

Support Vector Regression with a Novel ϵ -Insensitive Loss Function

by

Gayathri K. Subasinghe

A Thesis submitted to the Faculty of Graduate and
Postdoctoral Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Statistics
University of Manitoba
Winnipeg

Copyright © 2026 by Gayathri K. Subasinghe

Abstract

Support Vector Regression (SVR) is a widely used technique for predictive modeling and function estimation. SVR employs a predefined tolerance margin, known as the ϵ -tube, where only deviations that fall outside the tube are penalized through a loss function. However, a major limitation of standard SVR is its sensitivity to outliers. As a result, the choice of loss function is crucial, since it determines how penalties are assigned and directly shapes the optimization objective while influencing robustness to outliers. In this work, we propose a new class of robust asymmetric ϵ -insensitive loss functions for SVR. The proposed loss preserves the interpretability of the standard ϵ -insensitive formulation and introduces a smooth, differentiable asymmetric penalty beyond the tube. Both theoretical analysis and empirical results demonstrate the competitive performance of the proposed approach. This enhancement improves SVR's robustness to outliers under noisy and asymmetric error distributions, thereby broadening its applicability in machine learning and statistical modelling tasks.

Acknowledgment Page

I would like to express my sincere gratitude to my supervisors, Dr. Mohammad Jafari Jozani and Dr. Kevin McGregor, for their invaluable guidance, encouragement, and continuous support throughout my Master's research. Their expertise, insightful feedback, and patience have greatly contributed to the successful completion of this thesis. I am truly grateful for the time and effort they devoted to mentoring me throughout this journey.

I would also like to extend my sincere appreciation to my thesis examining committee members, Dr. Sumeet Kalia and Dr. Xiaoting Li, for taking the time to review my thesis and for providing thoughtful comments and constructive suggestions that have helped improve the quality of this work.

Lastly, I am grateful to the Department of Statistics at the University of Manitoba for providing an excellent academic environment and the resources that supported my graduate studies. I also gratefully acknowledge the financial support provided by my supervisors, which made this research possible. In addition, I sincerely appreciate the opportunity to serve as a Teaching Assistant in the department throughout my graduate studies. This experience not only provided financial support but also enabled me to develop valuable teaching, communication, and professional skills.

Dedication Page

To my beloved parents and my husband, whose unconditional love, unwavering support, and constant encouragement have been my greatest source of strength throughout this journey. Thank you for always standing by my side and inspiring me to pursue my dreams.

Contents

1	Introduction	1
1.1	An Overview of Statistical Learning	1
1.2	Various Types of Regression Models	5
1.2.1	Mean Regression	6
1.2.2	Median Regression	7
1.2.3	Quantile Regression	9
1.3	SVR Problem and Motivation of This Research	10
1.4	SVR with Asymmetric Loss Functions	15
1.5	Lagrange Multiplier Method and Karush–Kuhn– Tucker Conditions	17
1.5.1	Lagrangian Duality in Constrained Optimization	17
1.5.2	The KKT Optimality Conditions	18
1.5.3	Application to ϵ -insensitive Loss Function in SVR	20
1.6	Nonlinear SVR via Kernel Methods	21
1.7	Outline of the Thesis	25

2	Developing a General Framework for SVR Analysis	26
2.1	Introduction	26
2.2	Dual Derivation Associated with ϵ -SVR	27
2.2.1	Primal Formulation with General Loss Functions	27
2.2.2	Lagrangian and KKT Conditions	34
2.2.3	Interpreting the KKT Conditions	36
2.2.4	Dual Problem Formulation	41
2.2.5	The Role of Conjugate Functions in Dual Formulation	42
2.2.6	ϵ -Calculation	46
2.3	Extension to θ^* -SVR	47
2.3.1	Lagrangian and KKT Conditions for θ^* -SVR	49
2.3.2	Dual Function and Legendre-Fenchel Conjugates for θ^* -SVR	51
2.4	Obtaining ϵ from θ^* -SVR	54
2.5	Application to Specific Loss Functions	56
3	SVR with a Novel Smooth Asymmetric ϵ-insensitive Loss Function	60
3.1	Mathematical Formulation	60
3.2	Properties of the Proposed Loss	62
3.3	Optimization Formulation for ϵ -SVR	67
3.3.1	Primal Problem	68
3.3.2	Dual Problem	68
3.4	Optimization Formulation for θ^* -SVR	69
3.5	Visual Comparison: ϵ -insensitive Pinball Loss vs. Proposed Loss	70

4	Simulation Studies and Real Data Analysis	75
4.1	Performance Criteria	75
4.2	Simulation Studies	77
4.2.1	Linear SVR: Low-Dimensional Settings	79
4.2.2	Non-linear SVR: Low-Dimensional Settings	85
4.2.3	Higher Dimensional Settings	91
4.3	Real Data Analysis	96
5	Conclusion and Future Work	104
5.1	Summary	104
5.2	Limitations and Future Work	106

List of Tables

4.1	Comparison of the performance of linear ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	81
4.2	Comparison of the performance of linear θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	83
4.3	Comparison of the performance of nonlinear ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	87
4.4	Comparison of the performance of nonlinear θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	89
4.5	Prediction performance of the ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for Gaussian and Student's t noise distributions.	93
4.6	Prediction performance of the ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for Gaussian and Student's t noise distributions.	93

4.7	Prediction performance of the θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for Gaussian and Student's t noise distributions.	94
4.8	Prediction performance of the θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for Gaussian and Student's t noise distributions.	95
4.9	Summary of the Real-World Datasets Used in the Analysis.	98
4.10	Organization of the real data analysis results under the ϵ -SVR and θ^* -SVR.	99
4.11	Prediction performance of ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for $\tau \in \{0.1, 0.5, 0.9\}$	100
4.12	Prediction performance of ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for $\tau \in \{0.1, 0.5, 0.9\}$	101
4.13	Prediction performance of θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for $\tau \in \{0.1, 0.5, 0.9\}$	102
4.14	Prediction performance of θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for $\tau \in \{0.1, 0.5, 0.9\}$	103

List of Figures

1.1	Soft Margin SVR with ϵ -tube.	11
1.2	Soft margin SVR corresponding to linear loss (Smola and Schölkopf, 2004).	13
1.3	Comparison of Symmetric ϵ -insensitive Loss Functions: L_1 , L_2 , and Huber Loss.	14
1.4	Comparison of SVR fitted using (a) a linear kernel and (b) an RBF kernel on a nonlinear dataset.	24
3.1	Behavior of ϵ -insensitive pinball loss (solid) and ϵ -insensitive proposed loss (dashed lines) for $\epsilon = 1$	61
3.2	Visual comparison of the ϵ -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss function with (a) linear and (b) RBF kernels with $\epsilon = 0.5$. Columns correspond to (left) the symmetric case ($\tau = 0.5$), (middle) upper outliers ($\tau < 0.5$), and (right) lower outliers ($\tau > 0.5$).	72
3.3	Visual comparison of the θ^* -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss function with (a) linear and (b) RBF kernels with $\theta^* = 0.2$. Columns correspond to (left) the symmetric case ($\tau = 0.5$), (middle) upper outliers ($\tau < 0.5$), and (right) lower outliers ($\tau > 0.5$).	74

4.1	Visual comparison of the performance of linear ϵ -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	82
4.2	Visual comparison of the performance of linear θ^* -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.	84
4.3	Visual comparison of the performance of nonlinear ϵ -SVR models fitted under the proposed loss function (Green) and the ϵ -insensitive pinball loss (Red) in the low-dimensional simulation setting. Solid line: true regression function.	88
4.4	Visual comparison of the performance of nonlinear θ^* -SVR models fitted under the proposed loss function (Green) and the ϵ -insensitive pinball loss (Red) in the low-dimensional simulation setting. Solid line: true regression function.	90

Chapter 1

Introduction

This chapter begins with a concise overview of fundamental concepts in machine learning, followed by a discussion of various classes of regression algorithms. It then focuses on Support Vector Regression (SVR), providing a detailed examination of its formulation and properties. The chapter further outlines the motivation behind this thesis and concludes by exploring how kernel methods extend SVR to effectively model nonlinear relationships.

1.1 An Overview of Statistical Learning

Statistical learning represents a fundamental framework for extracting knowledge and patterns from data through probabilistic and statistical methods. As the scale and scope of data collection expand in various fields, statistical learning has become a major tool for researchers and practitioners who wish to understand and leverage data effectively. At its core, statistical learning concerns itself with the development of algorithms and models that can identify patterns, make predictions, and support decision-making processes based on empirical data ([James et al., 2021](#)). In recent decades, the demand for statistical learning has grown exponentially, alongside increasing computational power

and data availability. This framework provides the theoretical underpinnings for many machine learning algorithms that have transformed fields as diverse as healthcare, finance, genomics, manufacturing, and so on.

In the late 1960s, statistical learning theory was initially developed and was primarily concerned with function estimation problems until the 1990s, when new learning algorithms emerged based on this theoretical foundation ([Vapnik, 1998](#)). Statistical learning approaches are particularly valuable because they provide principled methods for dealing with uncertainty, variability, and complexity in data. These methods provide mechanisms for quantifying the confidence of predictions, addressing incomplete or missing information, and achieving a balance between model complexity and generalization capability. Considering their versatility, statistical learning techniques can be applied not only to prediction tasks, where the primary objective is to predict future results, but also to inference tasks, where the focus lies on understanding the relationships between variables ([James et al., 2021](#)).

Statistical learning algorithms can be categorized based on the nature of their learning process and the type of problem they are designed to solve. The primary categories are Supervised Learning and Unsupervised Learning. Each approach has different characteristics, applications, and methodological considerations. Supervised learning is one of the most widely used paradigms in statistical learning. In this framework, the algorithm learns to map input data to specific outputs based on example input-output pairs ([Mitchell, 1997](#)). The goal of supervised learning is for the trained model to accurately predict the output for new, unseen data ([Hastie et al., 2009](#)). Typically, supervised learning can be further divided into two main types: classification and regression. Classification involves predicting discrete categorical labels, and regression involves predicting continuous numerical values. Real-world applications of supervised learning in-

clude medical diagnosis, image classification, email spam filtering, fraud detection, etc. (Provost and Fawcett, 2013). Unsupervised learning applies to unlabeled data, where the algorithm seeks to identify inherent patterns, structures, or relationships without predefined output labels. The most common unsupervised learning technique is clustering, which groups similar data points based on similarity of characteristics (Jain et al., 1999).

This study focuses on regression problems, where the objective is to predict a continuous target variable from a set of features. Regression analysis plays a central role in predictive modeling and has broad applications in areas such as economics, engineering, environmental science, and healthcare (Montgomery et al., 2012). Numerous algorithms have been proposed for regression tasks, including linear regression, decision trees, random forests, gradient boosting machines, and support vector regression (SVR). Among these methods, SVR provides a strong theoretical foundation and excellent generalization performance, making it a powerful tool for regression analysis.

Support Vector Machine (SVM) represents one of the most significant developments in statistical learning theory. Vladimir Vapnik and his colleagues conducted the foundational work on SVM in 1964 (Vapnik and Chervonenkis, 1964). SVM can be applied to both classification and regression tasks. For classification, the primary objective is to identify an optimal hyperplane that separates data points from different classes with the maximum possible margin. A significant milestone in the development of SVM was the introduction of the kernel trick, which enables the method to handle non-linear relationships by mapping the original input features into a higher-dimensional feature space where linear separation becomes possible (Schölkopf and Smola, 2003). SVR extends this framework to continuous response variables.

The primary objective of SVR is to estimate a regression function that accurately captures the underlying relationship between the predictors and the response while maintaining good generalization performance by controlling model complexity. A key feature of SVR is a predefined tolerance margin, known as the ϵ -tube, which is positioned around the regression function. Only deviations that fall outside this tube are penalized using a loss function. This design encourages data points that lie outside the ϵ -tube to become support vectors that directly influence the regression function. Thus, SVR balances model complexity and approximation accuracy by ignoring small deviations interpreted as acceptable noise while focusing on significant prediction errors (Smola and Schölkopf, 2004).

In formal terms, given a dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$ where $\mathbf{x}_i \in \mathbf{X}^p$ is a p -dimensional feature vector and $y_i \in \mathbb{R}$ represents the continuous target variable, the goal of SVR is to find a function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ such that $f(\mathbf{x}_i)$ is a good approximation of y_i for all $i \in \{1, 2, \dots, n\}$, while minimizing model complexity and allowing deviations within a predefined ϵ -insensitive region, thereby controlling prediction errors. That is, all deviations within ϵ are considered to be negligible, while any error larger than this value is not tolerated.

SVR has been widely used in many applied domains due to its flexibility and ability to model complex nonlinear relationships without imposing strong parametric assumptions. For instance, SVR performs very well for several important tasks in biomedical and health-related applications, such as disease risk prediction, pharmacokinetic modeling, and patient-specific outcome forecasting (Akay and Tunçeli, 2021). It has also been successfully used in economics and finance for stock return forecasting and volatility estimation (Tay and Cao, 2001), as well as in engineering, such as environmental modeling, energy demand estimation, and signal processing (Haben et al., 2021). SVR

formulation highly depends on the choice of loss function, which is essential because it determines how deviations from the ϵ -tube are penalized and directly influences the optimization objective in SVR. The flexibility of SVR as a regression framework offers strong motivation to study alternative loss functions or modifications that could further improve the performance of SVR models in some specific contexts.

Despite its strong theoretical foundation and practical success, SVR can face challenges when applied to large-scale datasets and in the presence of noisy and outlier-contaminated observations. Motivated by the need for robustness under such conditions, this study focuses on developing a unified framework for SVR modeling under a very general class of asymmetric loss functions. We also introduce a novel asymmetric loss function within the SVR framework to improve robustness and computational efficiency.

1.2 Various Types of Regression Models

The idea of regression analysis was coined by Sir Francis Galton in the 19th century to describe the biological phenomenon where extreme characteristics, like parent height, tend to revert towards the average in their offspring, known as regression toward the mean ([Allen, 1997](#)). Ronald Fisher expanded Galton’s initial ideas on regression by developing them into a rigorous statistical framework. Through this progression, regression analysis developed into a powerful tool for modeling relationships among variables by fitting mathematical functions to data points and enabling reliable predictions ([Aldrich, 2005](#)).

Over time, several regression techniques have been developed to model different characteristics of the conditional distribution of a response variable given a set of features.

While classical regression focuses on estimating the conditional mean, alternative methods such as median regression and quantile regression estimate other conditional characteristics of the response distribution, providing greater robustness and flexibility. In this thesis, we focus on SVR and work with regression functions written in the form:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b, \quad (1.1)$$

where $\mathbf{x} \in \mathbb{R}^p$ denotes the vector of explanatory variables, $\mathbf{w} \in \mathbb{R}^p$ is the vector of regression coefficients (weight vector), and b is the intercept or the bias term. The regression methods presented below differ primarily in the loss function used to estimate the regression parameters.

1.2.1 Mean Regression

Mean regression, estimated using Ordinary Least Squares (OLS), is the most widely used regression technique. Its objective is to estimate the conditional expectation of the response variable Y given $\mathbf{X} = \mathbf{x}$, i.e.,

$$E(Y \mid \mathbf{X} = \mathbf{x}).$$

This is obtained by minimizing expected squared prediction errors $\mathbb{E}[(Y - \theta)^2 \mid \mathbf{X} = \mathbf{x}]$ as a function of θ .

Following the general formulation in (1.1), a commonly used approximation to the mean regression model is:

$$Y = f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b. \quad (1.2)$$

The OLS estimator is obtained by minimizing the sum of squared residuals by solving the following optimization problem:

$$\min_{\mathbf{w}, b} \sum_{i=1}^n L_{\text{OLS}}(y_i, f(\mathbf{x}_i)), \quad (1.3)$$

where the loss function for OLS is the squared error loss:

$$L_{\text{OLS}}(y_i, f(\mathbf{x}_i)) = (y_i - f(\mathbf{x}_i))^2. \quad (1.4)$$

The OLS solution, which minimizes the sum of squared errors, has a well-known closed-form solution. However, its reliance on the squared error loss function makes it highly sensitive to outliers, as large residuals have a disproportionate influence on the estimated parameters. To overcome these limitations while still targeting a robust measure of the center, this framework naturally extends to median regression by substituting the squared error loss with the absolute error loss function ([Chen et al., 2017](#)).

1.2.2 Median Regression

Median regression, estimated using Least Absolute Deviations (LAD), is a robust alternative to OLS that targets the conditional median rather than the conditional mean. Its objective is to estimate the conditional median of the response variable,

$$\text{Median}(Y \mid \mathbf{X} = \mathbf{x}),$$

by determining the regression function that minimizes $\mathbb{E}[|Y - \theta| \mid \mathbf{X} = \mathbf{x}]$, expected absolute prediction errors. Unlike OLS, which is sensitive to outliers due to its squared loss, LAD provides a more reliable measure of central tendency when the data contain extreme values or follow skewed distributions ([Koenker, 2005](#)).

The LAD estimator is obtained by minimizing the sum of absolute residuals. This corresponds to solving the following optimization problem:

$$\min_{\mathbf{w}, b} \sum_{i=1}^n L_{\text{LAD}}(y_i, f(\mathbf{x}_i)), \quad (1.5)$$

where the loss function for LAD is the absolute error loss:

$$L_{\text{LAD}}(y_i, f(\mathbf{x}_i)) = |y_i - f(\mathbf{x}_i)|. \quad (1.6)$$

This loss function can be viewed as a special case of the pinball or check loss function (1.9) used in quantile regression, with the quantile level set to $\tau = 0.5$. At this specific value, the asymmetric pinball loss becomes symmetric and reduces to the absolute deviation (see Section 1.2.3).

The LAD estimator offers several advantages over OLS. Its use of the L_1 norm makes it highly robust to outliers, as large residuals are penalized linearly rather than quadratically, preventing extreme observations from exerting disproportionate influence on the estimated parameters. Additionally, median regression provides a more reliable estimate of central tendency in skewed distributions, where the mean may be misleading (Koenker, 2005).

Despite its robustness, median regression has certain limitations. It can be computationally more demanding than OLS, particularly for large datasets, due to the lack of a closed-form solution. Moreover, the LAD estimator is statistically less efficient than OLS, typically producing estimates with higher variance. The method also focuses exclusively on the median, providing no information about other aspects of the conditional distribution, such as the tails or spread. Furthermore, the median does not always correspond to an actual data point, which may complicate interpretation in some applications (Koenker, 2005).

1.2.3 Quantile Regression

Quantile regression, introduced by [Koenker and Bassett \(1978\)](#), generalizes median regression to any quantile $\tau \in (0, 1)$. Its objective is to estimate the conditional τ^{th} quantile of the response variable,

$$Q_\tau(Y \mid \mathbf{X} = \mathbf{x}),$$

by determining the regression function that minimizes a weighted expectation of absolute prediction errors with asymmetric weights, i.e., $\mathbb{E}[\rho_\tau(Y - \theta) \mid \mathbf{X} = \mathbf{x}]$, where $\rho_\tau(\cdot)$ is given in (1.9). Instead of modeling only the conditional mean or median, quantile regression provides a comprehensive analysis of the relationship between variables across the entire distribution of the dependent variable.

The quantile regression estimator is obtained by minimizing the sum of asymmetric absolute residuals:

$$\min_{\mathbf{w}, b} \sum_{i=1}^n L_{\text{QR}\tau}(y_i, f(\mathbf{x}_i)), \quad (1.7)$$

where the loss function for quantile regression is the pinball or check loss:

$$L_{\text{QR}\tau}(y_i, f(\mathbf{x}_i)) = \rho_\tau(y_i - f(\mathbf{x}_i)), \quad (1.8)$$

with

$$\rho_\tau(t) = \begin{cases} (1 - \tau)(-t), & \text{if } t < 0, \\ \tau t, & \text{if } t \geq 0, \end{cases} \quad \text{where } t = y - f(\mathbf{x}_i). \quad (1.9)$$

This loss function assigns a weight of τ to positive residuals (under-predictions) and a weight of $(1 - \tau)$ to negative residuals (over-predictions). This asymmetric weighting

scheme forces the optimization to fit the τ^{th} conditional quantile rather than the conditional mean.

Quantile regression offers several valuable properties that extend beyond OLS regression. Unlike OLS, quantile regression can estimate any conditional quantile, providing a more comprehensive view of the response distribution. It is also more robust to outliers than OLS and can effectively handle heteroscedasticity by allowing error variance to vary with predictor levels. Furthermore, the flexibility of quantile regression enables coefficients to differ between quantiles, providing deeper insight into how predictors influence different parts of the outcome distribution (Koenker, 2005).

Despite its strengths, quantile regression also has several limitations. Reliable estimation, particularly at extreme quantiles, generally requires large sample sizes, and data sparsity can pose challenges in high-dimensional settings. The results are often more complex to interpret than those from standard regression models, as different quantiles may produce divergent relationships. The results are also sensitive to the choice of quantiles modeled, which can make general conclusions difficult. Moreover, like OLS, quantile regression assumes linearity and independence, which may not always hold in practice (Koenker, 2005).

1.3 SVR Problem and Motivation of This Research

Mathematically, given a set of training data $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$, the goal of SVR is to learn a function:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b,$$

that fits the data within an ϵ tolerance while remaining as flat as possible (Smola and Schölkopf, 2004). Flatness in this context is achieved by minimizing the norm of the

weight vector $\|\mathbf{w}\|^2$, which controls the complexity of the model and helps prevent overfitting.

However, in practice it is not possible to capture all data points within the ϵ tube. Thus, SVR adopts the concept of a “soft margin”, by introducing slack variables u_i and l_i to allow for some violations of the ϵ -insensitive zone. These slack variables are non-negative variables that allow some flexibility in the model when the data points fall outside the ϵ -insensitive tube. The slack variables u_i account for errors above the tube, (i.e., $y_i > f(\mathbf{x}_i) + \epsilon$) and l_i for errors below tube, (i.e., $y_i < f(\mathbf{x}_i) - \epsilon$) for $i = 1, \dots, n$. If observations fall inside the ϵ -tube or on the boundaries of the tube, then their corresponding slack variables become zero. Figure 1.1 emphasizes this idea more precisely.

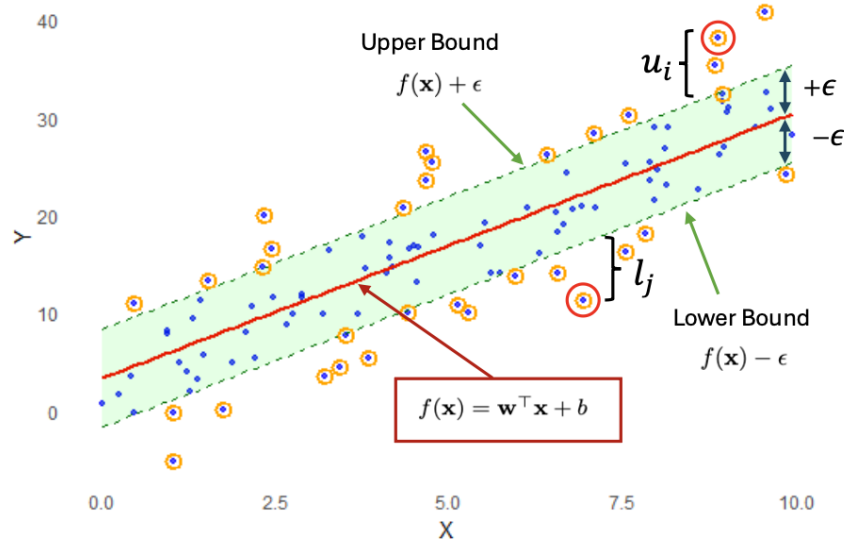


Figure 1.1: Soft Margin SVR with ϵ -tube.

