargava. Fourier transform infrared vibrational spectroscopic imaging: integrating microscopy and molecular recognition. *Annu. Rev. Phys. Chem.*, 56:429–474, 2005.

[2] Mariangela Di Donato, Rachel O Cohen, Bruce A Diner, Jacques Breton, Rienk Van Grondelle, and Marie Louise Groot. Primary charge separation in the photosystem ii core from synechocystis: a comparison of femtosecond visible/midinfrared pump-probe spectra of wild-type and two p680 mutants. *Biophysical Journal*, 94(12):4783–4795, 2008.

[3] Irvin Noel Booysen, Sanam Maikoo, Matthew Piers Akerman, and Bheki Xulu. Novel ruthenium (ii) and (iii) compounds with multidentate schiff base chelates bearing biologically significant moieties. *Polyhedron*, 79:250–257, 2014.

[4] MS Bhatia, AK Mulani, PB Choudhari, KB Ingale, and NM Bhatia. Synthesis and qsar analysis of 5-substituted(arylmethylene) pyridin-2-amine derivatives as potential antibacterials. *International journal of drug discovery*, 1(1), 2009.

[5] Haksoon Ahn, Elizabeth J. Greeno, Charlotte Lyn Bright, Samantha Hartzel, and Sarah Reiman. A survival analysis of the length of foster parenting duration and implications for recruitment and retention of foster parents. *Children and Youth Services Review*, 79:478–484, 2017.

[6] Ernest Yeboah Boateng, Daniel A Abaye, et al. A review of the logistic regression model with emphasis on medical research. *Journal of data analysis and information processing*, 7(04):190, 2019.

[7] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

[8] 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.

[9] Antonio R Linero. Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113(522):626–636, 2018.

[10] Philip J Brown, Marina Vannucci, and Tom Fearn. Multivariate bayesian variable selection and prediction. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(3):627–641, 1998.

[11] Yagyensh Chandra Pati, Ramin Rezaiifar, and Perinkulam Sambamurthy Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of 27th Asilomar conference on signals, systems and computers*, pages 40–44. IEEE, 1993.

[12] Grzegorz Swirszcz, Naoki Abe, and Aurelie C Lozano. Grouped orthogonal matching pursuit for variable selection and prediction. *Advances in Neural Information Processing Systems*, 22, 2009.

[13] Shao Thing Teoh, Miki Kitamura, Yasumune Nakayama, Sastia Putri, Yukio Mukai, and Eiichiro Fukusaki. Random sample consensus combined with partial least squares regression (ransac-pls) for microbial metabolomics data mining and phenotype improvement. *Journal of bioscience and bioengineering*, 122(2):168–175, 2016.

[14] Moumen T El-Melegy. Model-wise and point-wise random sample consensus for robust regression and outlier detection. *Neural networks*, 59:23–35, 2014.

[15] Tong Tong Wu, Yi Fang Chen, Trevor Hastie, Eric Sobel, and Kenneth Lange. Genome-wide association analysis by lasso penalized logistic regression. *Bioinformatics*, 25(6):714–721, 2009.

[16] Jian Huang, Patrick Breheny, and Shuangge Ma. A selective review of group selection in high-dimensional models. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 27(4), 2012.

[17] 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.

[18] Hege M Bøvelstad, Ståle Nygård, Hege L Størvold, Magne Aldrin, Ørnulf Borgan, Arnoldo Frigessi, and Ole Christian Lingjærde. Predicting survival from microarray data—a comparative study. *Bioinformatics*, 23(16):2080–2087, 2007.

[19] Bhaskaran David Prakash. Development and investigation of chemometric baseline correction approaches and metabonomic classification algorithms. 2015.

[20] Kim-Anh Lê Cao and Zoe Marie Welham. *Multivariate data integration using R: methods and applications with the mixOmics package.* Chapman and Hall/CRC, 2021.

[21] Christine De Mol, Domenico Giannone, and Lucrezia Reichlin. Forecasting using a large number of predictors: Is bayesian shrinkage a valid alternative to principal components? *Journal of Econometrics*, 146(2):318–328, 2008.

[22] Alexandre Belloni and Victor Chernozhukov. Least squares after model selection in high-dimensional sparse models. 2013.

[23] Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian data analysis.* Chapman and Hall/CRC, 1995.

[24] James S Clark and Alan E Gelfand. A future for models and data in environmental science. *Trends in Ecology & evolution*, 21(7):375–380, 2006.

[25] Gary Koop and Simon M Potter. Estimation and forecasting in models with multiple breaks. *The Review of Economic Studies*, 74(3):763–789, 2007.

[26] Edward Greenberg. *Introduction to Bayesian econometrics.* Cambridge University Press, 2013.

[27] Matthew Stephens and David J Balding. Bayesian statistical methods for genetic association studies. *Nature Reviews Genetics*, 10(10):681–690, 2009.

[28] Seunggeun Lee, Michael C Wu, and Xihong Lin. Optimal tests for rare variant effects in sequencing association studies. *Biostatistics*, 13(4):762–775, 2012.

[29] Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on information theory*, 53(12):4655–4666, 2007.

[30] David L Donoho, Michael Elad, and Vladimir N Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on information theory*, 52(1):6–18, 2005.

[31] Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, 2010.

[32] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1), 2010.

[33] Helen E Dowling. *Mapped permanent quadrats: A window through time into herbaceous plant demography.* Northern Arizona University, 2015.

[34] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[35] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Pattern Recognition: 25th DAGM Symposium, Magdeburg, Germany, September 10-12, 2003. Proceedings 25*, pages 236–243. Springer, 2003.

[36] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767, 2004.