The objective function of the soft-margin SVR thus balances two competing goals, minimizing the complexity of the model and minimizing the total error beyond the ϵ -margin. This is achieved through optimization using the ϵ -insensitive loss function,

which was introduced by [Vapnik \(1995a\)](#) defined as:

$$\rho_{L_1}^\epsilon(t) = \begin{cases} 0, & |t| \leq \epsilon, \\ |t| - \epsilon, & |t| > \epsilon, \end{cases} \quad \text{where } t = y - f(\mathbf{x}), \quad (1.10)$$

which ignores small prediction errors and penalizes only those exceeding the ϵ -margin. Using this loss function, the soft-margin SVR optimization problem can be formulated as:

$$Q = \min_{\mathbf{w} \in \mathbb{R}^p, b, u_i, l_i \in \mathbb{R}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (u_i + l_i), \quad (1.11)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \end{cases} \quad \text{for } i = 1, 2, \dots, n.$$

Here, the constant $C > 0$ determines the trade-off between the flatness of $f(\mathbf{x})$ and the amount up to which deviations larger than ϵ are tolerated. A larger value of C assigns a higher penalty to errors exceeding the ϵ -insensitive region, leading to lower training error but potentially reduced smoothness. Conversely, a smaller C promotes a flatter function at the cost of allowing more violations.

To further clarify the behavior of the ϵ -insensitive loss, we present the classical geometric interpretation of the soft-margin SVR framework. [Figure 1.2](#) illustrates how the regression function is surrounded by an ϵ -tube penalty. Observations that fall outside this tube incur linear penalties quantified through slack variables, thereby enabling controlled deviations while preserving model flatness.

In addition to the ϵ -insensitive loss, several alternative loss functions have been explored in the SVR literature to improve the robustness of the model. Some commonly used examples include:

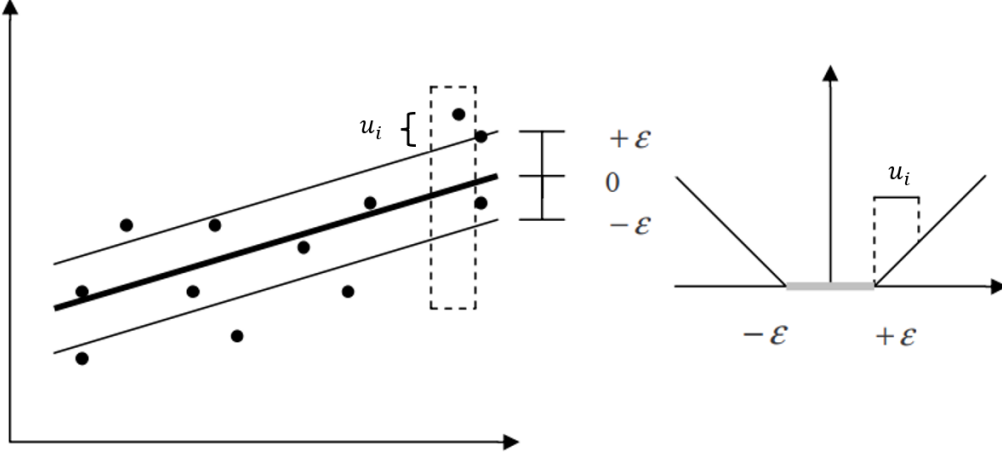


Figure 1.2: Soft margin SVR corresponding to linear loss (Smola and Schölkopf, 2004).

- ϵ -insensitive L_2 -loss

$$\rho_{L_2}^\epsilon(t) = \begin{cases} 0, & |t| \leq \epsilon, \\ \frac{1}{2} (|t| - \epsilon)^2, & |t| > \epsilon. \end{cases} \quad (1.12)$$

The loss function (1.12) emphasizes large errors and is useful when prediction accuracy must heavily penalize large deviations. By replacing the linear penalty outside the ϵ -tube with a quadratic penalty, the optimization problem becomes easier to solve due to its closed-form dual representation. However, the squared loss heavily penalizes large residuals, making the estimator highly sensitive to outliers (Legendre, 1805).

- ϵ -insensitive Huber loss

$$\rho_\delta^\epsilon(t) = \begin{cases} 0, & |t| \leq \epsilon, \\ \frac{1}{2} (|t| - \epsilon)^2, & \epsilon < |t| \leq \epsilon + \delta, \\ \delta (|t| - \epsilon) - \frac{1}{2} \delta^2, & |t| > \epsilon + \delta. \end{cases} \quad (1.13)$$

This loss combines quadratic penalization (L_2) for small residuals with linear penalization (L_1) for large residuals. The Huber loss improves robustness compared to pure quadratic loss while maintaining differentiability. Nevertheless, its symmetric structure assumes that overestimation and underestimation are equally undesirable (Huber, 1964).

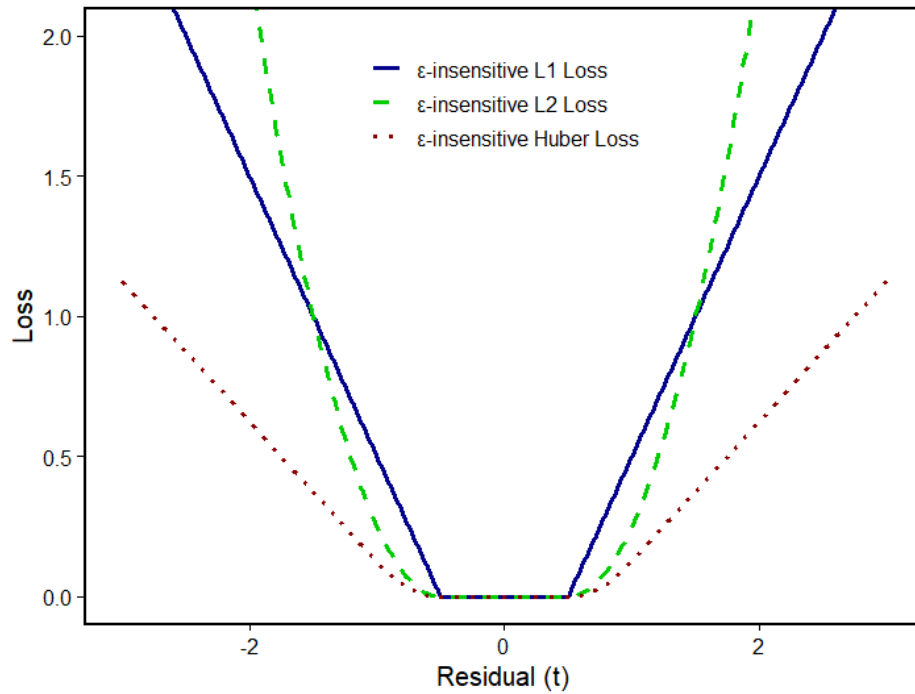


Figure 1.3: Comparison of Symmetric ϵ -insensitive Loss Functions: L_1 , L_2 , and Huber Loss.

Introducing these loss functions provides a foundation for understanding how SVR can be adapted to various modeling scenarios. However, all these loss functions above are symmetric (Figure 1.3), implying that they penalize overestimation and underestimation equally. This symmetry can be restrictive in applications where the cost of underestimation differs substantially from the cost of overestimation. In such cases, symmetric loss functions fail to capture the true risk structure of the problem. To address these limitations, asymmetric loss functions have been introduced into the SVR

framework, such as the quantile loss (also known as the check loss), which allows the model to assign different penalties to positive and negative deviations and estimate conditional quantiles rather than only the mean ([Koenker and Bassett, 1978](#)).

1.4 SVR with Asymmetric Loss Functions

The extension of quantile regression to the support vector framework was pioneered by [Takeuchi et al. \(2006\)](#), who established that a minimizer of the pinball loss function within a reproducing kernel Hilbert space asymptotically converges to the true quantile function under general conditions. Their Support Vector Quantile Regression (SVQR) model combines the pinball loss with a regularization term, maintaining consistency with the structural risk minimization principle. However, the standard SVQR model lacks sparsity, as all training data points contribute to the empirical risk, limiting its computational efficiency for large-scale applications ([Anand et al., 2019](#)).

To address this limitation, several researchers have attempted to incorporate asymmetric loss functions into ϵ -insensitive zones. A significant advancement in addressing these limitations was made by [Anand et al. \(2019\)](#), who proposed a novel asymmetric ϵ -insensitive pinball loss function that genuinely extends the ϵ -insensitive approach to quantile estimation. Their proposed loss function maintains a fixed total insensitive zone width while dividing it asymmetrically based on the quantile parameter τ . Formally, for a given quantile $\tau \in (0, 1)$, the asymmetric ϵ -insensitive pinball loss is defined as:

$$\rho_{\tau}^{\epsilon}(t) = \max\{-(1 - \tau)(t + \tau\epsilon), 0, \tau(t - (1 - \tau)\epsilon)\}. \quad (1.14)$$

This formulation ensures that the total width of the ϵ -insensitive zone remains fixed at ϵ , while the division between the lower and upper insensitive regions is determined by

the quantile parameter τ .

While asymmetric loss functions address the issue of differential error costs, they remain vulnerable to severe outliers due to their unbounded nature. When an outlier produces an extremely large error, the loss grows without bound, potentially exerting disproportionate influence on the estimated regression function (Wang and Zhong, 2014). This observation has motivated the development of bounded loss functions that limit the maximum penalty for any single observation.

The evolution from symmetric to asymmetric loss functions, and subsequently to bounded asymmetric losses, represents a significant progression in enhancing SVR robustness. Therefore, the choice of the loss function highlights fundamental impacts on model behavior, robustness, and the interpretation of prediction errors. Motivated by these considerations, this study proposes a new class of robust asymmetric ϵ -insensitive loss functions, denoted by $\rho_{\tau}^{\epsilon,k}(\cdot)$, for SVR.

The proposed loss maintains the interpretability of the standard ϵ -insensitive formulation, and will introduce a smooth and differentiable asymmetric penalty beyond the tube. The degree of asymmetry is controlled by the parameter $\tau \in (0, 1)$, allowing the model to target functions other than the conditional mean, such as conditional quantiles, within this framework. Moreover, the parameter k determines the sharpness of the transition from the flat ϵ -insensitive region to the penalized region, therefore allowing an explicit control over robustness with respect to noisy or contaminated observations. Through theoretical analysis and empirical evaluation, we demonstrate that the proposed formulation provides improved prediction stability and greater generalization than existing symmetric or pinball-style losses, thereby filling an important gap in the SVR literature. Therefore, we can guarantee that the new loss has greater resilience

of SVR in noisy or asymmetric environments, thereby broadening its applicability in modern machine learning and statistical modeling tasks.

1.5 Lagrange Multiplier Method and Karush–Kuhn–Tucker Conditions

Finding an SVR model involves optimization, and this naturally invites the application of classical optimization theory. The fundamental concepts of Lagrangian duality and the Karush–Kuhn–Tucker (KKT) optimality conditions presented here are not intended to solve the dual problem explicitly, but rather to establish the mathematical framework for SVR.

1.5.1 Lagrangian Duality in Constrained Optimization

The fundamental idea underlying the Lagrangian duality method originated in the work of Joseph-Louis Lagrange in the late 18th century. The method of Lagrange multipliers stands as one of the most elegant and enduring contributions to optimization theory, providing a systematic approach to solving constrained optimization ([Bertsekas, 1982](#)).

Consider a general constrained optimization problem in standard form:

$$\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}), \tag{1.15}$$

subject to:

$$\begin{cases} g_i(\mathbf{x}) \leq 0, & \text{for } i = 1, \dots, n, \\ h_j(\mathbf{x}) = 0, & \text{for } j = 1, \dots, q, \end{cases}$$

where f , g_i , and h_j are continuously differentiable functions. The key insight of Lagrangian duality is to incorporate the constraints into the objective function by intro-

ducing dual variables (Lagrange multipliers) associated with each constraint.

The Lagrangian function $\mathcal{L} : \mathbb{R}^p \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}$ is defined as follows:

$$L(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\mathbf{x}) + \sum_{i=1}^n \alpha_i g_i(\mathbf{x}) + \sum_{j=1}^q \beta_j h_j(\mathbf{x}), \quad (1.16)$$

where $\alpha_i \geq 0$ are the Lagrange multipliers for the inequality constraints, and $\beta_j \in \mathbb{R}$ are the multipliers for the equality constraints. The non-negativity of α_i ensures that the Lagrangian provides a lower bound on the primal objective when the constraints are satisfied (Bertsekas, 1999).

The dual function $\theta(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ provides a lower bound for the primal problem. The dual problem seeks to maximize this lower bound:

$$\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \theta(\boldsymbol{\alpha}, \boldsymbol{\beta}), \quad \text{subject to } \alpha_i \geq 0, \quad \text{for } i = 1, \dots, n. \quad (1.17)$$

Under appropriate convexity conditions, specifically when f and g_i are convex functions and h_j are affine, the duality gap (the difference between the primal and dual optimal values) becomes zero. This property, known as strong duality, ensures that solving the dual problem is equivalent to solving the primal problem (Boyd and Vandenberghe, 2004).

1.5.2 The KKT Optimality Conditions

For differentiable convex optimization problems where strong duality holds, the KKT conditions provide necessary and sufficient conditions for optimality (Künzi and Krelle, 1966). These conditions represent a generalization of the method of Lagrange multipliers to include inequality constraints and are fundamental to understanding the structure

of SVR solutions.

For a point $\mathbf{x}^*$ to be optimal, and for corresponding Lagrange multipliers α_i^* and β_j^* , the following conditions must be satisfied.

1. **Stationarity:** The gradient of the Lagrangian with respect to the primal variables vanishes at the optimum:

$$\nabla f(\mathbf{x}^*) + \sum_{i=1}^n \alpha_i^* \nabla g_i(\mathbf{x}^*) + \sum_{j=1}^q \beta_j^* \nabla h_j(\mathbf{x}^*) = 0. \quad (1.18)$$

In this step, we differentiate the Lagrangian function with respect to the primal variables and set the derivative equal to zero in order to determine the point $\mathbf{x}^*$ that minimizes $L(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta})$.

2. **Primal Feasibility:** The original constraints are satisfied:

$$g_i(\mathbf{x}^*) \leq 0, \quad \text{for } i = 1, \dots, n; \quad h_j(\mathbf{x}^*) = 0, \quad \text{for } j = 1, \dots, q. \quad (1.19)$$

3. **Dual Feasibility:** The Lagrange multipliers for inequality constraints are non-negative:

$$\alpha_i^* \geq 0, \quad \text{for } i = 1, \dots, n. \quad (1.20)$$

4. **Complementary Slackness:** For each inequality constraint, either the constraint is active or its corresponding multiplier is zero:

$$\alpha_i^* g_i(\mathbf{x}^*) = 0, \quad \text{for } i = 1, \dots, n. \quad (1.21)$$

The complementary slackness condition is particularly significant in SVR, as it directly leads to the sparsity property. When a constraint is inactive ($g_i(\mathbf{x}^*) < 0$), the corresponding multiplier must be zero, meaning that the data point does

not contribute to the solution. Conversely, when a multiplier is positive, the corresponding constraint must be active, indicating that the data point lies exactly on the boundary of the feasible region.

1.5.3 Application to ϵ -insensitive Loss Function in SVR

Returning to the SVR primal problem from (1.11), the optimization problem involves:

$$\min_{\mathbf{w} \in \mathbb{R}^p, b, u_i, l_i \in \mathbb{R}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (u_i + l_i) \right\},$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \end{cases} \quad \text{for } i = 1, \dots, n.$$

The objective function is convex (quadratic in $\mathbf{w}$ and linear in the slack variables), and all constraints are affine. Therefore, strong duality holds, and the KKT conditions provide necessary and sufficient conditions for optimality.

The Lagrangian for this problem incorporates Lagrange multipliers $\alpha_i, \beta_i \geq 0$, for the two sets of inequality constraints involving the regression function, and $\gamma_i, \delta_i \geq 0$, for the non-negativity constraints on the slack variables:

$$\begin{aligned} L(\mathbf{w}, b, u_i, l_i, \epsilon, \alpha, \beta, \gamma, \delta) &= \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (u_i + l_i) \\ &\quad + \sum_{i=1}^n \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) \\ &\quad + \sum_{i=1}^n \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) \\ &\quad - \sum_{i=1}^n \gamma_i u_i - \sum_{i=1}^n \delta_i l_i. \end{aligned} \tag{1.22}$$

The stationarity conditions require the partial derivatives of L with respect to the primal variables $(\mathbf{w}, b, u_i, l_i, \epsilon)$ to vanish at the optimum. These conditions yield:

$$\begin{cases} \frac{\partial L}{\partial \mathbf{w}} = 0 \\ \frac{\partial L}{\partial b} = 0 \\ \frac{\partial L}{\partial u_i} = 0 \\ \frac{\partial L}{\partial l_i} = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{w} = \sum_{i=1}^n (\alpha_i - \beta_i) \mathbf{x}_i, \\ \sum_{i=1}^n (\alpha_i - \beta_i) = 0, \\ C - \alpha_i - \gamma_i = 0, \\ C - \beta_i - \delta_i = 0. \end{cases} \quad (1.23)$$

The primal feasibility follows:

$$\begin{cases} y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i \leq 0, \\ \mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i \leq 0, \\ u_i \geq 0, \quad l_i \geq 0, \quad \text{for } i = 1, \dots, n. \end{cases} \quad (1.24)$$

The dual feasibility is identified as the following:

$$\begin{cases} \alpha_i \geq 0, \quad \beta_i \geq 0, \\ \gamma_i \geq 0, \quad \delta_i \geq 0, \end{cases} \quad \text{for } i = 1, \dots, n. \quad (1.25)$$

The complementary slackness conditions provide the critical insight for understanding support vectors.

$$\begin{cases} \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) = 0, \\ \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) = 0, \\ \gamma_i u_i = 0, \quad \delta_i l_i = 0, \quad \text{for } i = 1, \dots, n. \end{cases} \quad (1.26)$$

These conditions lead to obtaining the corresponding dual problem and to several important observations about the structure of the SVR solution; we will discuss them in later chapters.

1.6 Nonlinear SVR via Kernel Methods

The linear SVR formulation assumes that the regression function has the following form:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b.$$

However, in many practical applications, the relationship between predictors and response is inherently nonlinear. To address this limitation, SVR can be extended to nonlinear settings through the use of kernel functions. This extension preserves convexity of the optimization problem while enabling highly flexible function estimation (Vapnik, 1998; Smola and Schölkopf, 2004).

The key idea is to map the input data from the original space $\mathbb{R}^p$ into a higher-dimensional (possibly infinite-dimensional) feature space $\mathcal{H}$ (Hilbert space) via a nonlinear mapping;

$$\Phi : \mathbb{R}^p \rightarrow \mathcal{H}.$$

In this transformed space, the regression function becomes:

$$f(\mathbf{x}) = \mathbf{w}_{\mathcal{H}}^{\top} \Phi(\mathbf{x}) + b.$$

Although $\Phi(\mathbf{x})$ may be high-dimensional, the dual representation of SVR depends only on inner products of mapped observations, i.e.,

$$\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle = \Phi(\mathbf{x}_i)^{\top} \Phi(\mathbf{x}_j).$$

This observation leads directly to the kernel trick.

The Kernel Trick. A kernel function is defined as:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle.$$

By substituting inner products in the dual formulation with kernel evaluations, one avoids explicitly computing the feature mapping. This allows nonlinear regression to be performed implicitly in high-dimensional spaces without incurring the computational cost of constructing Φ (Schölkopf and Smola, 2003).

After solving the dual optimization problem, the estimated SVR function is expressed as:

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^n (\alpha_i - \beta_i) K(\mathbf{x}_i, \mathbf{x}) + b, \quad (1.27)$$

where $\hat{f}(\mathbf{x})$ denotes the fitted regression function, α_i and β_i are the optimal dual variables obtained from the optimization problem, and b is the intercept term.

Equation (1.27) shows that prediction depends only on kernel evaluations between the new observation ($\mathbf{x}$) and the training data ($\mathbf{x}_i$), without requiring the explicit computation of the feature mapping $\Phi(\mathbf{x})$. Thus, the kernel trick enables SVR to model complex nonlinear relationships while avoiding the computational burden of working directly in the high-dimensional feature space. Furthermore, since only observations with nonzero dual coefficients contribute to the summation, the fitted regression function remains sparse, with predictions determined solely by the support vectors.

The flexibility of nonlinear SVR is determined by the choice of kernel function. Different kernels correspond to different notions of similarity between observations and induce different feature spaces, allowing SVR to capture a wide range of nonlinear relationships. Consequently, selecting an appropriate kernel is an important aspect of model development and can substantially influence predictive performance. Some of the most commonly used kernel functions are:

1. Linear Kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle.$$

2. Polynomial Kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \langle \mathbf{x}_i, \mathbf{x}_j \rangle + v)^e.$$

3. Radial Basis Function (RBF) Kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp \left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2 \right).$$

4. Sigmoid Kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh \left(\gamma \langle \mathbf{x}_i, \mathbf{x}_j \rangle + v \right).$$

Among these kernels, the RBF kernel is the most widely used in practice because of its ability to model complex nonlinear relationships without requiring prior knowledge of the functional form.

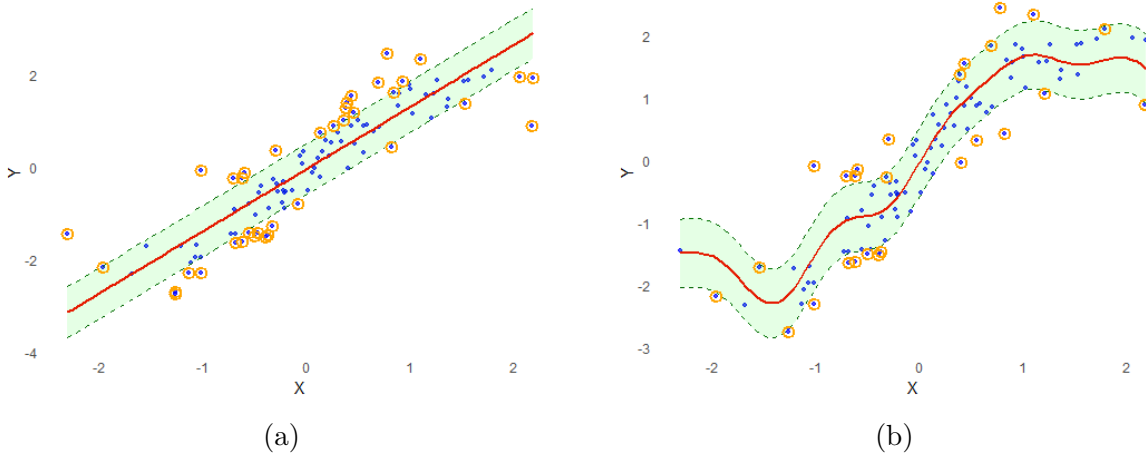


Figure 1.4: Comparison of SVR fitted using (a) a linear kernel and (b) an RBF kernel on a nonlinear dataset.

To illustrate the effect of kernel choice on the fitted regression function, Figure 1.4 compares SVR models fitted using the linear and RBF kernels on the same nonlinear dataset. The linear kernel produces a linear regression function that is unable to capture the underlying nonlinear pattern in the data. In contrast, the RBF kernel generates a flexible nonlinear regression function by implicitly mapping the observations into a higher-dimensional feature space through the kernel trick, resulting in a substantially better fit.

1.7 Outline of the Thesis

In Chapter 1, we discussed the motivation for this research and provided an overview of the techniques to be applied throughout the thesis. In Chapter 2, we develop a very general framework for SVR analysis and study its corresponding dual optimization problem. We first introduce a general class of ϵ -insensitive loss functions for SVR and then develop a systematic procedure for obtaining their corresponding dual problems. In Chapter 3, we introduce a new robust ϵ -insensitive loss function within the SVR framework. We derive and solve the associated dual problem and provide theoretical and graphical illustrations to demonstrate its advantages over the standard SVR. In Chapter 4, simulation studies are conducted to evaluate the performance of the new method using kernel approaches, followed by a real-data analysis to demonstrate its practical applicability. Finally, Chapter 5 concludes the thesis with a summary of the findings and a discussion of possible future work.

Chapter 2

Developing a General Framework for SVR Analysis

In this chapter, we develop a general mathematical framework for deriving the dual formulation in SVR that accommodates a broad class of asymmetric convex loss functions. Starting from its primal problem formulation, we construct the corresponding Lagrangian and establish the KKT conditions. By applying Legendre–Fenchel conjugates, we obtain a unified dual representation for our proposed loss function. The proposed framework provides a unified solution to a general class of SVR problems and, in addition to solving several new problems, yields standard ϵ -insensitive SVR as its special case. We further extend this to the θ^* -SVR setting, highlighting its flexibility and general applicability.

2.1 Introduction

Most SVR formulations rely on specific predefined loss functions and are typically derived using Lagrangian method. While these approaches have been highly successful, they often treat each loss function differently, requiring separate derivations for different variants of SVR ([Smola and Schölkopf, 2004](#)).

Despite these advances, the application of conjugate duality to SVR has not been fully unified in the literature. Existing formulations either focus on specific loss functions or develop general duality results in broader settings without explicitly tailoring them to the SVR structure. For instance, conjugate-based dual derivations have been explored in general convex learning problems and regression settings (Bot and Heinrich, 2014), as well as in broader SVM frameworks, but a cohesive framework dedicated to SVR remains limited.

In practice, there is an increasing interest in flexible modeling frameworks that can accommodate a wide range of loss functions tailored to different applications, including robustness to outliers, asymmetric penalties, and distributional considerations. Therefore, this chapter aims to bridge this gap by developing a general mathematical framework for deriving the dual formulation of SVR under arbitrary convex loss functions.

2.2 Dual Derivation Associated with ϵ -SVR

In this section, we introduce a general framework for SVR analysis and, rather than restricting attention to a specific loss function, develop a unified derivation procedure that accommodates a broad class of loss functions in the SVR setting. The purpose of this framework is to provide a systematic and reusable approach for transforming the primal optimization problem into its corresponding dual form.

2.2.1 Primal Formulation with General Loss Functions

SVR seeks to estimate a function of the form:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b,$$

where $\mathbf{w} \in \mathbb{R}^p$ is the weight vector and $b \in \mathbb{R}$ is the bias term. The objective is to obtain a function that is both simple (flat) and accurate in prediction. In SVR, deviations from the ϵ -insensitive tube are measured using two non-negative slack variables, $u_i \geq 0$, and $l_i \geq 0$ defined respectively as:

$$u_i = \max\{0, y_i - f(\mathbf{x}_i) - \epsilon\}, \quad \text{and} \quad l_i = \max\{0, f(\mathbf{x}_i) - y_i - \epsilon\}.$$

Here, u_i measures the amount by which the observation lies above the tube, while l_i measures the amount by which it lies below the tube.

Referring to the primal formulation in (1.11), the objective function can be written as:

$$\min_{\mathbf{w}, b, u_i, l_i} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (u_i + l_i) \right\}, \quad (2.1)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \end{cases} \quad \text{for } i = 1, \dots, n.$$

To develop a more general framework, we replace the linear penalty on the slack variables by two convex, non-decreasing functions g_1 and g_2 , respectively, where g_1 and g_2 satisfy the following regularity conditions:

1. $g_1, g_2 : [0, \infty) \rightarrow [0, \infty)$ are continuous functions with $g_i(0) = 0$, $i = 1, 2$.
2. g_1 and g_2 are non-decreasing and convex on $[0, \infty)$.

In addition, when deriving the dual optimization problem, we assume that g_1 and g_2 are continuously differentiable on $(0, \infty)$. For some of the derivations, we further require g_1, g_2 to be strictly convex so that their derivatives g'_1 and g'_2 are one-to-one and hence have an inverse. This allows the slack variables to be expressed explicitly in terms of

the corresponding dual variables. We note that our results are more general, and, using the conjugate function argument, we can accommodate cases where the underlying function is not strictly convex. In such non-strictly convex cases, the dual formulation is obtained directly from the Lagrangian and KKT conditions, with the resulting constraints determined by the slopes of the linear components of the loss.

Replacing the linear penalty in (2.1) by the convex functions g_1 and g_2 yields the generalized primal optimization problem:

$$\min_{\mathbf{w}, b, u_i, l_i} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (g_1(u_i) + g_2(l_i)) \right\}, \quad (2.2)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \end{cases} \quad \text{for } i = 1, \dots, n.$$

Eliminating the slack variables using, $u_i = \max\{0, y_i - f(\mathbf{x}_i) - \epsilon\}$, and $l_i = \max\{0, f(\mathbf{x}_i) - y_i - \epsilon\}$, the generalized primal problem can be written as:

$$\min_{\mathbf{w}, b} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \left[g_1(\max\{0, y_i - f(\mathbf{x}_i) - \epsilon\}) + g_2(\max\{0, f(\mathbf{x}_i) - y_i - \epsilon\}) \right],$$

or equivalently,

$$\min_{\mathbf{w}, b} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \left[\max\{0, g_1(y_i - f(\mathbf{x}_i) - \epsilon)\} + \max\{0, g_2(f(\mathbf{x}_i) - y_i - \epsilon)\} \right].$$

Therefore, the generalized optimization problem can be written in the regularized empirical risk minimization form:

$$\min_{\mathbf{w}, b} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n L_{\epsilon, g_1, g_2}(y_i, f(\mathbf{x}_i)),$$

where the generalized loss function is defined as:

$$\begin{aligned}
L_{\epsilon, g_1, g_2}(y, f(\mathbf{x})) &= \rho_{\epsilon, g_1, g_2}(y - f(\mathbf{x})) \\
&= g_1(\max\{0, y_i - f(\mathbf{x}_i) - \epsilon\}) + g_2(\max\{0, f(\mathbf{x}_i) - y_i - \epsilon\}) \\
&= \max\{0, g_1(y_i - f(\mathbf{x}_i) - \epsilon)\} + \max\{0, g_2(f(\mathbf{x}_i) - y_i - \epsilon)\}. \quad (2.3)
\end{aligned}$$

In other words, the generalized loss can be expressed using indicator functions as:

$$\rho_{\epsilon, g_1, g_2}(t) = g_1(t - \epsilon)\mathbb{I}(t > \epsilon) + g_2(-t - \epsilon)\mathbb{I}(t < -\epsilon), \quad \text{where } t = y - f(\mathbf{x}). \quad (2.4)$$

Different choices of the convex functions g_1 and g_2 recover many existing ϵ -insensitive loss functions used in the SVR literature. Some examples are given below.

- **ϵ -Insensitive Absolute Loss**

The standard ϵ -insensitive absolute loss introduced by [Vapnik \(1995a\)](#) is obtained by choosing:

$$g_1(r) = g_2(r) = r, \quad r \geq 0.$$

Which yields,

$$\rho_{\epsilon}(t) = (t - \epsilon)\mathbb{I}(t > \epsilon) + (-t - \epsilon)\mathbb{I}(t < -\epsilon),$$

or equivalently,

$$\rho_{\epsilon}(t) = \begin{cases} -t - \epsilon, & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ t - \epsilon, & t > \epsilon. \end{cases}$$

- **ϵ -Insensitive Squared Loss**

If

$$g_1(r) = g_2(r) = \frac{1}{2}r^2, \quad r \geq 0,$$

gives,

$$\rho_\epsilon(t) = \frac{1}{2}(t - \epsilon)^2 \mathbb{I}(t > \epsilon) + \frac{1}{2}(t + \epsilon)^2 \mathbb{I}(t < -\epsilon),$$

or equivalently,

$$\rho_\epsilon(t) = \begin{cases} \frac{1}{2}(t + \epsilon)^2, & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \frac{1}{2}(t - \epsilon)^2, & t > \epsilon, \end{cases}$$

indicates the ϵ -insensitive squared loss ([Vapnik, 1998](#)).

- **ϵ -Insensitive Huber Loss**

Let,

$$g_1(r) = g_2(r) = \begin{cases} \frac{1}{2}r^2, & 0 \leq r \leq \delta, \\ \delta r - \frac{1}{2}\delta^2, & r > \delta. \end{cases}$$

Then,

$$\rho_{\epsilon,\delta}(t) = g_1(t - \epsilon) \mathbb{I}(t > \epsilon) + g_2(-t - \epsilon) \mathbb{I}(t < -\epsilon),$$

which is the ϵ -insensitive Huber loss ([Huber, 1964](#)).

- **ϵ -Insensitive Pinball Loss**

Choosing

$$g_1(r) = \tau r, \quad g_2(r) = (1 - \tau)r,$$

where $0 < \tau < 1$, gives the ϵ -insensitive pinball loss ([Koenker, 2005](#)) such that,

$$\rho_\tau^\epsilon(t) = \tau(t - \epsilon) \mathbb{I}(t > \epsilon) + (\tau - 1)(t + \epsilon) \mathbb{I}(t < -\epsilon),$$

or equivalently,

$$\rho_\tau^\epsilon(t) = \begin{cases} (\tau - 1)(t + \epsilon), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \tau(t - \epsilon), & t > \epsilon. \end{cases}$$

- **Polynomial ϵ -Insensitive Loss**

A broader family of ϵ -insensitive losses can be obtained by choosing a polynomial penalty of degree $p \geq 1$:

$$g_1(r) = g_2(r) = \frac{1}{p} r^p.$$

Substituting these functions into (2.4) gives:

$$\rho_{\epsilon,p}(t) = \frac{1}{p}(t - \epsilon)^p \mathbb{I}(t > \epsilon) + \frac{1}{p}(-t - \epsilon)^p \mathbb{I}(t < -\epsilon).$$

Equivalently,

$$\rho_{\epsilon,p}(t) = \begin{cases} \frac{1}{p}(-t - \epsilon)^p, & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \frac{1}{p}(t - \epsilon)^p, & t > \epsilon. \end{cases}$$

Thus, the polynomial family includes several familiar losses as special cases. In particular, $p = 1$ gives the standard ϵ -insensitive absolute loss, while $p = 2$ gives the ϵ -insensitive squared loss. More generally, polynomial penalties provide different rates of growth outside the insensitive tube (Smola and Schölkopf, 2004).

- **ϵ -Insensitive Logistic Loss**

A smooth logistic penalty can be defined by choosing:

$$g_1(r) = g_2(r) = \frac{1}{a} \log \left(\frac{1 + e^{ar}}{2} \right), \quad a > 0.$$

yields,

$$\rho_{\epsilon,a}(t) = \frac{1}{a} \log \left(\frac{1 + e^{a(t-\epsilon)}}{2} \right) \mathbb{I}(t > \epsilon) + \frac{1}{a} \log \left(\frac{1 + e^{-a(t+\epsilon)}}{2} \right) \mathbb{I}(t < -\epsilon).$$

Equivalently,

$$\rho_{\epsilon,a}(t) = \begin{cases} \frac{1}{a} \log\left(\frac{1 + e^{-a(t+\epsilon)}}{2}\right), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \frac{1}{a} \log\left(\frac{1 + e^{a(t-\epsilon)}}{2}\right), & t > \epsilon. \end{cases}$$

This smooth ϵ -insensitive logistic loss was used by [Hui-zhong et al. \(2005\)](#) to construct a differentiable support vector regression model.

- **ϵ -Insensitive LINEX Loss**

The LINEX loss is an asymmetric loss that penalizes errors approximately linearly in one direction and exponentially in the other ([Varian, 1975](#)). Within the generalized ϵ -insensitive framework, LINEX loss can be expressed as:

$$g_1(r) = g_2(r) = e^{ar} - ar - 1, \quad a \neq 0.$$

Substitution into (2.4) gives:

$$\rho_{\epsilon,a}^{\text{LINEX}}(t) = e^{a(t-\epsilon)} - a(t-\epsilon) - 1\mathbb{I}(t > \epsilon) + e^{a(-t-\epsilon)} - a(-t-\epsilon) - 1\mathbb{I}(t < -\epsilon).$$

Equivalently,

$$\rho_{\epsilon,p,q}^{\text{LINEX}}(t) = \begin{cases} e^{-a(t+\epsilon)} + a(t+\epsilon) - 1, & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ e^{a(t-\epsilon)} - a(t-\epsilon) - 1, & t > \epsilon, \end{cases}$$

thereby producing an asymmetric ϵ -insensitive LINEX-type loss.

2.2.2 Lagrangian and KKT Conditions

To solve the constrained optimization problem (2.2), we employ the Lagrange multipliers method (see Section 1.5.1). The Lagrangian function is constructed as:

$$\begin{aligned}
L(\mathbf{w}, b, u_i, l_i, \epsilon, \alpha, \beta, \gamma, \delta) = & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (g_1(u_i) + g_2(l_i)) \\
& + \sum_{i=1}^n \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) \\
& + \sum_{i=1}^n \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) \\
& - \sum_{i=1}^n \gamma_i u_i - \sum_{i=1}^n \delta_i l_i,
\end{aligned} \tag{2.5}$$

where $\alpha_i \geq 0, \beta_i \geq 0, \gamma_i \geq 0$ and $\delta_i \geq 0$ are Lagrange multipliers for $i = 1, \dots, n$.

The key idea is how the loss function is incorporated into the primal formulation. In this setting, the loss directly influences the terms $g_1(u_i)$ and $g_2(l_i)$. By assigning the appropriate expressions to these components, the dual objective naturally follows. Moreover, the KKT conditions provide the necessary and sufficient optimality criteria that ensure consistency between the primal and dual formulations and lead to the solution. Hence, we take the partial derivatives of L with respect to the primal variables $\mathbf{w}, b, u_i, l_i$.

1. **Stationarity:** Ensures that the solution minimizes the primal objective while satisfying the dual constraints.

- For $\mathbf{w}$:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} - \sum_{i=1}^n \alpha_i \mathbf{x}_i + \sum_{i=1}^n \beta_i \mathbf{x}_i = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^n (\alpha_i - \beta_i) \mathbf{x}_i. \tag{2.6}$$

- For b :

$$\frac{\partial L}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^n (\alpha_i - \beta_i) = 0. \quad (2.7)$$

- For u_i :

$$\frac{\partial L}{\partial u_i} = 0 \quad \Rightarrow \quad C g_1'(u_i) - \alpha_i - \gamma_i = 0 \quad \Rightarrow \quad u_i = g_1'^{-1} \left(\frac{\alpha_i + \gamma_i}{C} \right). \quad (2.8)$$

- For l_i :

$$\frac{\partial L}{\partial l_i} = 0 \quad \Rightarrow \quad C g_2'(l_i) - \beta_i - \delta_i = 0 \quad \Rightarrow \quad l_i = g_2'^{-1} \left(\frac{\beta_i + \delta_i}{C} \right). \quad (2.9)$$

2. **Primal Feasibility:** These conditions ensure that the original constraints are satisfied.

$$\begin{cases} y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i \leq 0, \\ \mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i \leq 0, \\ u_i \geq 0, \quad l_i \geq 0 \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.10)$$

3. **Dual Feasibility:** These conditions specify that the Lagrange multipliers must be non-negative.

$$\begin{cases} \alpha_i \geq 0, \quad \beta_i \geq 0, \\ \gamma_i \geq 0, \quad \delta_i \geq 0 \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.11)$$

4. **Complementary Slackness:** These conditions enforce that each constraint is either exactly satisfied or inactive.

$$\begin{cases} \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) = 0, \\ \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) = 0, \\ \gamma_i u_i = 0, \quad \delta_i l_i = 0 \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.12)$$

This complementary slackness condition is particularly significant in SVR, as it directly leads to some important properties.

2.2.3 Interpreting the KKT Conditions

The KKT conditions derived above provide several useful insights into the structure of the SVR solution. Before proceeding to the dual formulation, it is essential to understand what these optimality conditions reveal about the solution. This understanding will later help us interpret the dual variables and implement the algorithm efficiently.

1. Identification of Support Vectors

From the complementary slackness conditions in (2.12), we can classify each data point based on the values of its Lagrange multipliers into one of the three groups:

Group 1: Observations strictly inside the ϵ -insensitive tube ($|y_i - f(\mathbf{x}_i)| < \epsilon$).

For an observation $\mathbf{x}_i$, both constraints are strictly satisfied, and we have $y_i - f(\mathbf{x}_i) - \epsilon - u_i < 0$ and $f(\mathbf{x}_i) - y_i - \epsilon - l_i < 0$. Since these inequality constraints are inactive at the optimum, the complementary slackness conditions in (2.12) immediately force the corresponding Lagrange multipliers to be zero, i.e., $\alpha_i = 0$ and $\beta_i = 0$. Furthermore, because these points lie within the tube, no slack is needed, so $u_i = 0$ and $l_i = 0$. Substituting these values into the stationarity conditions (2.8) and (2.9) yields $Cg'_1(0) - \gamma_i = 0$ and $Cg'_2(0) - \delta_i = 0$. For loss functions where $g'_1(0) = g'_2(0) = 0$, which is typical for robust loss functions to ensure differentiability at the origin, then we obtain $\gamma_i = 0$ and $\delta_i = 0$ as well. Consequently, all four Lagrange multipliers vanish for points inside the tube, meaning these points do not contribute to the solution expansion $\mathbf{w} = \sum_{i=1}^n (\alpha_i - \beta_i) \mathbf{x}_i$. Hence, such points are not support vectors and play no role in determining the regression function once the model is trained.

Group 2: Observations on the boundary of the ϵ -tube ($|y_i - f(\mathbf{x}_i)| = \epsilon$).

Consider a point on the upper boundary satisfying $y_i - f(\mathbf{x}_i) = \epsilon$. For such a point, the corresponding inequality constraint becomes $y_i - f(\mathbf{x}_i) - \epsilon - u_i = 0$. Since the point lies exactly on the edge of the ϵ -tube, no slack is required, so we have $u_i = 0$. Complementary slackness then permits the associated Lagrange multiplier $\alpha_i > 0$, while the condition $\gamma_i u_i = 0$ is automatically satisfied regardless of the value of γ_i . Substituting $u_i = 0$ into the stationarity condition derived from the Lagrangian, equation (2.8), gives us the relationship $Cg'_1(0) - \alpha_i - \gamma_i = 0$.

This leads to an important observation about the nature of support vectors. For many robust loss functions, a common and desirable property is that their derivative at zero is zero, i.e., $g'_1(0) = 0$. In such case, $\alpha_i + \gamma_i = 0$. However, given the dual feasibility constraints that require all Lagrange multipliers to be non-negative ($\alpha_i \geq 0, \gamma_i \geq 0$), the only possible solution is $\alpha_i = 0$ and $\gamma_i = 0$.

Thus, this analysis reveals a fundamental distinction: for loss functions with a non-zero derivative at the origin ($g'_1(0) \neq 0$), points on the boundary can act as support vectors. From the stationarity condition (2.8) with $u_i = 0$, we have:

$$Cg'_1(0) - \alpha_i - \gamma_i = 0 \quad \Rightarrow \quad \gamma_i = Cg'_1(0) - \alpha_i.$$

For a point on the boundary to be a support vector, we require $\alpha_i > 0$. Furthermore, the complementary slackness condition $\gamma_i u_i = 0$ is automatically satisfied γ_i , since $u_i = 0$ allowing γ_i to be positive. If $\gamma_i > 0$, then from the equation above we obtain:

$$\gamma_i > 0 \quad \Rightarrow \quad Cg'_1(0) - \alpha_i > 0 \quad \Rightarrow \quad \alpha_i < Cg'_1(0).$$

Thus, when $\gamma_i > 0$, the multiplier α_i is strictly bounded above by $Cg'_1(0)$. Conversely, if $\gamma_i = 0$, then $\alpha_i = Cg'_1(0)$, representing the maximum possible value. Therefore, for loss functions with $g'_1(0) \neq 0$, points on the margin boundary can serve as support vectors with α_i taking any value in the interval $(0, Cg'_1(0)]$, where the upper bound is achieved only when $\gamma_i = 0$. This condition similarly applies for the lower bound as well (when $l_i = 0$). Conversely, for losses with $g'_1(0) = g'_2(0) = 0$, the support vectors are exclusively the points that lie strictly outside ($\alpha_i > 0$ and $\beta_i > 0$) the ϵ -tube. This understanding is crucial for correctly interpreting the role of different data points in the final regression model.

Group 3: Observations outside the ϵ -tube ($|y_i - f(\mathbf{x}_i)| > \epsilon$).

Let's consider the situation where points above the tube ($y_i - f(\mathbf{x}_i) > \epsilon$). In this case positive slack variables i.e., $u_i > 0$. The complementary slackness condition $\gamma_i u_i = 0$ then forces $\gamma_i = 0$. Substituting this into the stationarity condition (2.8) yields $Cg'_1(u_i) - \alpha_i = 0$, or equivalently, $\alpha_i = Cg'_1(u_i)$. Since $u_i > 0$ and for any meaningful loss function we have $g'_1(u_i) \geq 0$ (the penalty increases with the error), it follows that $\alpha_i > 0$. These points are therefore support vectors that actively contribute to the solution.

A completely analogous argument applies to points lying below the tube where $f(\mathbf{x}_i) - y_i > \epsilon$; here $l_i > 0$, complementary slackness forces $\delta_i = 0$, and the stationarity condition (2.9) gives $\beta_i = Cg'_2(l_i) > 0$. Thus, points outside the tube, whether above or below, are always support vectors with positive multipliers, and their influence on the model is directly modulated by the derivative of the loss function evaluated at their respective slack values.

2. The Role of Loss Function Derivatives

The stationarity conditions in (2.8) and (2.9) reveal how the choice of loss function fundamentally shapes the solution structure. These equations establish a direct relationship between the slack variables and the Lagrange multipliers through the derivative of the loss function:

$$g'_1(u_i) = \left(\frac{\alpha_i + \gamma_i}{C} \right), \quad g'_2(l_i) = \left(\frac{\beta_i + \delta_i}{C} \right). \quad (2.13)$$

For points lying outside the ϵ -tube, where $\gamma_i = 0$ and $\delta_i = 0$, these relationships simplify to:

$$g'_1(u_i) = \left(\frac{\alpha_i}{C} \right), \quad g'_2(l_i) = \left(\frac{\beta_i}{C} \right). \quad (2.14)$$

This simplification reveals that the derivative of the loss function directly determines the magnitude of the Lagrange multipliers, and consequently, the influence each data point exerts on the final regression model. When considering loss functions with bounded derivatives, they naturally limit the influence of outliers.

3. Bounds on Lagrange Multipliers

The stationarity conditions not only relate the multipliers to the loss function derivatives but also impose implicit bounds on the allowable values of α_i and β_i . Rearranging (2.8) gives:

$$Cg'_1(u_i) - \alpha_i - \gamma_i = 0 \quad \Rightarrow \quad \alpha_i = Cg'_1(u_i) - \gamma_i.$$

Since the Lagrange multiplier γ_i is constrained to be non-negative ($\gamma_i \geq 0$), it follows directly that $\alpha_i \leq Cg'_1(u_i)$. Combined with the dual feasibility condition $\alpha_i \geq 0$, we obtain the following box constraint for each upper-bound multiplier:

$$0 \leq \alpha_i \leq Cg'_1(u_i). \quad (2.15)$$

A completely analogous derivation using (2.9) yields the corresponding bounds for the lower-bound multipliers:

$$0 \leq \beta_i \leq Cg'_2(l_i). \quad (2.16)$$

These bounds will prove essential when we later derive the dual problem, as they translate into constraints on the dual variables that do not explicitly involve the slack variables u_i and l_i .

4. Bias Term Calculation

Once the optimal multipliers α_i and β_i are determined by solving the dual problem, the bias term b can be obtained using any support vector that lies exactly on the margin boundary. For a point on the upper boundary, we have $u_i = 0$ and the corresponding constraint is active: $y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon = 0$. Solving for b yields:

$$b = y_i - \mathbf{w}^\top \mathbf{x}_i - \epsilon.$$

Such a point is characterized by a multiplier satisfying $0 < \alpha_i < Cg'_1(0)$, where the upper bound ensures $\gamma_i > 0$ and confirms that the point lies exactly on the boundary rather than outside the tube. Similarly, for a point on the lower boundary, we have $l_i = 0$ with $\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon = 0$, giving:

$$b = y_i - \mathbf{w}^\top \mathbf{x}_i + \epsilon,$$

where the corresponding multiplier satisfies $0 < \beta_i < Cg'_2(0)$. In principle, any single such support vector suffices to compute b . However, due to numerical rounding errors and the approximate nature of practical optimization, it is recommended to compute b as the average over all support vectors lying on the margin boundaries. This averaging approach enhances numerical stability and produces a more reliable estimate of the bias term.