[37] Luis E Navarro-Serment, Christoph Mertz, and Martial Hebert. Pedestrian detection and tracking using three-dimensional ladar data. *The International Journal of Robotics Research*, 29(12):1516–1528, 2010.

[38] Tianzi Jiang, Nassir Navab, Josien PW Pluim, and Max A Viergever. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010: 13th International Conference, Beijing, China, September 20-24, 2010, Proceedings, Part III*, volume 6363. Springer, 2010.

[39] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[40] Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[41] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

[42] Harold Hotelling. Analysis of a complex of statistical variables with principal components. *J. Educ. Psy.*, 24:498–520, 1933.

[43] Lei Fan, Shuai Chen, Qun Li, and Zhouli Zhu. Variable selection and model prediction based on lasso, adaptive lasso and elastic net. In *2015 4th International Conference on Computer Science and Network Technology (ICCSNT)*, volume 1, pages 579–583. IEEE, 2015.

[44] Yin Lu. Variable selection method via the elastic net in generalized linear models. *Beijing: Jiaotong University*, 2011.

[45] Jennifer A Hoeting, Adrian E Raftery, and David Madigan. Bayesian variable and transformation selection in linear regression. *Journal of Computational and Graphical Statistics*, 11(3):485–507, 2002.

[46] Edward I George and Robert E McCulloch. Approaches for bayesian variable selection. *Statistica sinica*, pages 339–373, 1997.

[47] Zihang Lu and Wendy Lou. Bayesian approaches to variable selection: a comparative study from practical perspectives. *The International Journal of Biostatistics*, 18(1):83–108, 2022.

[48] Cathy WS Chen, David B Dunson, Craig Reed, and Keming Yu. Bayesian variable selection in quantile regression. *Statistics and its Interface*, 6(2):261–274, 2013.

[49] Yongtao Guan and Matthew Stephens. Bayesian variable selection regression for genome-wide association studies and other large-scale problems. 2011.

[50] Yanxi Xie, Yuewen Li, Victor Shi, and Quan Lu. An orthogonal matching pursuit variable screening algorithm for high-dimensional linear regression models. *Scientific Programming*, 2022(1):6446903, 2022.

[51] Antony Joseph. Variable selection in high-dimension with random designs and orthogonal matching pursuit. *Journal of Machine Learning Research*, 14(7), 2013.

[52] Aurelie Lozano, Grzegorz Swirszcz, and Naoki Abe. Group orthogonal matching pursuit for logistic regression. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 452–460. JMLR Workshop and Conference Proceedings, 2011.

[53] Omer F Alcin, Abdulkadir Sengur, Jiang Qian, and Melih C Ince. Omp-elm: orthogonal matching pursuit-based extreme learning machine for regression. *Journal of Intelligent Systems*, 24(1):135–143, 2015.

[54] Xiaoshuang Shi, Fuyong Xing, Zhenhua Guo, Hai Su, Fujun Liu, and Lin Yang. Structured orthogonal matching pursuit for feature selection. *Neurocomputing*, 349:164–172, 2019.

[55] David J Nott, Minh-Ngoc Tran, and Chenlei Leng. Variational approximation for heteroscedastic linear models and matching pursuit algorithms. *Statistics and Computing*, 22:497–512, 2012.

[56] Sundeep Rangan and Alyson K Fletcher. Orthogonal matching pursuit from noisy random measurements: A new analysis. *Advances in Neural Information Processing Systems*, 22, 2009.

[57] Mladen Kolar and Eric P Xing. Ultra-high dimensional multiple output learning with simultaneous orthogonal matching pursuit: A sure screening approach. *arXiv preprint arXiv:1012.3880*, 2010.

[58] Charles Soussen, Rémi Gribonval, Jérôme Idier, and Cédric Herzet. Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares. *IEEE Transactions on Information Theory*, 59(5):3158–3174, 2013.

[59] Konstantinos Skianis, Nikolaos Tziortziotis, and Michalis Vazirgiannis. Orthogonal matching pursuit for text classification. *arXiv preprint arXiv:1807.04715*, 2018.

[60] Yudong Chen and Constantine Caramanis. Orthogonal matching pursuit with noisy and missing data: Low and high dimensional results. *arXiv preprint arXiv:1206.0823*, 2012.

[61] Rahul Raguram, Ondrej Chum, Marc Pollefeys, Jiri Matas, and Jan-Michael Frahm. Usac: A universal framework for random sample consensus. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):2022–2038, 2012.

[62] André Veríssimo, Marta B Lopes, Eunice Carrasquinha, and Susana Vinga. Random sample consensus for the robust identification of outliers in cancer data. In *International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics*, pages 108–118. Springer, 2019.

[63] Savio Pereira and John Ferris. Random sampling and probabilistic consensus for identifying outliers in road surface datasets. *International Journal of Vehicle Systems Modelling and Testing*, 14(2-3):133–148, 2020.

[64] Yingmao Li and Nicholas R Gans. Predictive ransac: Effective model fitting and tracking approach under heavy noise and outliers. *Computer Vision and Image Understanding*, 161:99–113, 2017.

[65] Omer Kaspi, Abraham Yosipof, and Hanoch Senderowitz. Random sample consensus (ransac) algorithm for material-informatics: application to photovoltaic solar cells. *Journal of cheminformatics*, 9:1–15, 2017.

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

[67] Fan Wang, Sach Mukherjee, Sylvia Richardson, and Steven M Hill. High-dimensional regression in practice: an empirical study of finite-sample prediction, variable selection and ranking. *Statistics and computing*, 30:697–719, 2020.

[68] Kaushalya Dissanayake and Md Gapar Md Johar. Comparative study on heart disease prediction using feature selection techniques on classification algorithms. *Applied Computational Intelligence and Soft Computing*, 2021(1):5581806, 2021.