2.2.4 Dual Problem Formulation

With these interpretations in hand, we are now ready to derive the dual problem. After taking the necessary derivations and substituting the stationarity conditions back into the Lagrangian (2.5), we obtain the dual function L^* . We first write the expanded Lagrangian:

$$\begin{aligned} L^* = & \frac{1}{2} \mathbf{w}^\top \mathbf{w} + \sum_{i=1}^n C(g_1(u_i) - (\alpha_i + \gamma_i)u_i) + \sum_{i=1}^n C(g_2(l_i) - (\beta_i + \delta_i)l_i) \\ & + \sum_{i=1}^n (\alpha_i - \beta_i)y_i - \mathbf{w}^\top \sum_{i=1}^n (\alpha_i - \beta_i)\mathbf{x}_i - b \sum_{i=1}^n (\alpha_i - \beta_i) - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i). \end{aligned}$$

By rearranging the terms at the optimum point, incorporating (2.8) and (2.9), we obtain the following:

$$\begin{aligned} L^* = & -\frac{1}{2} \mathbf{w}^\top \mathbf{w} + \sum_{i=1}^n C(g_1(u_i) - g'_1(u_i)u_i) + \sum_{i=1}^n C(g_2(l_i) - g'_2(l_i)l_i) \\ & + \sum_{i=1}^n (\alpha_i - \beta_i)y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i). \end{aligned}$$

After substituting the stationary conditions into the above equation, which simplifies to:

$$\begin{aligned} L^* = & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j)\mathbf{x}_i^\top \mathbf{x}_j \\ & + \sum_{i=1}^n C(g_1(u_i) - g'_1(u_i)u_i) + \sum_{i=1}^n C(g_2(l_i) - g'_2(l_i)l_i) \\ & + \sum_{i=1}^n (\alpha_i - \beta_i)y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i). \end{aligned} \tag{2.17}$$

Then, combining all the derived conditions from KKT, the dual maximization problem is formulated as follows:

$$\begin{aligned}
\max_{\alpha, \beta} \quad & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\
& + \sum_{i=1}^n C(g_1(u_i) - g'_1(u_i)u_i) + \sum_{i=1}^n C(g_2(l_i) - g'_2(l_i)l_i) \\
& + \sum_{i=1}^n (\alpha_i - \beta_i)y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i),
\end{aligned}$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad 0 \leq \alpha_i \leq Cg'_1(u_i), \quad 0 \leq \beta_i \leq Cg'_2(l_i), \quad \text{for } i = 1, \dots, n. \quad (2.18)$$

However, we can observe that (2.18) still contains the slack variables u_i and l_i through the terms $(g_1(u_i) - g'_1(u_i)u_i)$ and $(g_2(l_i) - g'_2(l_i)l_i)$. To obtain a dual problem expressed solely in terms of the Lagrange multipliers, we need to eliminate these slack variables by optimizing over them.

2.2.5 The Role of Conjugate Functions in Dual Formulation

To obtain a fully explicit dual problem that eliminates the slack variables u_i and l_i , we employ the Legendre-Fenchel conjugate, a fundamental tool in convex analysis that provides an elegant characterization of the relationship between a function and its dual representation ([Rockafellar, 1970](#)).

Definition 1 (Legendre-Fenchel Conjugate).

Let $f : \mathbb{R}^n \rightarrow [-\infty, \infty]$ be a function (not necessarily convex). The conjugate function of f is the function $f^* : \mathbb{R}^n \rightarrow [-\infty, \infty]$ defined by:

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathbb{R}^n} \{\langle \mathbf{x}, \mathbf{y} \rangle - f(\mathbf{x})\}$$

where $\langle \cdot, \cdot \rangle : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ is the standard inner product, i.e., $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^n x_i y_i$.

Remark 1. A fundamental property of the Legendre-Fenchel transform is that f^* is always convex, regardless of whether f itself is convex. This convexity-preserving property is crucial for maintaining the tractability of the dual optimization problem.

Building on this concept, we now reformulate the dual function in (2.18) by eliminating the slack variables through a systematic procedure.

- **Step 1:** Expressing Slack Variables in Terms of Multipliers

Recall the stationarity conditions from (2.8) and (2.9). For points where the slack variables are strictly positive ($u_i > 0$ or $l_i > 0$), complementary slackness forces $\gamma_i = 0$ and $\delta_i = 0$, respectively. Under these conditions, the stationarity equations simplify to:

$$Cg'_1(u_i) = \alpha_i \implies u_i = (g'_1)^{-1} \left(\frac{\alpha_i}{C} \right), \quad (2.19)$$

$$Cg'_2(l_i) = \beta_i \implies l_i = (g'_2)^{-1} \left(\frac{\beta_i}{C} \right), \quad (2.20)$$

where $(g'_1)^{-1}$ and $(g'_2)^{-1}$ denote the inverse functions of the derivatives, assuming g_1 and g_2 are strictly convex and differentiable.

- **Step 2:** Applying the Legendre-Fenchel Conjugate

Consider the term $g_1(u_i) - g'_1(u_i)u_i$ appearing in (2.18). Evaluating this expression at u_i given by (2.19) yields a function that depends solely on α_i :

$$H_1(\alpha_i) = g_1\left((g'_1)^{-1}\left(\frac{\alpha_i}{C}\right)\right) - \frac{\alpha_i}{C} \cdot (g'_1)^{-1}\left(\frac{\alpha_i}{C}\right). \quad (2.21)$$

A fundamental result in convex analysis establishes that $H_1(\alpha_i)$ is intimately related to the convex conjugate of g_1 .

To make this relation more precise, we apply Definition 1 to the scalar case. Suppose a differentiable convex function $g_1 : \mathbb{R} \rightarrow \mathbb{R}$, its conjugate is:

$$g_1^*(y) = \sup_{u \in \mathbb{R}} \{yu - g_1(u)\}, \quad y \in \mathbb{R}.$$

Setting $y = \alpha_i/C$, the supremum is attained at a point u_i satisfying the first-order optimality condition:

$$y - g'_1(u_i) = 0 \quad \Longleftrightarrow \quad g'_1(u_i) = \frac{\alpha_i}{C}.$$

Assuming g'_1 is invertible, the unique maximizer is:

$$u_i = (g'_1)^{-1}\left(\frac{\alpha_i}{C}\right).$$

Substituting this maximizer into the definition of the conjugate gives:

$$g_1^*\left(\frac{\alpha_i}{C}\right) = \frac{\alpha_i}{C}(g'_1)^{-1}\left(\frac{\alpha_i}{C}\right) - g_1\left((g'_1)^{-1}\left(\frac{\alpha_i}{C}\right)\right).$$

Comparing this with the definition of $H_1(\alpha_i)$ in (2.21), we obtain:

$$H_1(\alpha_i) = -g_1^*\left(\frac{\alpha_i}{C}\right). \quad (2.22)$$

A similar reasoning applies to the lower-bound slack variables. For g_2 , setting $y = \beta_i/C$ and using the optimality condition $g'_2(l_i) = \beta_i/C$ yields:

$$l_i = (g'_2)^{-1}\left(\frac{\beta_i}{C}\right),$$

and therefore,

$$H_2(\beta_i) = g_2\left((g'_2)^{-1}\left(\frac{\beta_i}{C}\right)\right) - \frac{\beta_i}{C}(g'_2)^{-1}\left(\frac{\beta_i}{C}\right) = -g_2^*\left(\frac{\beta_i}{C}\right). \quad (2.23)$$

Since g_1^* and g_2^* are convex functions, by Remark 1 their negatives $-g_1^*$ and $-g_2^*$ are concave. Hence $H_1(\alpha_i)$ and $H_2(\beta_i)$ are concave in α_i and β_i , respectively, which guarantees that the dual objective remains a concave maximization problem.

- **Step 3:** The Complete Dual Formulation

Substituting the expressions from (2.21) and (2.23) into the dual function (2.18), and incorporating the kernel trick by replacing the inner product $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ with a Mercer kernel $K(\mathbf{x}_i, \mathbf{x}_j)$ (Vapnik, 1998), we arrive at the complete dual problem:

$$\begin{aligned} \max_{\alpha, \beta} & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + C \sum_{i=1}^n \left(H_1(\alpha_i) + H_2(\beta_i) \right) \\ & + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i), \end{aligned}$$

subject to:

$$\begin{aligned} & \sum_{i=1}^n (\alpha_i - \beta_i) = 0, \\ & 0 \leq \alpha_i \leq C g'_1(u_i), \\ & 0 \leq \beta_i \leq C g'_2(l_i), \quad \text{for } i = 1, \dots, n. \end{aligned} \quad (2.24)$$

The dual form involves kernel dot products and additional terms $H_1(\alpha_i)$ and $H_2(\beta_i)$, which come from the convex conjugates of g_1 and g_2 . The box constraints on α_i and β_i are also adjusted by the derivatives of g_1 and g_2 .

The dual formulation in (2.24) represents the completion of the general framework developed for ϵ -SVR. We started from the primal optimization problem with general loss functions g_1 and g_2 , then we systematically derived the Lagrangian, established the KKT conditions, and interpreted their implications for support vector characterization. By applying Legendre-Fenchel conjugates, we successfully eliminated the slack variables u_i and l_i and obtained a dual problem expressed entirely in terms of the Lagrange multipliers α_i and β_i . Therefore, this unified approach not only reveals the fundamental structure of SVR but also provides a modular template that can be readily specialized to specific loss functions.

2.2.6 ϵ -Calculation

[Cherkassky and Ma \(2004\)](#) proposed an analytical approach for selecting the ϵ parameter in SVR. Their method is based on the well-established relationship between the ϵ -insensitive zone and the noise level in the training data, while also accounting for the sample size. The proposed formula for calculating ϵ is:

$$\epsilon = \eta \sigma \sqrt{\frac{\ln n}{n}}, \quad (2.25)$$

where σ is the standard deviation of the noise in the training data, n is the number of training samples and η is a constant set to 3 based on empirical tuning.

This formulation builds upon two key theoretical insights. First, ϵ should be proportional to the input noise level, i.e., $\epsilon \propto \sigma$, as suggested by multiple researchers ([Vapnik](#),

1995b; Kwok et al., 2001). Second, from standard statistical theory, the variance of observations about the trend line is inversely proportional to the sample size, i.e., $\sigma_{y/\mathbf{x}}^2 \propto \sigma^2/n$.

The authors found that the simpler relationship $\epsilon \propto \sigma/\sqrt{n}$ works well for small sample sizes but yields values that are too small for large datasets. Therefore, they introduced the logarithmic term $\sqrt{\ln n}$ to provide more robust performance across different sample sizes.

For practical implementation, the noise standard deviation σ must be estimated from the training data. Cherkassky and Ma (2004) recommend using k_0 -nearest neighbors regression (with $k_0=3$) to estimate the noise variance:

$$\hat{\sigma}^2 = \frac{1.5}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2,$$

where $\hat{y}_i$ are the predictions from the k -nearest neighbors model. This approach provides accurate noise estimates even with small sample sizes and high noise levels. By following this approach, once we have calculated the noise level, we can plug that value into (2.25) and obtain the desired ϵ value for the data.

2.3 Extension to θ^* -SVR

While the standard ϵ -SVR formulation introduced in the previous sections provides a flexible framework for regression, it suffers from a practical limitation, as the parameter ϵ , which controls the width of the insensitive tube, is difficult to choose. Selecting an appropriate ϵ requires prior knowledge of the noise level in the data, which is often

unavailable in practice. To address this issue, [Schölkopf et al. \(2000\)](#) proposed an alternative parameterization known as ν -SVR, which throughout the thesis we refer to it as θ^* -SVR, which replaces ϵ with a new parameter $\theta^* \in [0, 1]$, where it specifies an upper bound on the proportion of training observations that may become margin errors and a lower bound on the proportion of support vectors. Consequently, θ^* provides an intuitive interpretation and provides automatic control over the number of support vectors and the width of the ϵ -tube. Here, we extend the ϵ -SVR results under our proposed general loss function to θ^* -SVR setting.

In the primal formulation of θ^* -SVR, the objective is to find a function that approximates the data as closely as possible while maintaining flatness, similar to ϵ -SVR. However, unlike ϵ -SVR, where ϵ is fixed in advance, θ^* -SVR treats ϵ as an optimization variable and introduces the parameter θ^* to regulate its value. The primal formulation of the general optimization problem can be expressed as follows:

$$\min_{\mathbf{w}, b, u_i, l_i, \epsilon} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \left(\theta^* \epsilon + \frac{1}{n} \sum_{i=1}^n (g_1(u_i) + g_2(l_i)) \right), \quad (2.26)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \\ \epsilon \geq 0 & \text{for } i = 1, 2, \dots, n. \end{cases}$$

Following the same systematic approach used for ϵ -SVR, we now derive the dual problem for θ^* -SVR within our unified framework.

2.3.1 Lagrangian and KKT Conditions for θ^* -SVR

Since this is a constrained optimization problem, we can again apply the Lagrange multipliers method (1.5.1) for this primal problem.

$$\begin{aligned}
L(\mathbf{w}, b, u_i, l_i, \epsilon, \theta^*, \alpha, \beta, \gamma, \delta, \lambda) &= \frac{1}{2} \|\mathbf{w}\|^2 + C\theta^*\epsilon + \frac{C}{n} \sum_{i=1}^n (g_1(u_i) + g_2(l_i)) \\
&+ \sum_{i=1}^n \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) \\
&+ \sum_{i=1}^n \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) \\
&- \sum_{i=1}^n \gamma_i u_i - \sum_{i=1}^n \delta_i l_i - \lambda \epsilon,
\end{aligned} \tag{2.27}$$

where $\alpha_i \geq 0$, $\beta_i \geq 0$, $\gamma_i \geq 0$, $\delta_i \geq 0$ and $\lambda \geq 0$ are Lagrange multipliers for $i = 1, \dots, n$. After that, take partial derivatives of L with respect to the primal variables $\mathbf{w}, b, \epsilon, u_i, l_i$ yields the stationary condition.

$$\begin{cases} \frac{\partial L}{\partial \mathbf{w}} = 0 \\ \frac{\partial L}{\partial b} = 0 \\ \frac{\partial L}{\partial \epsilon} = 0 \\ \frac{\partial L}{\partial u_i} = 0 \\ \frac{\partial L}{\partial l_i} = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{w} = \sum_{i=1}^n (\alpha_i - \beta_i) \mathbf{x}_i, \\ \sum_{i=1}^n (\alpha_i - \beta_i) = 0, \\ C\theta^* - \sum_{i=1}^n (\alpha_i + \beta_i) - \lambda = 0, \\ \frac{C}{n} g'_1(u_i) - \alpha_i - \gamma_i = 0, \\ \frac{C}{n} g'_2(l_i) - \beta_i - \delta_i = 0. \end{cases} \tag{2.28}$$

Since we know that $\lambda \geq 0$, we obtain the inequality constraint:

$$\sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*. \tag{2.29}$$

The primal feasibility, dual feasibility, and complementary slackness conditions for θ^* -SVR, respectively, are:

Primal feasibility:

$$\begin{cases} y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i \leq 0, \\ \mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i \leq 0, \\ u_i \geq 0, \quad l_i \geq 0, \\ \epsilon \geq 0, \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.30)$$

Dual feasibility:

$$\begin{cases} \alpha_i \geq 0, \quad \beta_i \geq 0, \\ \gamma_i \geq 0, \quad \delta_i \geq 0, \\ \lambda \geq 0, \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.31)$$

Complementary Slackness:

$$\begin{cases} \alpha_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) = 0, \\ \beta_i (\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) = 0, \\ \gamma_i u_i = 0, \quad \delta_i l_i = 0, \\ \lambda \epsilon = 0, \quad \text{for } i = 1, \dots, n. \end{cases} \quad (2.32)$$

The condition in complementary slackness, $\lambda \epsilon = 0$, is particularly important: it indicates that either the tube width ϵ is zero, or the multiplier λ is zero, which occurs when the constraint $\sum(\alpha_i + \beta_i) = C\theta^*$ is active.

From the above stationarity conditions (2.28) for u_i and l_i , we derive bounds analogous to those in ϵ -SVR:

$$0 \leq \alpha_i \leq \frac{C}{n} g'_1(u_i), \quad 0 \leq \beta_i \leq \frac{C}{n} g'_2(l_i). \quad (2.33)$$

2.3.2 Dual Function and Legendre-Fenchel Conjugates for θ^* -SVR

Substituting the stationarity conditions back into the Lagrangian, we obtain the dual function. Following the same simplification steps as in Section 2.2.4:

$$\begin{aligned} L^* = & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i=1}^n (\alpha_i - \beta_i) y_i \\ & + \frac{C}{n} \sum_{i=1}^n (g_1(u_i) - g'_1(u_i)u_i) + \frac{C}{n} \sum_{i=1}^n (g_2(l_i) - g'_2(l_i)l_i). \end{aligned} \quad (2.34)$$

Moving forward, we eliminate the slack variables using the Legendre-Fenchel conjugate (see Section 2.2.5). From stationary conditions of u_i and l_i (2.28), and noting that $\gamma_i = 0$ when $u_i > 0$ (similarly $\delta_i = 0$ when $l_i > 0$), we can express u_i and l_i in terms of α_i and β_i :

$$u_i = (g'_1)^{-1} \left(\frac{n\alpha_i}{C} \right), \quad l_i = (g'_2)^{-1} \left(\frac{n\beta_i}{C} \right). \quad (2.35)$$

The terms involve g_1 and g_2 then become:

$$\frac{C}{n} (g_1(u_i) - g'_1(u_i)u_i) = -\frac{C}{n} g_1^* \left(\frac{n\alpha_i}{C} \right), \quad (2.36)$$

and similarly for g_2 :

$$\frac{C}{n} (g_2(l_i) - g'_2(l_i)l_i) = -\frac{C}{n} g_2^* \left(\frac{n\beta_i}{C} \right), \quad (2.37)$$

Substituting these conjugate expressions into the dual function, we obtain the complete

dual problem for θ^* -SVR with general loss functions:

$$\begin{aligned} \max_{\alpha, \beta} \quad & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i \\ & - \frac{C}{n} \sum_{i=1}^n \left(g_1^* \left(\frac{n\alpha_i}{C} \right) + g_2^* \left(\frac{n\beta_i}{C} \right) \right), \end{aligned}$$

subject to:

$$\begin{aligned} \sum_{i=1}^n (\alpha_i - \beta_i) &= 0, \\ \sum_{i=1}^n (\alpha_i + \beta_i) &\leq C\theta^*, \\ 0 \leq \alpha_i &\leq \frac{C}{n} g_1'(u_i), \\ 0 \leq \beta_i &\leq \frac{C}{n} g_2'(l_i), \quad \text{for } i = 1, \dots, n. \end{aligned} \tag{2.38}$$

Similar to the ϵ -SVR dual, the θ^* -SVR dual also expresses the problem in terms of the Lagrange multipliers α_i and β_i , which correspond to the upper and lower constraints on the regression function. The key difference lies in the additional constraint, $\sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*$, which explicitly incorporates the parameter θ^* into the optimization process. This constraint allows θ^* to control the fraction of support vectors and the fraction of training errors (margin violations), effectively linking the sparsity of the model and the tolerance of errors. This condition leads us to the following Proposition 1.

Proposition 1. *Suppose θ^* -SVR is applied to some data set, and the resulting ϵ is nonzero. The following statements hold:*

- (i) θ^* is an upper bound on the fraction of errors.
- (ii) θ^* is a lower bound on the fraction of support vectors.

(iii) Suppose the data were generated i.i.d. from a distribution $P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x})$ with $P(y|\mathbf{x})$ continuous and the expectation of the modulus of absolute continuity of its density satisfying $\lim_{\delta \rightarrow 0} \mathbb{E} \in (\delta) = 0$. With probability 1, asymptotically, θ^* equals both the fraction of support vectors and the fraction of errors.

Proof. Let,

$$p^* = \frac{1}{n} \sum_{i=1}^n \mathbb{I}(|y_i - f(\mathbf{x}_i)| > \epsilon),$$

denote the proportion of training observations lying outside the ϵ -insensitive tube (i.e., the fraction of margin errors).

(i) Recall the constraint; $\sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*$. If we consider the left-hand side of this constraint, it represents the total support vector influence. Also, we know that there can never be a set of dual variables α_i and β_i which are both simultaneously nonzero. With that, we can consider the LHS of this constraint implies,

$$\begin{aligned} \sum_{i=1}^n (\alpha_i + \beta_i) &= \sum_{i=1}^n \frac{C}{n} \mathbb{I}(\text{Points out side the tube}) \\ &= \frac{C}{n} \sum_{i=1}^n \mathbb{I}(\text{Points out side the tube}) \\ &= C \cdot \text{Proportion of points outside the tube} \\ &= Cp^* \\ &\leq C\theta^* \quad (\text{this comes from the initial constraints}) \\ &\Rightarrow p^* \leq \theta^*, \quad \text{condition (i) proven.} \end{aligned}$$

(ii) From complementary slackness ($\lambda\epsilon = 0$), $\epsilon > 0$, implies $\lambda = 0$. This condition leads to (2.29) becoming $\sum_{i=1}^n (\alpha_i + \beta_i) = C\theta^*$. Therefore, we can now consider

the condition,

$$\alpha_i + \beta_i \leq \frac{C}{n} \text{ (i.e., the maximum possible contribution for any single support vector).}$$

Accordingly,

$$\sum_{i=1}^n (\alpha_i + \beta_i) \leq \frac{C}{n} \sum_{i=1}^n \mathbb{I}(\text{Point becomes a support vector}),$$

or equivalently:

$$C\theta^* \leq Cp^* \Rightarrow \theta^* \leq p^*,$$

which proves (ii).

(iii) After proving (i) and (ii), we have that our θ^* falls between:

$$\begin{aligned} \text{Proportion of points outside the tube} &\leq \theta^* \leq \text{Proportion of points that become support vectors,} \\ p^* &\leq \theta^* \leq p^*, \end{aligned}$$

our result follows.

□

This insight makes θ^* -SVR interpretable: a single parameter simultaneously controls sparsity, robustness, and the tolerated level of error.

2.4 Obtaining ϵ from θ^* -SVR

In θ^* -SVR, ϵ is a variable in the primal problem (2.26). However, since we solve the dual problem, ϵ disappears from the equations and must be recovered afterwards. The value of ϵ is recovered using the KKT optimality conditions. In particular, observations

that lie exactly on the upper and lower boundaries of the ϵ -insensitive tube provide equations from which the optimal tube width can be computed. This approach was originally proposed by [Schölkopf et al. \(2000\)](#).

If we recall our complementary slackness conditions (2.32), we have two conditions that would help us to solve for ϵ . Those conditions are shown below:

$$\begin{cases} \alpha_i(y_i - \mathbf{w}^\top \mathbf{x}_i - b - \epsilon - u_i) = 0, \\ \beta_i(\mathbf{w}^\top \mathbf{x}_i + b - y_i - \epsilon - l_i) = 0, \end{cases} \quad \text{for } i = 1, \dots, n.$$

For a support vector, if the error is exactly ϵ , then $u_i = 0$ or $l_i = 0$. In this case, for a point $\mathbf{x}_i$ lying exactly on the upper edge, $(0 < \alpha_i < \frac{C}{n}g'_1(u_i))$ and the lower edge $(0 < \beta_i < \frac{C}{n}g'_2(l_i))$ can be expressed as:

$$\begin{cases} y_i - \mathbf{w}^\top \mathbf{x}_i - b = \epsilon, \\ \mathbf{w}^\top \mathbf{x}_i + b - y_i = \epsilon. \end{cases}$$

Let c and d be the indices of support vectors on the upper and lower bounds. Then we have:

$$\begin{cases} y_c - \mathbf{w}^\top \mathbf{x}_c - b = \epsilon, \\ \mathbf{w}^\top \mathbf{x}_d + b - y_d = \epsilon. \end{cases}$$

Solving for ϵ yields:

$$\epsilon = \frac{1}{2} \left[(y_c - y_d) - \mathbf{w}^\top (\mathbf{x}_c - \mathbf{x}_d) \right].$$

For the standard θ^* -SVR and for its kernelized version, we get:

$$\epsilon = \frac{1}{2} \left[(y_c - y_d) - \sum_{i=1}^n (\alpha_i - \beta_i) K(\mathbf{x}_i, \mathbf{x}_c) + \sum_{i=1}^n (\alpha_i - \beta_i) K(\mathbf{x}_i, \mathbf{x}_d) \right].$$

In practice, there may be several support vectors on both boundaries. Therefore, ϵ can be computed for each valid pair of upper and lower boundary support vectors, and the resulting values are averaged to obtain a more numerically stable estimate.

2.5 Application to Specific Loss Functions

Having established the general framework for deriving dual problems in both ϵ -SVR and θ^* -SVR, we now demonstrate the versatility of our approach by applying it to several well-known loss functions.

Example 1. Linear ϵ -Insensitive Loss

The linear ϵ -insensitive loss is the standard loss function in SVR ([Vapnik, 1995a](#)). Under this loss function, $g_1(r) = g_2(r) = r$ with $g'_1(r) = g'_2(r) = 1$. After substituting those terms into (2.24), $H_1(\alpha_i)$ and $H_2(\beta_i)$ become zero, and the final dual problem is given by:

$$\max_{\alpha, \beta} -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i),$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad 0 \leq \alpha_i, \beta_i \leq C, \quad \text{for } i = 1, \dots, n. \quad (2.39)$$

Similarly, following (2.38), the θ^* -SVR dual formulation becomes:

$$\max_{\alpha, \beta} -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i,$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad \sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*, \quad 0 \leq \alpha_i, \beta_i \leq \frac{C}{n}, \quad \text{for } i = 1, \dots, n. \quad (2.40)$$

Example 2. ϵ -Insensitive Pinball Loss

The ϵ -insensitive pinball loss is linear outside the insensitive tube but assigns different penalties to positive and negative residuals. In that case, the upper slack variable u_i receives weight τ , whereas the lower slack variable l_i receives weight $(1 - \tau)$. Referring to our general formulation, the corresponding terms for the g_1 and g_2 of this loss can be expressed as:

$$g_1(r) = \tau r \quad \text{and} \quad g_2(r) = (1 - \tau)r,$$

and their first derivatives become:

$$g'_1(r) = \tau \quad \text{and} \quad g'_2(r) = (1 - \tau).$$

Substituting those terms into (2.24) yields the ϵ -SVR dual problem:

$$\max_{\alpha, \beta} -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i),$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad 0 \leq \alpha_i \leq C\tau, \quad 0 \leq \beta_i \leq C(1 - \tau), \quad \text{for } i = 1, \dots, n. \quad (2.41)$$

Following (2.38) gives the dual form for the θ^* -SVR:

$$\max_{\alpha, \beta} -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i,$$

subject to:

$$\begin{cases} \sum_{i=1}^n (\alpha_i - \beta_i) = 0, & \sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*, \\ 0 \leq \alpha_i \leq \frac{C}{n}\tau, & 0 \leq \beta_i \leq \frac{C}{n}(1 - \tau), \end{cases} \quad \text{for } i = 1, \dots, n. \quad (2.42)$$

Example 3. ϵ -SVR with Huber Loss

Huber loss is also one of the most widely used loss functions in regression tasks. Let's extend our found method for this loss and see how it behaves in the ϵ -SVR case. Based on (1.13), the loss can be expressed as follows:

$$g_1(r) = g_2(r) = \begin{cases} \frac{1}{2}r^2, & 0 \leq r \leq \delta, \\ \delta r - \frac{1}{2}\delta^2, & r > \delta. \end{cases}$$

The derivatives are obtained by differentiating each piece:

$$g'_1(r) = g'_2(r) = \begin{cases} r, & 0 \leq r \leq \delta, \\ \delta, & r > \delta. \end{cases}$$

Once we evaluate both first derivatives at 0, we get $g'_1(0) = g'_2(0) = \delta$. Also, following the derivations procedure for $H_1(\alpha_i)$ and $H_2(\beta_i)$, we get:

$$H_1(\alpha_i) = \frac{1}{2} \left(\frac{\alpha_i}{C} \right)^2 - \left(\frac{\alpha_i}{C} \right)^2 = -\frac{1}{2} \left(\frac{\alpha_i}{C} \right)^2 \text{ and } H_2(\beta_i) = -\frac{1}{2} \left(\frac{\beta_i}{C} \right)^2 \text{ similarly.}$$

By combining all the results, the final dual problem for ϵ -SVR associated with Huber loss yields:

$$\begin{aligned} \max_{\alpha, \beta} \quad & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i) \\ & - \frac{1}{2C} \sum_{i=1}^n (\alpha_i^2 + \beta_i^2), \end{aligned}$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad 0 \leq \alpha_i, \beta_i \leq C\delta, \quad \text{for } i = 1, \dots, n. \quad (2.43)$$

Similarly, we can derive the dual of θ^* -SVR.

Example 4. θ^* -SVR with L_2 -Loss

In SVR, L_2 loss (1.12) is often implemented as the squared ϵ -insensitive loss. A key advantage of this approach is that it makes the optimization problem strictly convex, leading to a more mathematically tractable formulation and a unique solution. Based on our formulation the terms g_1 and g_2 are expressed as:

$$g_1(r) = g_2(r) = \frac{r^2}{2} \quad \text{with} \quad g'_1(r) = g'_2(r) = r.$$

Once we substitute these back into (2.38), the corresponding dual θ^* -SVR would be derived as:

$$\max_{\alpha, \beta} -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \frac{1}{2nC} \sum_{i=1}^n (\alpha_i^2 + \beta_i^2),$$

subject to:

$$\sum_{i=1}^n (\alpha_i - \beta_i) = 0, \quad \sum_{i=1}^n (\alpha_i + \beta_i) \leq C\theta^*, \quad \alpha_i, \beta_i \geq 0, \quad \text{for } i = 1, \dots, n. \quad (2.44)$$

Referring to (2.24), the dual of ϵ -SVR can be obtained.

In this chapter, we developed a unified mathematical framework for deriving dual problems in SVR, extending the existing convex analysis literature (Rockafellar, 1970; Boyd and Vandenberghe, 2004). While dual formulations have been derived for specific loss functions (e.g., ϵ -insensitive, Huber, and pinball), a unified method linking the primal and dual problems through conjugate functions remained limited. The proposed framework addresses this gap and provides the foundation for developing new loss functions. Building on this framework and motivated by the need for a loss function that is more robust to outliers while retaining the desirable properties of the ϵ -insensitive formulation, the next chapter introduces a novel class of smooth asymmetric ϵ -insensitive loss functions for SVR.

Chapter 3

SVR with a Novel Smooth Asymmetric ϵ -insensitive Loss Function

Although existing symmetric and asymmetric ϵ -insensitive loss functions perform well in many applications, they can be sensitive to extreme outliers, which may adversely affect the fitted SVR model. Motivated by the limitations of existing loss functions in SVR and to provide an application of our developed framework in Chapter 2, we now introduce a novel class of robust asymmetric ϵ -insensitive loss functions for SVR analysis.

3.1 Mathematical Formulation

In this chapter, we introduce a new loss function, denoted by $\rho_{\tau}^{\epsilon,k}(\cdot)$, which is designed to extend the standard ϵ -insensitive loss function to a more robust framework. This loss function is designed to lie below the standard ϵ -insensitive pinball loss while preserving convexity and retaining smooth logistic-based transitions and flexible asymmetry. This improved structure equips the model to handle heavy-tailed errors more effectively, improves robustness to outliers, and provides greater adaptability through the parameters

τ and k . Our proposed loss function,

$$\rho_{\tau}^{\epsilon,k}(t) = \begin{cases} \frac{(1-\tau)}{k} \cdot \log\left(\frac{1+e^{-k(t+\epsilon)}}{2}\right), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \frac{\tau}{k} \cdot \log\left(\frac{1+e^{k(t-\epsilon)}}{2}\right), & t > \epsilon, \end{cases} \quad (3.1)$$

affords a more stable and flexible alternative to the traditional ϵ -insensitive loss, especially in applications where over- and under-estimation have different consequences.

To better understand the behavior of $\rho_{\tau}^{\epsilon,k}(t)$, as a function of τ , Figure 3.1 plots the loss against the residual t for different τ configurations. When $\tau < 0.5$, the loss is asymmetric and penalizes negative residuals ($t < 0$) more than positive ones. Also, it is useful when underestimation is more tolerable than overestimation. In contrast, when $\tau > 0.5$, positive residuals ($t > 0$) are penalized more than negative residuals, and overestimation is more tolerable here. For $\tau = 0.5$, the loss is symmetric, and overestimates and underestimates are penalized equally.

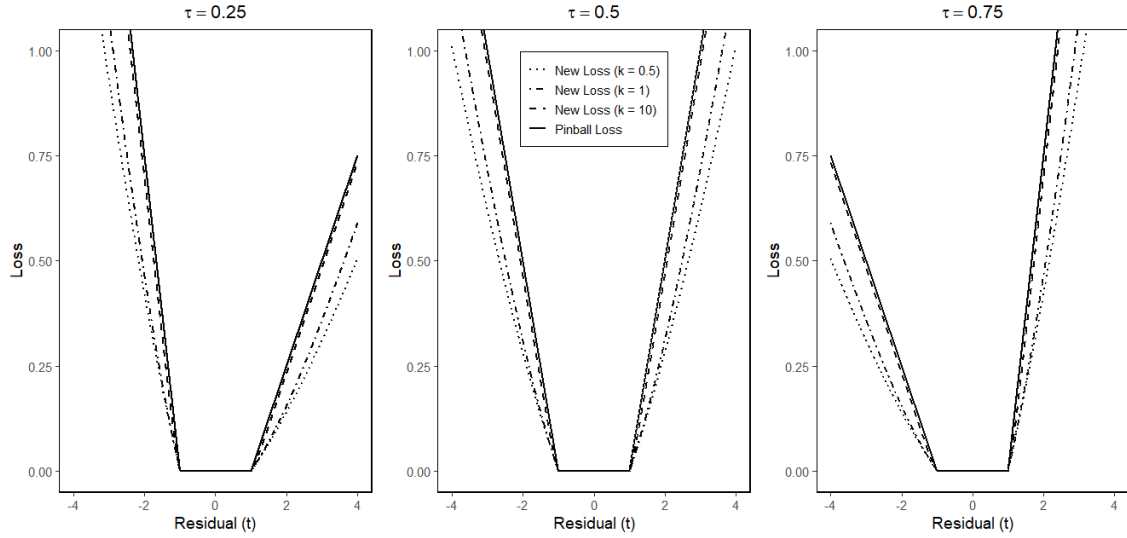


Figure 3.1: Behavior of ϵ -insensitive pinball loss (solid) and ϵ -insensitive proposed loss (dashed lines) for $\epsilon = 1$.

In this proposed loss function, the parameter k is a smoothing factor. Figure 3.1 clearly demonstrates the effect of k . Within the interval $[-1, 1]$, the loss is zero due to the ϵ -insensitive property of ignoring small errors. Beyond this region, the loss increases with the residual, and the rate of this increase is governed by the parameter k . For smaller values of k ($k = 0.5$), the loss growth is more gradual and smooth, whereas for larger values of k ($k = 10$), the function becomes sharper and more angular. This demonstrates that k acts as a smoothing parameter, controlling the transition of the loss function outside the ϵ -insensitive zone.

Furthermore, by comparing the introduced loss with the ϵ -insensitive pinball loss used throughout this thesis, the insensitive zone is placed symmetrically at $\pm\epsilon$, as defined in (3.2).

$$\rho_{\tau}^{\epsilon}(t) = \begin{cases} (\tau - 1)(t + \epsilon), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \tau(t - \epsilon), & t > \epsilon, \end{cases} \quad (3.2)$$

The proposed loss is less sensitive to extreme deviations (Figure 3.1). The ϵ -insensitive pinball loss grows linearly outside the ϵ -insensitive region while preserving quantile asymmetry through the parameter τ . In contrast, the proposed loss replaces the linear penalty with a smooth logistic penalty that grows sub-linearly, where the rate of growth is controlled by the smoothing parameter k . This sub-linear growth reduces the sensitivity to outliers while preserving convexity, and the asymmetry parameter τ allows different penalties for overestimation and underestimation.

3.2 Properties of the Proposed Loss

The proposed loss function $\rho_{\tau}^{\epsilon,k}(\cdot)$ possesses several desirable properties that distinguish it from existing loss functions in the SVR literature.

1. Convexity

For $t > \epsilon$, the proposed loss function $\rho_{\tau}^{\epsilon,k}(\cdot)$ reduces to:

$$R(t) = \frac{\tau}{k} \cdot \log \left(\frac{1 + e^{k(t-\epsilon)}}{2} \right).$$

At the boundary point $t = \epsilon$,

$$R(\epsilon) = \frac{\tau}{k} \log \left(\frac{1 + e^0}{2} \right) = 0.$$

The first derivative of $R(t)$ is:

$$R'(t) = \tau \frac{e^{k(t-\epsilon)}}{1 + e^{k(t-\epsilon)}}.$$

Since $\tau > 0$, we have $R'(t) > 0$, when $t > \epsilon$. The second derivative is:

$$R''(t) = k\tau \frac{e^{k(t-\epsilon)}}{(1 + e^{k(t-\epsilon)})^2}.$$

Because $k > 0$, $\tau > 0$, and $e^{k(t-\epsilon)} > 0$, it follows that $R''(t) > 0$, when $t > \epsilon$.

Therefore, $R(t)$ is convex on (ϵ, ∞) . An analogous argument establishes the convexity of the side corresponding to $t < -\epsilon$. Since the loss is zero on $[-\epsilon, \epsilon]$ and is continuous at $t = \pm\epsilon$, it follows that $\rho_{\tau}^{\epsilon,k}(t)$ is globally convex on $\mathbb{R}$.

2. Non-negativity and Insensitive Zone

By construction $\rho_{\tau}^{\epsilon,k}(t) \geq 0$ for all t , and $\rho_{\tau}^{\epsilon,k}(t) = 0$ precisely when $|t| \leq \epsilon$. This ϵ -insensitive flat region is essential in sparse regression: residuals smaller than ϵ do not incur any penalty. Thus, it generalizes the classical Vapnik's ϵ -insensitive loss while preserving convexity.

3. Differentiability and Smoothness

$\rho_\tau^{\epsilon,k}(\cdot)$ is continuously differentiable on $(-\infty, -\epsilon) \cup (-\epsilon, \epsilon) \cup (\epsilon, \infty)$. At the transition points $t = \pm\epsilon$, the function is continuous but not differentiable. However, since $\rho_\tau^{\epsilon,k}(\cdot)$ is convex, its subdifferential exists and is given by:

$$\partial\rho_\tau^{\epsilon,k}(t) = \begin{cases} \{-(1-\tau)\sigma(k(-t-\epsilon))\}, & t < -\epsilon, \\ [-(1-\tau)/2, 0], & t = -\epsilon, \\ \{0\}, & |t| < \epsilon, \\ [0, \tau/2], & t = \epsilon, \\ \{\tau\sigma(k(t-\epsilon))\}, & t > \epsilon. \end{cases}$$

A key advantage of this loss is that it provides a smooth transition in the penalty region compared with the classical ϵ -insensitive and ϵ -insensitive pinball loss functions, both of which exhibit abrupt changes in their gradients at the transition points. This smoother behavior can lead to more stable optimization while preserving the robustness of the ϵ -insensitive formulation.

4. Asymptotic Linearity as a function of t

As $|t| \rightarrow \infty$, the loss grows linearly with slopes reflecting the desired quantile τ :

$$\lim_{t \rightarrow -\infty} \frac{\rho_\tau^{\epsilon,k}(t)}{-t} = 1 - \tau, \quad \lim_{t \rightarrow +\infty} \frac{\rho_\tau^{\epsilon,k}(t)}{t} = \tau.$$

This ensures that, for large residuals, the loss behaves exactly like the standard pinball loss, thereby preserving the theoretical quantile consistency property.

5. Asymptotic Behavior as a function of k

- I. As $k \rightarrow \infty$, the loss function $\rho_\tau^{\epsilon,k}(t)$ converges pointwise to the standard ϵ -insensitive pinball loss function.

$$\lim_{k \rightarrow \infty} \rho_\tau^{\epsilon,k}(t) = \begin{cases} (\tau - 1)(t + \epsilon), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \tau(t - \epsilon), & t > \epsilon. \end{cases} \quad (3.3)$$

Proof. We consider three cases.

Case 1: $t > \epsilon$. Let $a = t - \epsilon > 0$. Then consider:

$$\frac{1}{k} \log \left(\frac{1 + e^{ka}}{2} \right) = a + \frac{1}{k} \log \left(\frac{1 + e^{-ka}}{2} \right).$$

Taking the limits on both sides as $k \rightarrow \infty$, and noting that $\log(1 + e^{-ka}) \rightarrow 0$, we have,

$$\lim_{k \rightarrow \infty} \frac{1}{k} \log \left(\frac{1 + e^{ka}}{2} \right) = a = t - \epsilon.$$

Multiplying by τ yields the desired result.

Case 2: $t < -\epsilon$. Let $b = -t - \epsilon > 0$. Then we have:

$$\frac{1}{k} \log \left(\frac{1 + e^{kb}}{2} \right) = b + \frac{\log(1 + e^{-kb}) - \log 2}{k}$$

$$\xrightarrow{k \rightarrow \infty} b = -t - \epsilon.$$

Multiplying by $(1 - \tau)$ completes this case.

Case 3: $|t| \leq \epsilon$. For all $k > 0$, $\rho_\tau^{\epsilon,k}(t) = 0$, hence the limit is trivially 0. $\square$

II. As $k \rightarrow 0$, the loss function $\rho_\tau^{\epsilon,k}(t)$ converges pointwise to half of the standard ϵ -insensitive pinball loss function.

$$\lim_{k \rightarrow 0} \rho_\tau^{\epsilon,k}(t) = \begin{cases} (\tau - 1) \cdot \frac{t + \epsilon}{2}, & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \tau \cdot \frac{t - \epsilon}{2}, & t > \epsilon. \end{cases} \quad (3.4)$$

Proof. We again consider the three cases.

Case 1: $t > \epsilon$. Let $a = t - \epsilon > 0$ and set $z = ka$. For small z , we use the Taylor series expansion:

$$\begin{aligned} \log\left(\frac{1 + e^z}{2}\right) &= \log\left(1 + \frac{z}{2} + \frac{z^2}{4} + \mathcal{O}(z^3)\right) \\ &= \left(\frac{z}{2} + \frac{z^2}{4}\right) - \frac{1}{2}\left(\frac{z}{2}\right)^2 + \mathcal{O}(z^3) \quad (\text{expansion of } \log(1 + u)) \\ &= \frac{z}{2} + \frac{z^2}{8} + \mathcal{O}(z^3). \end{aligned}$$

Substituting $z = ka$ and taking the limit when $k \rightarrow 0$ yield:

$$\begin{aligned} \frac{1}{k} \log\left(\frac{1 + e^{ka}}{2}\right) &= \frac{a}{2} + \frac{ka^2}{8} + \mathcal{O}(k^2) \\ &\xrightarrow{k \rightarrow 0} \frac{a}{2} = \frac{t - \epsilon}{2}. \end{aligned}$$

Multiplying by τ gives the result.

Case 2: $t < -\epsilon$. Let $b = -t - \epsilon > 0$. Applying the same expansion with b :

$$\begin{aligned} \frac{1}{k} \log\left(\frac{1 + e^{kb}}{2}\right) &= \frac{b}{2} + \frac{kb^2}{8} + \mathcal{O}(k^2) \\ &\xrightarrow{k \rightarrow 0} \frac{b}{2} = \frac{-t - \epsilon}{2}. \end{aligned}$$

Multiplying by $(1 - \tau)$ yields the desired expression.

Case 3: $|t| \leq \epsilon$. The function is identically zero for all k , so the limit is 0. $\square$

Combining the asymptotic results with the monotonicity of $\rho_{\tau}^{\epsilon,k}(t)$ with respect to k , we conclude that for every fixed t ,

$$\frac{\rho_{\tau}^{\epsilon}(t)}{2} \leq \rho_{\tau}^{\epsilon,k}(t) \leq \rho_{\tau}^{\epsilon}(t), \quad k > 0,$$

where $\rho_{\tau}^{\epsilon}(t)$ is the standard ϵ -insensitive pinball loss function.

3.3 Optimization Formulation for ϵ -SVR

Following the general optimization framework presented in Chapter 2, we derive the primal and dual optimization problems corresponding to the proposed loss function. For convenience, the loss function is first rewritten in the following equivalent form:

$$\rho_{\tau}^{\epsilon,k}(t) = \begin{cases} (1 - \tau) \cdot h_k(-t - \epsilon), & t < -\epsilon, \\ 0, & |t| \leq \epsilon, \\ \tau \cdot h_k(t - \epsilon), & t > \epsilon, \end{cases} \quad (3.5)$$

where $h_k(r) = \frac{1}{k} \log \left(\frac{1 + e^{kr}}{2} \right)$, $r \geq 0$, $k > 0$ and $t = y - f(\mathbf{x})$. This representation highlights the decomposition of the proposed loss function and makes it directly compatible with the general SVR optimization framework presented in Section 2.2.

3.3.1 Primal Problem

Substituting the loss function (3.5) into the general primal formulation (2.2) yields the following optimization problem:

$$\min_{\mathbf{w}, b, u_i, l_i} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \left(\tau h_k(u_i) + (1 - \tau) h_k(l_i) \right), \quad (3.6)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \end{cases} \quad \text{for } i = 1, 2, \dots, n.$$

3.3.2 Dual Problem

Under the general dual problem derivation method introduced in Section 2.2.1, the proposed new loss function (3.5) can be expressed as:

$$g_1(r) = \tau \cdot h_k(r) \quad \text{and} \quad g_2(r) = (1 - \tau) \cdot h_k(r).$$

Then the first derivative satisfies the following:

$$g'_1(r) = \tau \cdot h'_k(r) \quad \text{and} \quad g'_2(r) = (1 - \tau) \cdot h'_k(r), \quad \text{where } h'_k(r) = \frac{e^{kr}}{1 + e^{kr}}.$$

Applying the derivation to $H_1(\alpha_i)$ (2.21) and $H_2(\beta_i)$ (2.23) become,

$$H_1(\alpha_i) = -\frac{\alpha_i}{C\tau} \log\left(\frac{\alpha_i}{C\tau}\right) - \left(1 - \frac{\alpha_i}{C\tau}\right) \log\left(1 - \frac{\alpha_i}{C\tau}\right) - \log(2),$$

and

$$H_2(\beta_i) = -\frac{\beta_i}{C(1 - \tau)} \log\left(\frac{\beta_i}{C(1 - \tau)}\right) - \left(1 - \frac{\beta_i}{C(1 - \tau)}\right) \log\left(1 - \frac{\beta_i}{C(1 - \tau)}\right) - \log(2).$$

Substituting these expressions into the general dual formulation (2.24) results in:

$$\begin{aligned} \max_{\alpha, \beta} \quad & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \frac{C}{k} \sum_{i=1}^n \left[\tau H_1(\alpha_i) + (1 - \tau) H_2(\beta_i) \right] \\ & + \sum_{i=1}^n (\alpha_i - \beta_i) y_i - \epsilon \sum_{i=1}^n (\alpha_i + \beta_i) \end{aligned}$$

subject to:

$$\begin{aligned} & \sum_{i=1}^n (\alpha_i - \beta_i) = 0, \\ & 0 \leq \alpha_i \leq C\tau, \\ & 0 \leq \beta_i \leq C(1 - \tau), \quad \text{for } i = 1, \dots, n. \end{aligned} \tag{3.7}$$

Despite the introduction of asymmetry and smoothing, the resulting optimization problem remains convex and can be extended to nonlinear settings using the kernel trick. Therefore, the dual problem is solved numerically in R using the CVXR package (Fletcher et al., 2020). Since the dual objective is concave and the feasible set is defined by linear constraints, the problem satisfies the disciplined convex programming (DCP) requirements and can be solved using standard convex optimization solvers. The optimization procedure will be discussed in Chapter 4.

3.4 Optimization Formulation for θ^* -SVR

The optimization framework developed in the previous section can be naturally extended to the θ^* -SVR setting. Incorporating the proposed loss function (3.5) into the θ^* -SVR framework preserves the convexity of the optimization problem and yields a formulation that can be efficiently solved with convex optimization techniques. By referring to the general procedure developed in Chapter 2, the corresponding primal and dual optimization problems can easily be presented as follows.

Primal Problem: The primal problem is to do the following optimization:

$$\min_{\mathbf{w}, b, u_i, l_i} \frac{1}{2} \|\mathbf{w}\|^2 + C \left(\theta^* \epsilon + \frac{1}{n} \sum_{i=1}^n \left(\tau h_k(u_i) + (1 - \tau) h_k(l_i) \right) \right), \quad (3.8)$$

subject to:

$$\begin{cases} u_i \geq y_i - f(\mathbf{x}_i) - \epsilon, & u_i \geq 0, \\ l_i \geq f(\mathbf{x}_i) - y_i - \epsilon, & l_i \geq 0, \\ \epsilon \geq 0, & \text{for } i = 1, \dots, n. \end{cases}$$

Dual Problem: By following (2.38), we obtain the corresponding dual form;

$$\begin{aligned} \max_{\alpha, \beta} & -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \beta_i)(\alpha_j - \beta_j) \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \frac{C}{nk} \sum_{i=1}^n \left[\tau H_1(\alpha_i) + (1 - \tau) H_2(\beta_i) \right] \\ & + \sum_{i=1}^n (\alpha_i - \beta_i) y_i, \end{aligned}$$

subject to:

$$\begin{aligned} & \sum_{i=1}^n (\alpha_i - \beta_i) = 0, \\ & \sum_{i=1}^n (\alpha_i + \beta_i) \leq C \theta^*, \\ & 0 \leq \alpha_i \leq \frac{C}{n} \tau, \\ & 0 \leq \beta_i \leq \frac{C}{n} (1 - \tau), \quad \text{for } i = 1, \dots, n. \end{aligned} \quad (3.9)$$

3.5 Visual Comparison: ϵ -insensitive Pinball Loss vs. Proposed Loss

The theoretical properties established in the previous sections demonstrate that the proposed loss function, $\rho_{\tau}^{\epsilon, k}(\cdot)$, is convex, smooth, and suitable for both the ϵ -SVR and

θ^* -SVR optimization. However, these theoretical results alone do not clearly illustrate how this loss influences the data in practice. Therefore, before proceeding to the comprehensive numerical studies in Chapter 4, it is useful to visually compare the behavior of the ϵ -SVR and θ^* -SVR models fitted under the proposed loss function (3.1) and the ϵ -insensitive pinball loss function (3.2).

For this illustration, both losses were incorporated into the optimization formulations developed in previous sections. The resulting dual optimization problems were solved using the `CVXR` package in R. Since this section is intended only to provide a qualitative visual comparison, the optimization procedure, implementation, and tuning parameter selection follow the methodology described in Chapter 4.

In this simulation study, two synthetic datasets, each containing $n = 100$ observations, were generated to represent linear and nonlinear settings. The predictor variable X was independently sampled from a standard normal distribution, $X \sim N(0, 1)$. For the linear setting, the response variable was generated according to the model:

$$Y = 3 + 3X + \xi,$$

where $\xi \sim N(0, 3^2)$ is an independent random error term. For the nonlinear setting, the response variable was generated according to:

$$Y = 2\sin(X) + \xi,$$

where $\xi \sim N(0, 1)$ is an independent random error term. This nonlinear relationship between X and Y makes a dataset for assessing the performance of the RBF kernel. The ϵ -SVR and θ^* -SVR models fitted under the proposed loss function were evaluated with the values of the smoothing parameter, $k = 0.25, 0.5$, and 1 . However, if all these k values were tested during model fitting, only the best-performing one is plotted.

To examine the robustness of the fitted models under the above-mentioned loss functions, outliers were generated from the base population. Approximately 20% of the observations were randomly selected, and their original values were replaced by observations sampled from predefined regions. For the linear setting, upper outliers were generated by sampling $X \sim U(0, 2)$ and $Y \sim U(12, 14)$, while lower outliers were generated by sampling $X \sim U(-2, 0)$ and $Y \sim U(-8, -6)$. For the nonlinear setting, upper outliers were generated from $X \sim U(-2, -1)$ and $Y \sim U(1, 2)$, whereas lower outliers were generated from $X \sim U(1, 2)$ and $Y \sim U(-2, -1)$.

Figure 3.2 represents the ϵ -SVR models fitted under the proposed loss and ϵ -insensitive pinball loss functions with linear and RBF kernels. Here, ϵ is fixed at 0.5.

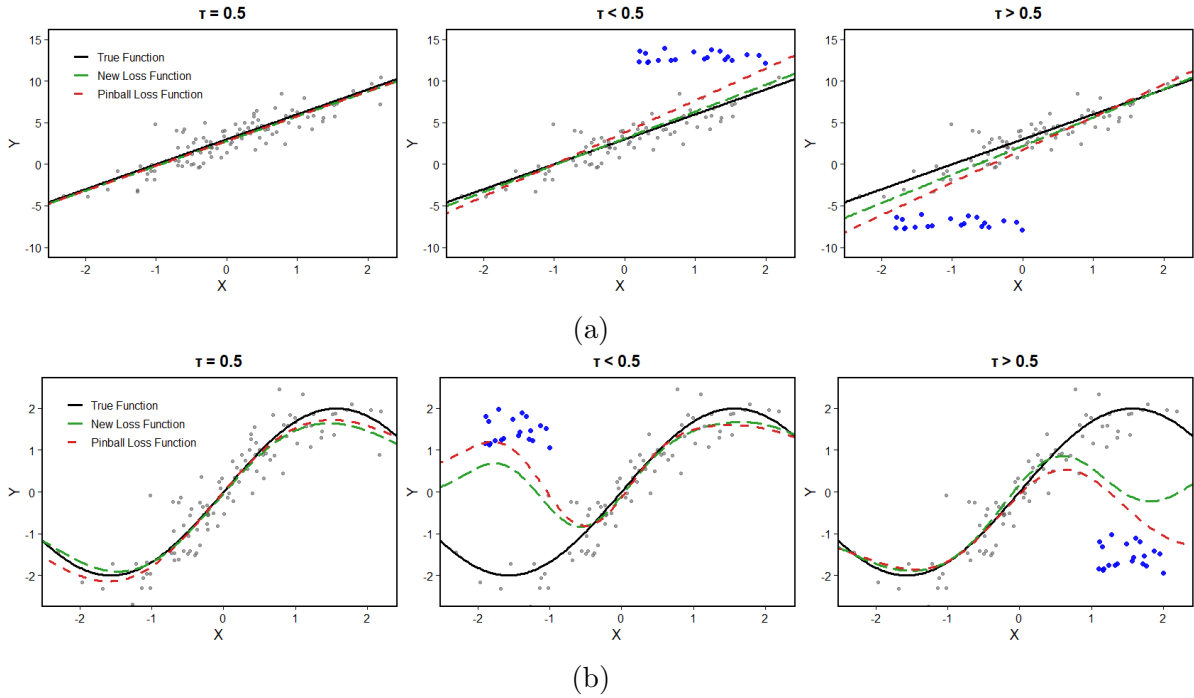


Figure 3.2: Visual comparison of the ϵ -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss function with (a) linear and (b) RBF kernels with $\epsilon = 0.5$. Columns correspond to (left) the symmetric case ($\tau = 0.5$), (middle) upper outliers ($\tau < 0.5$), and (right) lower outliers ($\tau > 0.5$).

In the absence of outliers, the ϵ -SVR models fitted under the both loss functions with a linear kernel closely follow the true regression function. Although the models fitted with the RBF kernel capture the overall trend, the model fitted under the proposed loss more closely follows the true nonlinear pattern, whereas the model fitted under the pinball loss exhibits slight deviations from the underlying function.

The middle column illustrates the case of upper outliers ($\tau < 0.5$). Both loss functions, therefore, adjust the fitted ϵ -SVR model upward in response to the contaminated observations. However, the models fitted under the proposed loss exhibit a substantially smaller upward shift than those fitted under the pinball loss. This difference is particularly evident in the nonlinear setting, where the model under pinball loss is pulled toward the outlying observations, whereas the model under the proposed loss somehow maintains the shape of the underlying true function and returns more quickly to the main data cloud after passing the outliers.

The right column considers lower outliers ($\tau > 0.5$). In this case, larger values of τ reduce the influence of observations below the main data cloud. As expected, both fitted ϵ -SVR models are shifted downward due to the impact of outliers. Nevertheless, the model fitted under the proposed loss again remains noticeably closer to the true function than the model fitted under the pinball loss. In both the linear and nonlinear settings, the fitted models under pinball loss bend more strongly toward the lower outliers, while the models fitted under the proposed loss preserve the overall trend of the observations.

To further examine the behavior of the θ^* -SVR models fitted under proposed and ϵ -insensitive pinball loss functions with linear and RBF kernels, we refer to Figure 3.3, where θ^* is fixed at 0.2.

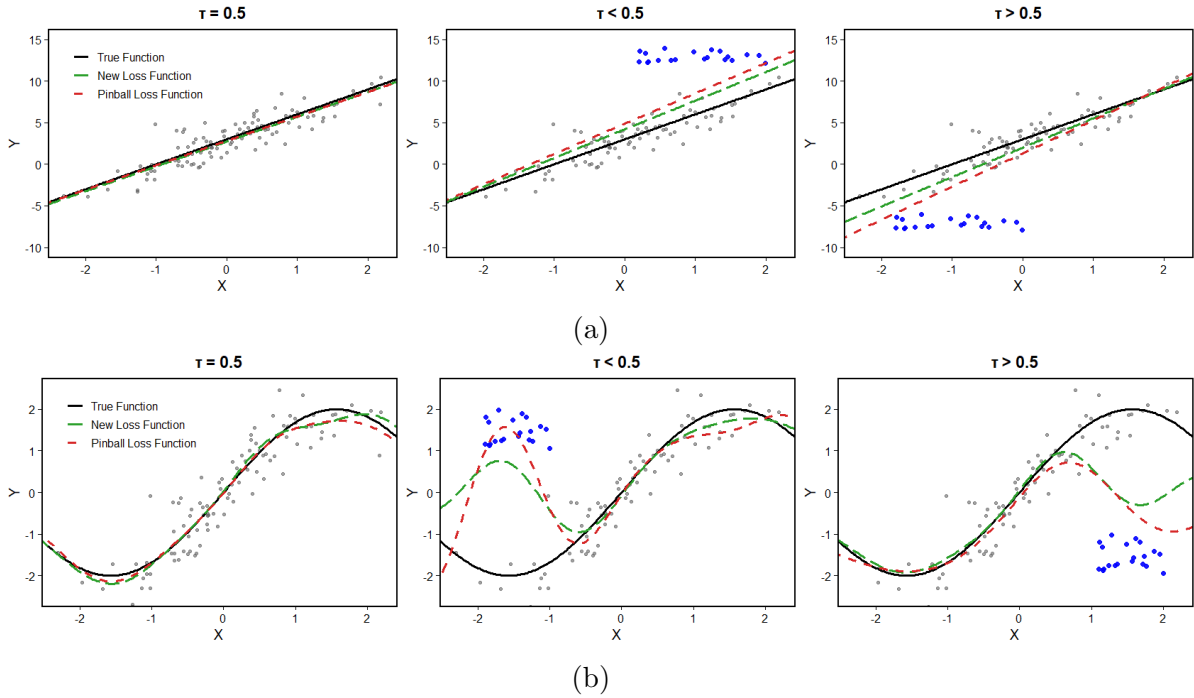


Figure 3.3: Visual comparison of the θ^* -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss function with (a) linear and (b) RBF kernels with $\theta^* = 0.2$. Columns correspond to (left) the symmetric case ($\tau = 0.5$), (middle) upper outliers ($\tau < 0.5$), and (right) lower outliers ($\tau > 0.5$).

The overall behavior of Figure 3.3 is consistent with that observed for the ϵ -SVR models in Figure 3.2. Similar conclusions can therefore be drawn. In particular, the θ^* -SVR models fitted under the proposed loss function more accurately capture the underlying true regression function and exhibit greater robustness to asymmetric outliers than the corresponding models fitted under the ϵ -insensitive pinball loss function. The next chapter presents a comprehensive numerical analysis using simulation studies and real data analysis to further assess its predictive performance and robustness compared to the ϵ -insensitive pinball loss function.

Chapter 4

Simulation Studies and Real Data Analysis

In this chapter, we evaluate the performance of SVR modeling under the proposed loss function using both simulation studies and real data analysis. The simulation studies are conducted using a linear kernel and extended to a nonlinear setting via an RBF kernel, with varying outlier proportions considered to assess robustness. In addition, the proposed method is applied to five real datasets available in `R` to demonstrate its practical applicability.

4.1 Performance Criteria

To evaluate the predictive accuracy and robustness of SVR models under the proposed loss function, we employ three commonly used performance metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2). These metrics are computed to assess the predictive performance of SVR models fitted with both the linear and RBF kernels under varying proportions of outlier contamination. The three metrics are defined as follows:

- **Mean Squared Error (MSE):** The MSE is defined as:

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

and it quantifies the average squared deviation between the predicted ($\hat{y}_i$) and actual values (y_i). Due to the squaring operation, it heavily penalizes large prediction errors. In the context of our study, comparing the MSE of the proposed robust method against standard benchmarks under outlier contamination is crucial; a significantly lower MSE indicates that the proposed loss function successfully mitigates the disproportionate influence of extreme outliers on the model fit.

- **Mean Absolute Error (MAE):** The MAE is defined as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|.$$

Unlike the MSE, the MAE treats all residual magnitudes linearly and is inherently more robust to outliers. It provides a direct and interpretable measure of the typical prediction error in the original scale of the response variable. By evaluating both MSE and MAE simultaneously, we gain deeper insight into the trade-off between overall fit (captured by MSE) and robust central tendency (captured by MAE). A method that maintains a low MAE while substantially reducing MSE in contaminated scenarios would provide evidence of improved robustness.

- **Coefficient of Determination (R^2):** The R^2 statistic is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2},$$

where $(\bar{y}_i)$ is the mean response. This metric represents the proportion of variance in the response variable that is explained by the predictive model. An R^2 value closer to 1 indicates a superior fit. However, it is important to note that traditional R^2 can be notoriously sensitive to outliers, potentially inflating or deflating the value depending on the direction of the extreme observations. In our study, we carefully monitor the behaviour of R^2 alongside MSE and MAE to assess whether the proposed loss function yields a stable and reliable explanation of variance across different real datasets.

4.2 Simulation Studies

In this section, we conduct extensive simulation studies to evaluate the finite-sample performance of linear and nonlinear SVR models fitted under the proposed loss function. All analyses were conducted in R version 4.5.2. To implement the optimization problem, as mentioned in Chapter 3, we employ the CVXR package version 1.0-15 in R, which provides an object-oriented modeling language for Disciplined Convex Programming (DCP) (Fu et al., 2020). CVXR allows users to formulate convex optimization problems in a natural way following mathematical convention, and it seamlessly interfaces with high-performance solvers to ensure numerical stability. The use of CVXR is particularly advantageous in our setting, as the proposed loss function also preserves convexity.

The model is then fitted by solving the corresponding dual optimization problem, namely Example 2 for the ϵ -insensitive pinball loss and the dual formulation developed in Chapter 3 for $\rho_{\tau}^{\epsilon,k}(\cdot)$ loss. The resulting optimal dual variables α and β are used to construct the corresponding SVR function through the kernel representation (1.27) and then the fitted SVR functions were evaluated over a grid of predictor values

and compared with the underlying true regression function.

The simulation study is carried out under both the ϵ -SVR and θ^* -SVR. Further, the experiments include both low-dimensional and higher-dimensional data-generating processes. For the low-dimensional setting, we consider both linear and nonlinear data-generating processes. To thoroughly illustrate the effect of the asymmetry parameter τ , we incorporate different outlier configurations. In the higher-dimensional setting, we consider two distinct noise levels to investigate the stability of the estimation procedure under varying noise conditions. Throughout this study, we use linear and RBF kernels to assess the performance.

For model fitting, we adopted a nested cross-validation procedure combined with grid search for hyperparameter selection (Varma and Simon, 2006; Arlot and Celisse, 2010). The nested cross-validation framework consisted of a 5-fold outer loop for model evaluation and a 3-fold inner loop for hyperparameter tuning. Within each outer iteration, candidate hyperparameter combinations were evaluated using the average validation risk over the three inner folds. For each value of τ , the optimal tuning parameters were first selected using the inner cross-validation procedure. The corresponding SVR model was then refitted on the complete outer training set and evaluated using the held-out outer test fold. This approach prevents optimistic bias in performance estimation that may arise when hyperparameter tuning and model evaluation are performed on the same data (Varma and Simon, 2006), thereby providing a more reliable assessment of predictive performance.

For the proposed loss function, three values of the smoothing parameter are considered, $k \in \{0.25, 0.5, 1\}$, to investigate the influence of the degree of smoothness on model performance. The remaining tuning parameters, C and γ for the RBF kernel, are op-

timized using the nested cross-validation procedure described previously.

For each simulation setting, we repeat the entire data generation, and the resulting dataset is analyzed using the nested cross-validation procedure described above. The resulting MSE and MAE values were averaged across the five outer test folds and used to compare the predictive performance of the fitted SVR models under the proposed loss function with those fitted under the ϵ -insensitive pinball loss in both the ϵ -SVR and θ^* -SVR. The numerical results and their graphical summaries are presented in the following sections.

4.2.1 Linear SVR: Low-Dimensional Settings

For the low-dimensional linear setting, we generate data using a single covariate $\mathbf{X} = (1, x)$ with $x \sim U(0, 5)$ and a response variable Y . The data-generating process follows the form:

$$Y = \mathbf{X}\boldsymbol{\beta} + \mathbf{X}\boldsymbol{\lambda}\xi,$$

where the error term ξ is standard normal. The parameters are set to $\boldsymbol{\beta} = (2, 3)^\top$ and $\boldsymbol{\lambda} = (1, 1)^\top$, which yields a heteroskedastic structure in the error term as the variance scales with the covariate. We generate a sample of size $n = 500$ from this method and consider it as the base population. This population is used as our sample for model fitting. The data-generating mechanism is adapted from the simulation framework of [Loewen and Jafari Jozani \(2026\)](#).

To evaluate the robustness of SVR models under competing loss functions with different contamination mechanisms, four simulation scenarios are considered as described below:

- (i) In the first scenario, we focus on a base population where no outliers are introduced. Here we set the asymmetry parameter τ to 0.5, which allows the model

to estimate the conditional mean of the response distribution.

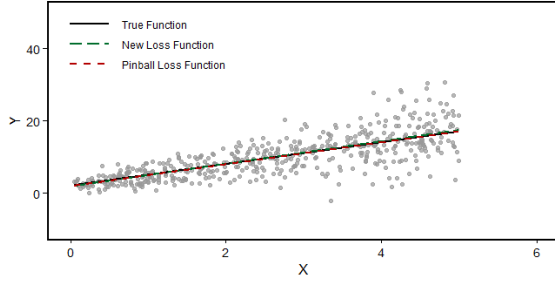
- (ii) Next, we introduce 10% of the observations as outliers on both sides of the base population. Specifically, half are generated from a bivariate normal distribution with mean vector $(4, 40)$, representing the upper outliers, while the other half are generated with mean vector $(4, -5)$, representing the lower outliers. Here, we set τ to 0.5 to evaluate the performance of the proposed method when outliers exist in both directions simultaneously.
- (iii) The third scenario considers 10% outliers that are located towards the top of the base population. These outliers are generated from a bivariate normal distribution with a mean vector $(5, 45)$. For this setting, we examine three different values of the asymmetry parameter: $\tau = 0.1, 0.3$ and 0.5 . These choices allow us to observe how the proposed loss adapts when outliers push the response values upward.
- (iv) Finally, 10% outliers positioned at the bottom of the population. These outliers are generated from a bivariate normal distribution with mean vector $(5, -10)$. In this case, we select τ values of $0.5, 0.7$ and 0.9 to capture the behavior of the loss function when the contamination pulls the response values downward.

We first evaluate the predictive performance of the ϵ -SVR models fitted under the proposed loss function and compare them with the corresponding models fitted under the ϵ -insensitive pinball loss. For this simulation setting, the ϵ parameter was estimated using the procedure described in Section 2.2.6, resulting in an estimated value of approximately $\epsilon = 1.3$. Table 4.1 summarizes the performance of fitted linear ϵ -SVR models under the proposed and pinball loss functions with the four contamination scenarios described above.

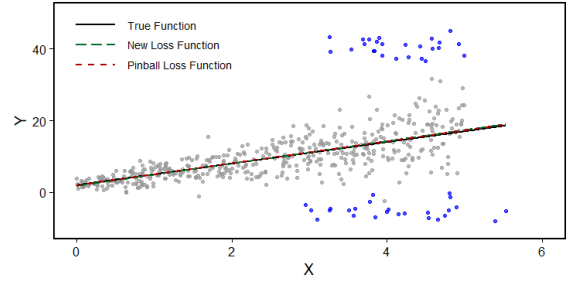
Table 4.1: Comparison of the performance of linear ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

Setting	τ values	New Loss Function		Pinball Loss Function	
		MSE	MAE	MSE	MAE
Base Population	$\tau = 0.5$	0.061 $\pm$ 0.050	0.183 $\pm$ 0.099	0.063 $\pm$ 0.069	0.186 $\pm$ 0.113
10% outliers on both sides of base population	$\tau = 0.5$	0.04 $\pm$ 0.031	0.143 $\pm$ 0.0801	0.049 $\pm$ 0.043	0.173 $\pm$ 0.078
10% outliers above the base population	$\tau = 0.1$	8.298 $\pm$ 1.49	1.915 $\pm$ 0.683	9.497 $\pm$ 0.922	2.098 $\pm$ 0.319
	$\tau = 0.3$	0.902 $\pm$ 0.514	0.763 $\pm$ 0.233	1.089 $\pm$ 0.623	0.829 $\pm$ 0.244
	$\tau = 0.5$	7.25 $\pm$ 1.998	2.102 $\pm$ 0.238	4.522 $\pm$ 4.988	1.72 $\pm$ 0.434
10% outliers below the base population	$\tau = 0.5$	6.636 $\pm$ 1.675	2.647 $\pm$ 1.035	5.840 $\pm$ 1.633	1.527 $\pm$ 0.356
	$\tau = 0.7$	0.843 $\pm$ 0.633	0.671 $\pm$ 0.132	1.074 $\pm$ 0.102	0.927 $\pm$ 0.158
	$\tau = 0.9$	9.078 $\pm$ 2.83	2.647 $\pm$ 1.034	10.979 $\pm$ 0.877	3.695 $\pm$ 0.204

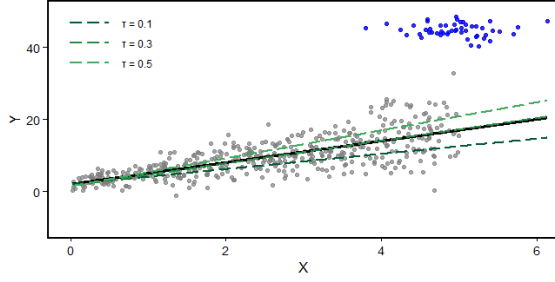
In the absence of outliers, both fitted ϵ -SVR models exhibit comparable predictive performance, although the model fitted under the proposed loss function yields slightly lower MSE and MAE. When outliers are introduced on both sides of the base population, the ϵ -SVR model fitted under the proposed loss function again achieves lower prediction errors than the corresponding model fitted under the ϵ -insensitive pinball loss. For the asymmetric contamination settings, models' performances depend on the choice of the asymmetry parameter τ . When the value of τ is chosen to reflect the direction of the outliers (i.e., $\tau = 0.1, 0.3$, or 0.5 for upper outliers, and $\tau = 0.5, 0.7$, or 0.9 for lower outliers), the ϵ -SVR models fitted under the proposed loss function consistently achieve lower MSE and MAE than the corresponding models fitted under the ϵ -insensitive pinball loss when $\tau \neq 0.5$. However, when $\tau = 0.5$ is used under asymmetric contamination, the ϵ -SVR model fitted under the ϵ -insensitive pinball loss performs slightly better. To implement the numerical results presented in Table 4.1, Figure 4.1 provides a visual comparison of the corresponding fitted ϵ -SVR models under each loss function.



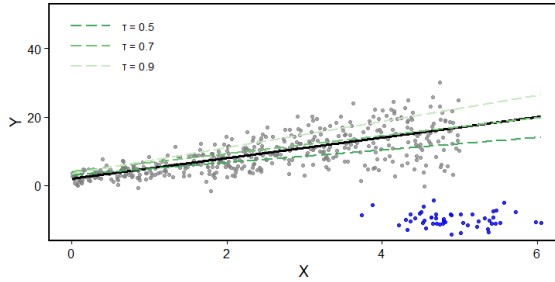
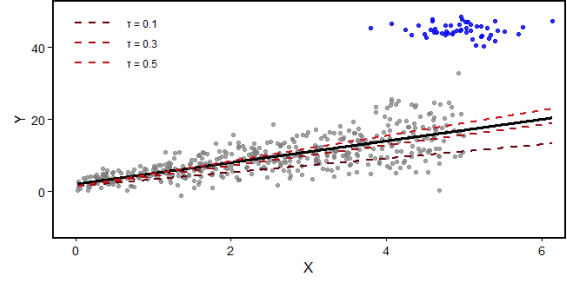
(a) No outliers.



(b) Outliers on both sides.



(c) Upper asymmetric outliers. ϵ -SVR models under the proposed loss function (left) and the ϵ -insensitive pinball loss (right) for $\tau = 0.1, 0.3$ and 0.5 . Solid line: true regression function.



(d) Lower asymmetric outliers. ϵ -SVR models under the proposed loss function (left) and the ϵ -insensitive pinball loss (right) for $\tau = 0.5, 0.7$ and 0.9 . Solid line: true regression function.

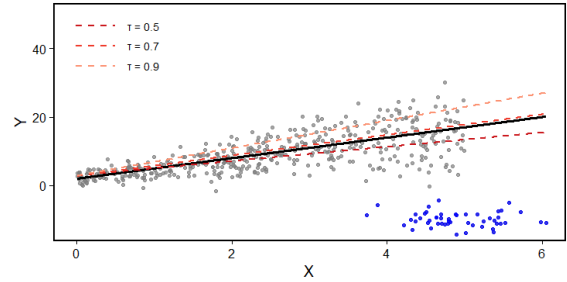


Figure 4.1: Visual comparison of the performance of linear ϵ -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

According to Figure 4.1, in the absence of outliers, both loss approaches produce fitted ϵ -SVR models that closely follow the true regression function. When outliers are introduced on both sides of the data, the fitted models remain similar, although the model fitted under the proposed loss is slightly less affected by the contaminated ob-

servations. Panels (c) and (d) illustrate the influence of the asymmetry parameter τ under upper and lower outlier contamination, respectively. As τ is adjusted to reflect the direction of the outliers, the fitted ϵ -SVR model obtained under the proposed loss moves toward the underlying true regression function while reducing the influence of the contaminated observations. In comparison, the models fitted under the ϵ -insensitive pinball loss exhibit slight deviations from the true function and somehow try to capture the underlying regression structure. These visual observations agree with the numerical results presented in Table 4.1, where the proposed approach generally achieves lower MSE and MAE when an appropriate value of τ is selected.

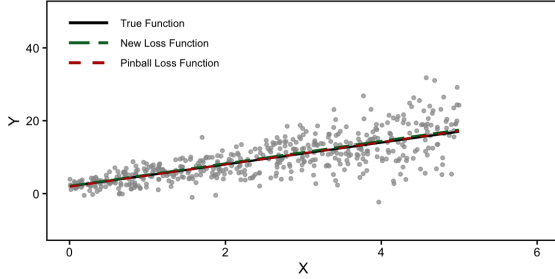
We now extend the analysis to the θ^* -SVR framework, where the parameter $\theta^* = 0.2$. Table 4.2 compares the predictive performance of linear θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss.

Table 4.2: Comparison of the performance of linear θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

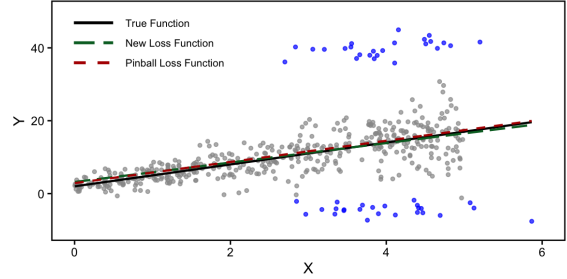
Setting	τ values	New Loss Function		Pinball Loss Function	
		MSE	MAE	MSE	MAE
Base Population	$\tau = 0.5$	0.276 $\pm$ 0.135	0.121 $\pm$ 0.023	0.395 $\pm$ 0.198	0.0187 $\pm$ 0.05
10% outliers on both sides of base population	$\tau = 0.5$	0.984 $\pm$ 0.153	0.638 $\pm$ 0.0175	1.027 $\pm$ 0.0.739	1.145 $\pm$ 0.627
10% outliers above the base population	$\tau = 0.1$	10.537 $\pm$ 2.953	1.783 $\pm$ 1.465	12.248 $\pm$ 1.645	1.813 $\pm$ 0.199
	$\tau = 0.3$	0.53 $\pm$ 0.152	0.546 $\pm$ 0.090	0.923 $\pm$ 0.172	0.867 $\pm$ 0.443
	$\tau = 0.5$	18.825 $\pm$ 3.854	4.38 $\pm$ 0.335	22.740 $\pm$ 5.686	4.913 $\pm$ 1.702
10% outliers below the base population	$\tau = 0.5$	6.247 $\pm$ 1.504	2.061 $\pm$ 0.288	7.258 $\pm$ 2.83	2.296 $\pm$ 0.984
	$\tau = 0.7$	0.584 $\pm$ 0.359	0.626 $\pm$ 0.184	0.703 $\pm$ 0.167	0.714 $\pm$ 0.098
	$\tau = 0.9$	14.248 $\pm$ 1.76	3.568 $\pm$ 0.287	12.435 $\pm$ 2.409	3.085 $\pm$ 1.964

Referring to Table 4.2, the θ^* -SVR models fitted under the proposed loss function generally achieve lower MSE and MAE than the corresponding models fitted under the ϵ -insensitive pinball loss across most contamination scenarios. This improvement is particularly evident when the asymmetry parameter τ is chosen according to the direc-

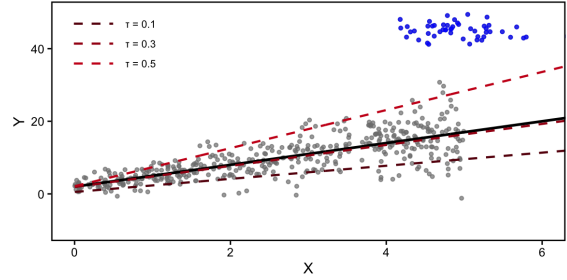
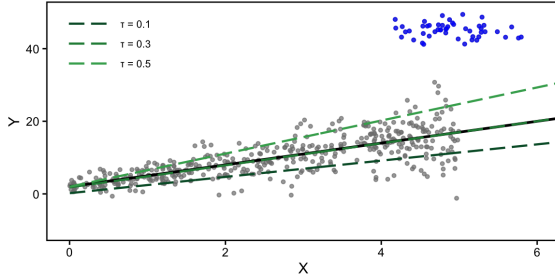
tion of the outliers. In only one case does the model fitted under the ϵ -insensitive pinball loss achieve slightly lower prediction errors. Overall, these results suggest that the proposed loss provides competitive predictive performance within the θ^* -SVR framework. To further illustrate the results in Table 4.2, Figure 4.2 shows the corresponding fitted θ^* -SVR models for each contamination scenario.



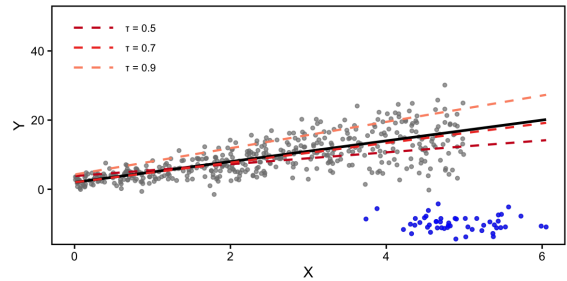
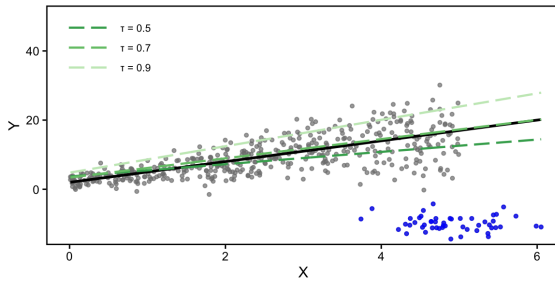
(a) No outliers.



(b) Outliers on both sides.



(c) Upper asymmetric outliers. θ^* -SVR models under the proposed loss function (left) and the ϵ -insensitive pinball loss (right) for $\tau = 0.1, 0.3$ and 0.5 . Solid line: true regression function.



(d) Lower asymmetric outliers. θ^* -SVR models under the proposed loss function (left) and the ϵ -insensitive pinball loss (right) for $\tau = 0.5, 0.7$ and 0.9 . Solid line: true regression function.

Figure 4.2: Visual comparison of the performance of linear θ^* -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

Figure 4.2 visually supports the numerical results presented in Table 4.2. In the situation of (a) and (b), the fitted θ^* -SVR models obtained under both loss functions closely follow the true regression function. Under asymmetric contamination (c and d), however, the fitted θ^* -SVR models become more sensitive to the choice of the asymmetry parameter τ . When τ is selected according to the direction of the outliers, the model fitted under the proposed loss function generally remains closer to the true regression function than the corresponding model fitted under the ϵ -insensitive pinball loss.

Overall, the low-dimensional simulation study demonstrates that the proposed loss function performs competitively under the both linear ϵ -SVR and θ^* -SVR frameworks. While both approaches exhibit similar predictive performance in the absence of outliers, the SVR models fitted under the proposed loss function generally achieve lower MSE and MAE under asymmetric contamination when the asymmetry parameter τ is chosen appropriately.

4.2.2 Non-linear SVR: Low-Dimensional Settings

In the nonlinear low-dimensional setting, we adopt the data-generating mechanism proposed by [Takeuchi et al. \(2006\)](#). Let X denote a scalar covariate generated from the uniform distribution: $X \sim U(-1, 1)$. Conditional on $X = x$, the scalar response variable Y is generated according to:

$$Y = f(x) + \xi,$$

where $\xi \mid X = x \sim N(0, \sigma^2(x))$. The true regression function and the conditional standard deviation are defined as:

$$f(x) = \sin(x) \quad \text{and} \quad \sigma(x) = 0.1 \exp(1 - x), \quad x \in [-1, 1],$$

respectively. Since $\sigma(x)$ depends on x , the generated data exhibit heteroscedasticity.

More generally, for a quantile level $\tau \in (0, 1)$, the true conditional quantile function is:

$$\mu_\tau(x) = f(x) + \sigma(x)\Phi^{-1}(\tau),$$

where Φ^{-1} denotes the quantile function of the standard normal distribution. A sample of size $n = 500$ is generated independently from this model and considered as the base population.

Since the underlying relationship is nonlinear, an RBF kernel is employed for model fitting under both the ϵ -SVR and θ^* -SVR frameworks. Under this nonlinear setting, we consider three distinct scenarios to evaluate the performance of ϵ -SVR and θ^* -SVR models under the proposed loss function. In all scenarios, the asymmetry parameter is fixed at $\tau = 0.5$, corresponding to the conditional median of the response distribution. Three scenarios can be defined as follows:

- (i) The first scenario serves as a baseline, where we consider only the true behavior of the data without any contamination (base population).
- (ii) In the second scenario, we introduce outliers on both sides of the base population, with a total contamination proportion of 10%. Specifically, approximately half of the selected observations with $X \in [-0.3, 0.3]$ are assigned response values Y generated from $U(1.8, 2.2)$, placing them above the base population, while the remaining outliers with $X \in [0.3, 0.7]$ are assigned response values generated from $U(-1.2, -0.8)$, placing them below the base population. Thus, all contaminated observations remain within the original covariate range $[-1, 1]$.
- (iii) The third scenario introduces a 5% contamination consisting of a mixture of local outliers within the original covariate range and far outliers beyond it. For the far

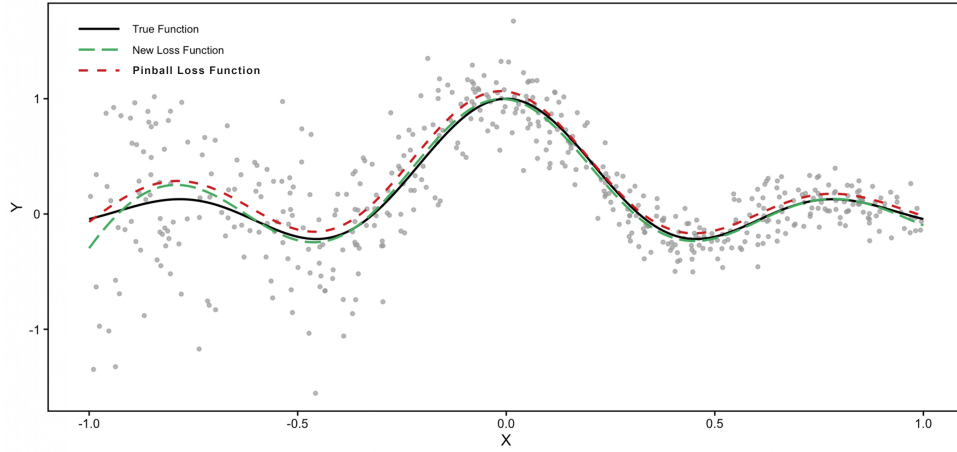
outliers, the covariate values are generated outside $[-1, 1]$, while the responses are placed in the direction opposite to the true regression curve. Specifically, responses are generated from $U(-1.2, -0.8)$ when $f(x) \geq 0$ and from $U(1.8, 2.2)$ when $f(x) < 0$. This setting evaluates the robustness of the proposed method against extreme outliers in both the covariate and response spaces.

We now compare the predictive performance of nonlinear ϵ -SVR models fitted under the proposed loss function and the ϵ -insensitive pinball loss. Here, the parameter ϵ yields an estimated value of approximately $\epsilon = 0.12$ (see Section 2.2.6). The resulting MSE and MAE for the three contamination scenarios are reported in Table 4.3.

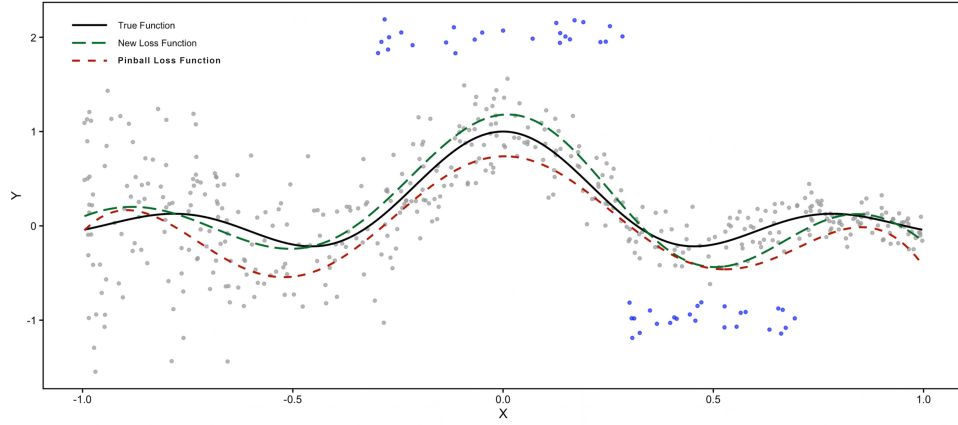
Table 4.3: Comparison of the performance of nonlinear ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

Setting	τ values	New Loss Function		Pinball Loss Function	
		MSE	MAE	MSE	MAE
Base Population	$\tau = 0.5$	0.295 ± 0.062	0.116 ± 0.007	0.432 ± 0.185	0.147 ± 0.082
10% outliers on both sides of the base population	$\tau = 0.5$	1.322 ± 0.012	0.098 ± 0.028	1.578 ± 0.032	0.144 ± 0.0395
5% outliers beyond the base population	$\tau = 0.5$	0.372 ± 0.073	0.289 ± 0.027	0.488 ± 0.053	0.299 ± 0.022

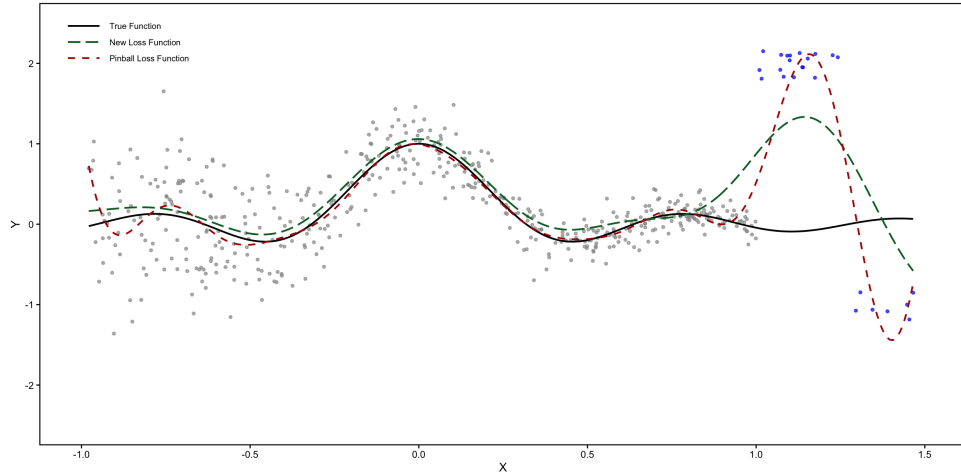
Referring to Table 4.3, the nonlinear ϵ -SVR models fitted under the proposed loss function consistently achieve lower MSE and MAE than the corresponding models fitted under the ϵ -insensitive pinball loss across all three contamination scenarios. The improvement is observed not only for the uncontaminated data but also in the presence of outliers and outliers beyond the original covariate range. To complement the numerical results presented in Table 4.3, Figure 4.3 presents the corresponding fitted nonlinear ϵ -SVR regression functions under the proposed loss function and the ϵ -insensitive pinball loss for each contamination scenario with the RBF kernel.



(a) No outliers.



(b) Upper and lower asymmetric outliers.



(c) Outliers beyond the original covariate range.

Figure 4.3: Visual comparison of the performance of nonlinear ϵ -SVR models fitted under the proposed loss function (Green) and the ϵ -insensitive pinball loss (Red) in the low-dimensional simulation setting. Solid line: true regression function.

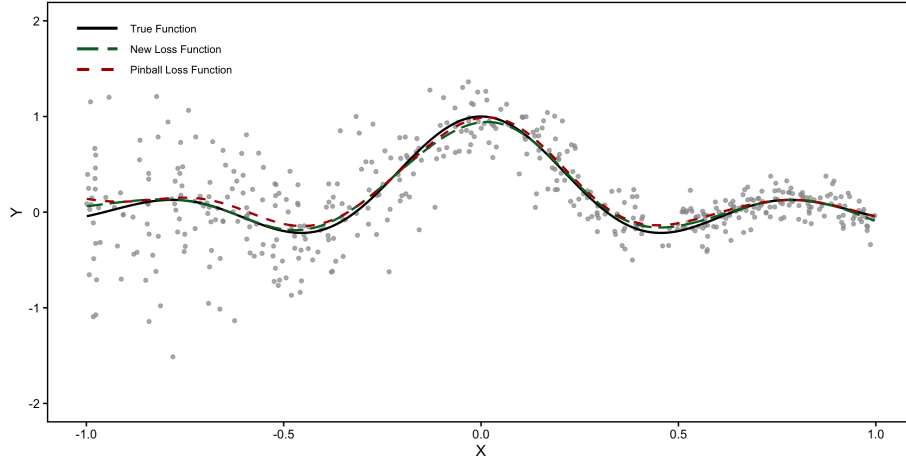
In the absence of outliers, both fitted nonlinear ϵ -SVR models closely follow the true regression function, with the model fitted under the proposed loss function providing a slightly better approximation to the underlying nonlinear pattern. When symmetric outliers are introduced, both fitted models are affected by the contaminated observations; however, the model fitted under the proposed loss function remains closer to the true regression function than the corresponding model fitted under the ϵ -insensitive pinball loss. A similar trend is observed when outliers occur beyond the original covariate range, where the proposed model better preserves the underlying nonlinear structure, while the model fitted under the ϵ -insensitive pinball loss exhibits larger deviations caused by the extreme observations.

Having examined the predictive performance of the nonlinear ϵ -SVR models, we now extend the analysis to the nonlinear θ^* -SVR framework where $\theta^* = 0.2$. The corresponding numerical results are presented in Table 4.4.

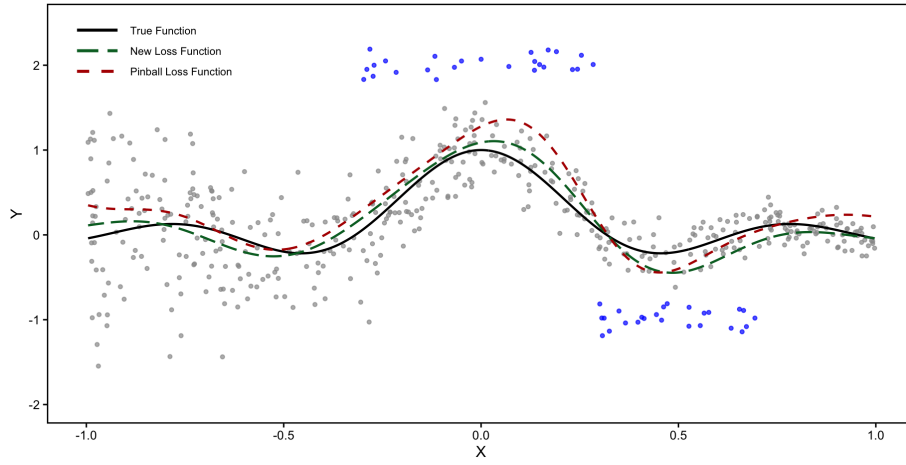
Table 4.4: Comparison of the performance of nonlinear θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss in the low-dimensional simulation setting.

Setting	τ values	New Loss Function		Pinball Loss Function	
		MSE	MAE	MSE	MAE
Base Population	$\tau = 0.5$	0.22 $\pm$ 0.009	0.035 $\pm$ 0.005	0.294 $\pm$ 0.041	0.061 $\pm$ 0.027
10% outliers on both sides of the base population	$\tau = 0.5$	1.521 $\pm$ 0.782	0.439 $\pm$ 0.201	1.748 $\pm$ 0.322	0.572 $\pm$ 0.296
5% outliers beyond the base population	$\tau = 0.5$	0.248 $\pm$ 0.025	0.278 $\pm$ 0.113	0.387 $\pm$ 0.091	0.322 $\pm$ 0.057

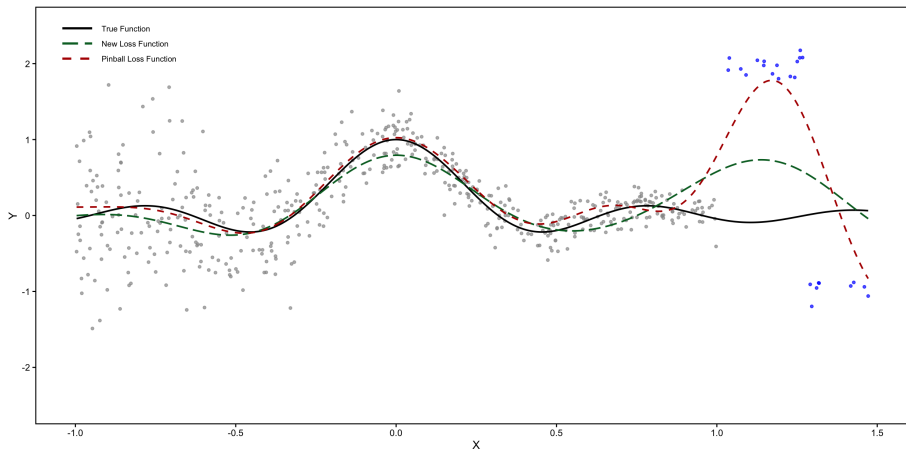
Consistent with the nonlinear ϵ -SVR results, the nonlinear θ^* -SVR models fitted under the proposed loss function achieve lower MSE and MAE than the corresponding models fitted under the ϵ -insensitive pinball loss across all three contamination scenarios. Figure 4.4 visually confirms these findings.



(a) No outliers.



(b) Upper and lower asymmetric outliers.



(c) Outliers beyond the original covariate range.

Figure 4.4: Visual comparison of the performance of nonlinear θ^* -SVR models fitted under the proposed loss function (Green) and the ϵ -insensitive pinball loss (Red) in the low-dimensional simulation setting. Solid line: true regression function.

According to Figure 4.4, the nonlinear θ^* -SVR models fitted under the proposed loss function produce models that remain closer to the true nonlinear structure of the data across all three scenarios. While both loss approaches provide similar fits in the absence of outliers, the differences become more apparent when contamination is introduced. In particular, the fitted models under the proposed loss are less affected by both local outliers and extreme observations beyond the original covariate range, whereas the corresponding models fitted under the ϵ -insensitive pinball loss exhibit larger deviations from the true regression function.

Overall, the nonlinear simulation study demonstrates that the ϵ -SVR and θ^* -SVR models fitted under the proposed loss function provide improved predictive performance and robustness across the different outlier contamination scenarios considered. The graphical results are consistent with the numerical findings and show that the fitted models under the proposed loss more closely follow the underlying nonlinear regression function than the corresponding models fitted under the ϵ -insensitive pinball loss.

4.2.3 Higher Dimensional Settings

In the higher-dimensional setting, we follow the data-generating mechanism proposed by Cherkassky and Ma (2004), which provides a challenging additive target function commonly used to evaluate SVM regression performance in higher dimensions. The true regression function is defined as:

$$r(\mathbf{x}) = 10 \sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5,$$

where the input vector $\mathbf{x} = (x_1, \dots, x_5)^\top$ is uniformly distributed over the hypercube $[0, 1]^5$. This five-dimensional additive function incorporates both nonlinear and linear components, making it a suitable benchmark for evaluating the predictive performance

of regression methods under varying noise conditions. The response variable is generated according to the statistical model:

$$Y = r(\mathbf{x}) + \xi,$$

where ξ represents additive zero-mean noise with standard deviation σ . The sample size is fixed at $n = 500$, consistent with the previous simulation settings.

Two simulation scenarios are considered. In the first scenario, the noise term ξ is generated from a Gaussian distribution with mean zero and standard deviation $\sigma = 0.2$, representing a standard light-tailed setting. In the second scenario, δ follows a Student's t -distribution with five degrees of freedom and is scaled to have the same standard deviation, $\sigma = 0.2$. The heavy-tailed nature of this distribution produces occasional extreme observations, providing a more challenging environment for evaluating robustness. For each noise scenario, two versions of the dataset are considered. The first uses the complete simulated dataset, while the second removes influential observations identified using Cook's distance, thereby providing datasets with and without outliers. The performance of ϵ -SVR and θ^* -SVR models under the proposed loss function is then compared with those under the ϵ -insensitive pinball loss using linear and RBF kernels and the models are evaluated for $\tau \in \{0.1, 0.5, 0.9\}$. Our goal is to assess the predictive performance and robustness of the competing methods under different noise structures and in the presence and absence of influential observations.

The predictive performance of ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss is summarized in Tables 4.5 and 4.6. Following the method described in Section 2.2.6, the estimated value of ϵ is approximately, 0.5. The results are presented separately for the linear and RBF kernels under both Gaussian and Student's t error distributions using the MSE and MAE.

Table 4.5: Prediction performance of the ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for Gaussian and Student's t noise distributions.

Dist.	Outliers	τ	New Loss		Pinball Loss	
			MSE	MAE	MSE	MAE
Gaussian	With Outliers	0.1	9.797 $\pm$ 2.066	2.56 $\pm$ 0.30	14.854 $\pm$ 2.599	3.192 $\pm$ 0.265
		0.5	6.468 $\pm$ 0.297	1.895 $\pm$ 0.057	9.144 $\pm$ 2.157	2.45 $\pm$ 0.361
		0.9	8.904 $\pm$ 1.8	2.337 $\pm$ 0.284	13.912 $\pm$ 1.534	2.975 $\pm$ 0.183
	Without Outliers	0.1	6.290 $\pm$ 0.325	1.896 $\pm$ 0.048	10.72 $\pm$ 0.447	2.703 $\pm$ 0.063
		0.5	6.339 $\pm$ 0.304	1.902 $\pm$ 0.047	6.403 $\pm$ 0.310	1.876 $\pm$ 0.055
		0.9	6.381 $\pm$ 0.275	1.921 $\pm$ 0.046	12.686 $\pm$ 0.87	2.817 $\pm$ 0.113
Student's t	With Outliers	0.1	9.4 $\pm$ 1.405	2.507 $\pm$ 0.216	15.589 $\pm$ 2.081	3.257 $\pm$ 0.242
		0.5	6.566 $\pm$ 0.353	1.916 $\pm$ 0.074	8.196 $\pm$ 1.539	2.302 $\pm$ 0.237
		0.9	9.24 $\pm$ 0.836	2.395 $\pm$ 0.122	11.754 $\pm$ 2.892	2.708 $\pm$ 0.409
	Without Outliers	0.1	6.319 $\pm$ 0.332	1.824 $\pm$ 0.063	11.100 $\pm$ 0.85	2.755 $\pm$ 0.127
		0.5	6.303 $\pm$ 0.339	1.894 $\pm$ 0.047	6.407 $\pm$ 0.297	1.985 $\pm$ 0.286
		0.9	6.377 $\pm$ 0.317	1.911 $\pm$ 0.055	12.766 $\pm$ 0.95	2.821 $\pm$ 0.145

Table 4.6: Prediction performance of the ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for Gaussian and Student's t noise distributions.

Dist.	Outliers	τ	New Loss		Pinball Loss	
			MSE	MAE	MSE	MAE
Gaussian	With Outliers	0.1	0.146 $\pm$ 0.036	0.266 $\pm$ 0.031	0.176 $\pm$ 0.018	0.310 $\pm$ 0.16
		0.5	0.083 $\pm$ 0.022	0.202 $\pm$ 0.039	0.134 $\pm$ 0.019	0.273 $\pm$ 0.017
		0.9	0.134 $\pm$ 0.026	0.250 $\pm$ 0.026	0.227 $\pm$ 0.016	0.344 $\pm$ 0.013
	Without Outliers	0.1	0.157 $\pm$ 0.029	0.231 $\pm$ 0.019	0.208 $\pm$ 0.055	0.312 $\pm$ 0.025
		0.5	0.085 $\pm$ 0.018	0.171 $\pm$ 0.015	0.158 $\pm$ 0.035	0.275 $\pm$ 0.021
		0.9	0.160 $\pm$ 0.038	0.235 $\pm$ 0.018	0.289 $\pm$ 0.087	0.338 $\pm$ 0.038
Student's t	With Outliers	0.1	0.103 $\pm$ 0.027	0.226 $\pm$ 0.020	0.163 $\pm$ 0.019	0.304 $\pm$ 0.016
		0.5	0.063 $\pm$ 0.023	0.171 $\pm$ 0.021	0.118 $\pm$ 0.022	0.258 $\pm$ 0.017
		0.9	0.157 $\pm$ 0.048	0.284 $\pm$ 0.057	0.201 $\pm$ 0.053	0.324 $\pm$ 0.022
	Without Outliers	0.1	0.166 $\pm$ 0.047	0.254 $\pm$ 0.049	0.202 $\pm$ 0.026	0.306 $\pm$ 0.016
		0.5	0.071 $\pm$ 0.012	0.159 $\pm$ 0.007	0.145 $\pm$ 0.017	0.261 $\pm$ 0.010
		0.9	0.138 $\pm$ 0.022	0.229 $\pm$ 0.018	0.239 $\pm$ 0.051	0.315 $\pm$ 0.017

Across both kernels and simulation settings (Tables 4.5 and 4.6), the ϵ -SVR models fitted under the proposed loss function generally achieve lower MSE and MAE than the corresponding models fitted under the ϵ -insensitive pinball loss. This improvement is observed for both uncontaminated and contaminated datasets and remains consistent under Gaussian and heavy-tailed Student's t noise.

We next evaluate the predictive performance of θ^* -SVR models under the proposed loss function. Here we select $\theta^* = 0.2$. Tables 4.7 and 4.8 summarize the results for the linear and RBF kernels under both Gaussian and Student's t noise distributions.

Table 4.7: Prediction performance of the θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for Gaussian and Student's t noise distributions.

Dist.	Outliers	τ	New Loss		Pinball Loss	
			MSE	MAE	MSE	MAE
Gaussian	With Outliers	0.1	10.998 $\pm$ 3.048	2.703 $\pm$ 0.371	14.79 $\pm$ 3.598	3.202 $\pm$ 0.45
		0.5	6.423 $\pm$ 0.673	1.995 $\pm$ 0.135	6.533 $\pm$ 0.125	2.009 $\pm$ 0.050
		0.9	10.275 $\pm$ 3.311	2.434 $\pm$ 0.865	12.168 $\pm$ 3.481	2.751 $\pm$ 0.057
	Without Outliers	0.1	7.946 $\pm$ 3.229	2.188 $\pm$ 0.517	10.171 $\pm$ 0.516	2.641 $\pm$ 0.074
		0.5	6.243 $\pm$ 0.308	1.887 $\pm$ 0.049	6.533 $\pm$ 0.292	1.947 $\pm$ 0.039
		0.9	7.770 $\pm$ 1.350	2.104 $\pm$ 0.255	12.297 $\pm$ 0.559	2.776 $\pm$ 0.064
Student's t	With Outliers	0.1	9.924 $\pm$ 3.471	2.549 $\pm$ 0.430	14.628 $\pm$ 3.172	3.185 $\pm$ 0.399
		0.5	6.426 $\pm$ 0.669	1.996 $\pm$ 0.134	6.771 $\pm$ 0.963	2.058 $\pm$ 0.201
		0.9	11.572 $\pm$ 6.968	2.585 $\pm$ 0.973	12.534 $\pm$ 0.931	2.805 $\pm$ 0.132
	Without Outliers	0.1	7.498 $\pm$ 1.816	2.145 $\pm$ 0.318	10.185 $\pm$ 0.324	2.648 $\pm$ 0.047
		0.5	6.408 $\pm$ 0.172	1.931 $\pm$ 0.070	6.714 $\pm$ 0.528	1.965 $\pm$ 0.103
		0.9	8.783 $\pm$ 2.249	2.267 $\pm$ 0.366	12.292 $\pm$ 0.887	2.776 $\pm$ 0.126

Table 4.8: Prediction performance of the θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for Gaussian and Student's t noise distributions.

Dist.	Outliers	τ	New Loss		Pinball Loss	
			MSE	MAE	MSE	MAE
Gaussian	With Outliers	0.1	0.891 ± 0.144	0.675 ± 0.067	3.386 ± 0.406	1.429 ± 0.66
		0.5	0.364 ± 0.030	0.45 ± 0.028	0.944 ± 0.046	0.721 ± 0.024
		0.9	1.436 ± 0.176	0.747 ± 0.094	1.706 ± 0.061	0.765 ± 0.012
	Without Outliers	0.1	1.113 ± 0.038	0.737 ± 0.008	1.193 ± 0.151	0.782 ± 0.09
		0.5	0.652 ± 0.083	0.517 ± 0.021	1.248 ± 0.912	0.707 ± 0.295
		0.9	2.152 ± 0.257	0.933 ± 0.151	2.168 ± 0.139	0.837 ± 0.023
Student's t	With Outliers	0.1	0.982 ± 0.231	0.720 ± 0.078	3.175 ± 1.984	1.276 ± 0.767
		0.5	0.373 ± 0.039	0.456 ± 0.018	1.412 ± 0.656	0.883 ± 0.231
		0.9	1.484 ± 0.235	0.808 ± 0.192	1.713 ± 0.105	0.769 ± 0.087
	Without Outliers	0.1	0.996 ± 0.119	0.706 ± 0.043	1.109 ± 0.053	0.728 ± 0.017
		0.5	0.661 ± 0.054	0.515 ± 0.009	1.292 ± 0.148	0.703 ± 0.284
		0.9	2.215 ± 0.118	0.706 ± 0.043	2.228 ± 0.288	0.875 ± 0.046

The results indicate that the θ^* -SVR models fitted using the proposed loss function generally outperform the corresponding models fitted using the ϵ -insensitive pinball loss in terms of both MSE and MAE. This improvement is observed across different noise distributions and contamination scenarios, demonstrating the predictive performance of the proposed loss function.

Overall, the results presented in this section demonstrate that the predictive performance of ϵ -SVR and θ^* -SVR models under the proposed loss function is consistently competitive and, in most cases, superior compared with their counterparts under the ϵ -insensitive pinball loss function. The improvement is evident for both linear and RBF kernels across Gaussian and Student's t noise distributions and is particularly pronounced in the presence of heavy-tailed noise and outliers. As expected for the non-linear data-generating mechanism, the RBF kernel yields substantially lower prediction errors than the linear kernel, while the proposed loss further enhances robustness and

predictive accuracy. We next apply our results on several real-world datasets.

4.3 Real Data Analysis

This section evaluates the practical performance of ϵ -SVR and θ^* -SVR models under the proposed loss function using real-world datasets. Results are compared with their counterparts under the ϵ -insensitive pinball loss. We use both the linear and RBF kernels to assess performance under linear and nonlinear relationships. From a computational perspective, using our proposed function requires additional computational effort compared with the ϵ -insensitive pinball loss because the inclusion of the smoothing parameter k increases the complexity of the hyperparameter tuning process and the number of optimization iterations required for convergence.

Here we choose 5 datasets (Table 4.9) which are available in the following R packages: `quantreg`, `alr4`, and `MASS` (Takeuchi et al., 2006). To identify the influence of observations in each dataset, Cook’s distance is used.

- **engel**: The Engel food expenditure dataset, used by Bassett et al. (1982), consists of 235 observations from Belgian working-class households. The data record household income and food expenditure and are frequently used to investigate how spending on food varies across different income levels.
- **crabs**: The crabs dataset, which contains morphological measurements of 200 rock crabs collected from the Gulf of Mexico and is commonly used for multivariate analysis and classification studies. It includes both categorical and continuous variables. The categorical variables are species (sp), indicating whether a crab belongs to the Blue (B) or Orange (O) species, and sex, indicating male or female individuals. The continuous variables consist of five morphometric measurements

in millimeters: frontal lobe size (FL), rear width (RW), carapace length (CL), carapace width (CW), and body depth (BD), which describe size and shape variation among crabs. The data originate from the study of rock crabs by [Campbell and Mahon \(1974\)](#).

- **ufc:** The ufc dataset contains forest inventory measurements from the Upper Flat Creek stand of the University of Idaho Experimental Forest, collected in 1991. This dataset includes both categorical and continuous variables. The only categorical variable is Species, which identifies tree species and includes Douglas-fir (DF), Grand fir (GF), Subalpine fir (SF), Western larch (WL), Western red cedar (WC), and White pine (WP). The continuous variables include Dbh (diameter at breast height, measured 137 cm above the ground in millimetres) and Height (tree height measured in decimetres). In addition, Plot identifies the forest plot number, and Tree represents the tree identifier within each plot ([Weisberg, 2014](#)).
- **gilgais:** The gilgais dataset, available in R, contains soil measurements collected from a line transect survey conducted in gilgai terrain in New South Wales, Australia. Gilgais are naturally occurring shallow depressions in otherwise flat landscapes, and the dataset was originally collected to investigate whether such spatial patterns are reflected in soil properties. Measurements were taken at 365 locations arranged on a linear grid with 4-meter spacing. At each location, soil samples were collected at three depth intervals: 0–10 cm, 30–40 cm, and 80–90 cm below the surface. The dataset includes measurements of pH (a measure of soil acidity), electrical conductivity (e), and chloride content (c) at each depth, resulting in nine numerical variables: pH00, pH30, pH80 for soil pH; e00, e30, e80 for electrical conductivity (mS/cm); and c00, c30, c80 for chloride concentration (ppm). The

dataset originates from [Webster \(1977\)](#) and [Laslett \(1989\)](#), and is documented in [Venables and Ripley \(2002\)](#).

- **cpus:** The cpus dataset, available in the R package MASS, contains performance measurements and hardware characteristics of 209 computer central processing units (CPUs). It consists of several numerical variables describing CPU architecture and system resources, including cycle time (syct), minimum and maximum main memory (mmin and mmax), cache size (cach), and the minimum and maximum number of channels (chmin and chmax). The variable perf represents the published CPU performance as reported in benchmark tests, while estperf is an estimated performance measure derived from hardware characteristics. An additional variable, name, identifies the manufacturer and model of each CPU ([Ein-Dor and Feldmesser, 1987](#); [Venables and Ripley, 2002](#)).

Table 4.9: Summary of the Real-World Datasets Used in the Analysis.

Data Set	Sample Size	No. of Regressors	Y Var.	Outliers	Dropped Vars
engel	235	1	foodexp	10	-
crabs	200	6	CW	12	index
ufc	372	4	Height	15	-
gilgais	365	8	e80	27	-
cpus	209	7	estperf	19	name

The datasets were chosen to avoid missing observations, have outliers, ensure appropriate data types, and maintain a size that allows all models to run effectively. Some categorical variables, those that effectively served as identifiers or created very small groupings of observations, were excluded from consideration. Finally, each dataset was standardized coordinate-wise on the full dataset before splitting, ensuring a zero mean and unit variance prior to running the algorithms.

Consistent with our simulation study, we employ the `CVXR` package available in `R` to solve our convex optimization problems. Model performance was evaluated using nested cross-validation, where the outer 5-fold cross-validation was used to estimate predictive performance and the inner 3-fold cross-validation was used for hyperparameter tuning. The regularization parameter C was selected from the set $\{0.1, 1, 10, 100, 1000, 10000\}$, the asymmetry parameter τ was fixed at three levels (0.1, 0.5, 0.9), and the smoothing parameter k for the proposed loss was selected from values not exceeding 10. The ϵ value is calculated using the procedure described in Section 2.2.6, and select $\theta^* = 0.2$, which controls the fraction of support vectors or the fraction of errors (Schölkopf et al., 2000). For performance evaluation, MSE and MAE are calculated, and R^2 is reported as the mean across folds. All metrics were calculated under both kernel settings.

The real data analysis results are presented in four tables corresponding to the ϵ -SVR and θ^* -SVR. For each SVR method, results are reported separately for the linear and RBF kernels at $\tau = 0.1, 0.5$, and 0.9. Table 4.10 provides an overview of the organization of these tables.

Table 4.10: Organization of the real data analysis results under the ϵ -SVR and θ^* -SVR.

SVR Method	Kernel	$\tau \in \{0.1, 0.5, 0.9\}$
ϵ -SVR	Linear	Table 4.11
	RBF	Table 4.12
θ^* -SVR	Linear	Table 4.13
	RBF	Table 4.14

The experimental results are presented in the following tables.

Table 4.11: Prediction performance of ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for $\tau \in \{0.1, 0.5, 0.9\}$.

τ	Dataset	New Loss			Pinball Loss		
		MSE	MAE	R^2	MSE	MAE	R^2
0.1	engel	0.278 $\pm$ 0.176	0.371 $\pm$ 0.121	0.721	0.308 $\pm$ 0.121	0.411 $\pm$ 0.092	0.691
	crabs	0.003 $\pm$ 0.0001	0.045 $\pm$ 0.002	0.997	0.005 $\pm$ 0.0001	0.056 $\pm$ 0.004	0.995
	ufc	0.428 $\pm$ 0.164	0.522 $\pm$ 0.099	0.571	0.520 $\pm$ 0.212	0.588 $\pm$ 0.141	0.478
	gilgais	0.453 $\pm$ 0.113	0.537 $\pm$ 0.115	0.546	0.478 $\pm$ 0.126	0.443 $\pm$ 0.031	0.520
	cpus	0.129 $\pm$ 0.05	0.155 $\pm$ 0.073	0.872	0.129 $\pm$ 0.062	0.192 $\pm$ 0.068	0.871
0.5	engel	0.273 $\pm$ 0.169	0.337 $\pm$ 0.076	0.726	0.286 $\pm$ 0.181	0.395 $\pm$ 0.166	0.713
	crabs	0.003 $\pm$ 0.0001	0.045 $\pm$ 0.002	0.9967	0.003 $\pm$ 0.0001	0.044 $\pm$ 0.001	0.9968
	ufc	0.334 $\pm$ 0.046	0.459 $\pm$ 0.038	0.665	0.319 $\pm$ 0.046	0.441 $\pm$ 0.039	0.681
	gilgais	0.389 $\pm$ 0.145	0.443 $\pm$ 0.126	0.610	0.310 $\pm$ 0.159	0.372 $\pm$ 0.114	0.690
	cpus	0.120 $\pm$ 0.013	0.2 $\pm$ 0.025	0.880	0.119 $\pm$ 0.052	0.176 $\pm$ 0.068	0.882
0.9	engel	0.321 $\pm$ 0.387	0.303 $\pm$ 0.117	0.677	0.402 $\pm$ 0.304	0.383 $\pm$ 0.078	0.596
	crabs	0.004 $\pm$ 0.0001	0.047 $\pm$ 0.004	0.997	0.005 $\pm$ 0.001	0.056 $\pm$ 0.007	0.995
	ufc	0.565 $\pm$ 0.096	0.593 $\pm$ 0.056	0.434	0.74 $\pm$ 0.67	0.628 $\pm$ 0.232	0.260
	gilgais	0.395 $\pm$ 0.289	0.374 $\pm$ 0.095	0.604	0.484 $\pm$ 0.33	0.434 $\pm$ 0.105	0.515
	cpus	0.103 $\pm$ 0.088	0.240 $\pm$ 0.133	0.897	0.075 $\pm$ 0.028	0.169 $\pm$ 0.02	0.925

Observing Table 4.11, the ϵ -SVR models under the proposed loss function demonstrate competitive predictive performance across the five real-world datasets and generally achieve lower prediction errors than those under the ϵ -insensitive pinball loss, particularly for $\tau = 0.1$ and $\tau = 0.9$. Although models fitted under the pinball loss perform slightly better for some datasets when $\tau = 0.5$, those fitted under the proposed loss remain highly competitive, demonstrating robustness across different levels of asymmetry. We next extend the comparison to the RBF kernel.

Table 4.12: Prediction performance of ϵ -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for $\tau \in \{0.1, 0.5, 0.9\}$.

τ	Dataset	New Loss			Pinball Loss		
		MSE	MAE	R^2	MSE	MAE	R^2
0.1	engel	0.221 $\pm$ 0.040	0.389 $\pm$ 0.034	0.708	0.361 $\pm$ 0.204	0.399 $\pm$ 0.072	0.638
	crabs	0.005 $\pm$ 0.002	0.051 $\pm$ 0.009	0.995	0.006 $\pm$ 0.003	0.053 $\pm$ 0.006	0.994
	ufc	0.357 $\pm$ 0.038	0.469 $\pm$ 0.027	0.652	0.452 $\pm$ 0.083	0.525 $\pm$ 0.049	0.533
	gilgais	0.251 $\pm$ 0.104	0.316 $\pm$ 0.051	0.748	0.205 $\pm$ 0.054	0.296 $\pm$ 0.043	0.794
	cpus	0.39 $\pm$ 0.053	0.128 $\pm$ 0.036	0.611	0.449 $\pm$ 0.061	0.198 $\pm$ 0.042	0.553
0.5	engel	0.229 $\pm$ 0.177	0.293 $\pm$ 0.077	0.770	0.296 $\pm$ 0.208	0.345 $\pm$ 0.112	0.703
	crabs	0.009 $\pm$ 0.002	0.051 $\pm$ 0.008	0.995	0.007 $\pm$ 0.004	0.055 $\pm$ 0.009	0.993
	ufc	0.350 $\pm$ 0.051	0.462 $\pm$ 0.04	0.645	0.377 $\pm$ 0.056	0.478 $\pm$ 0.04	0.622
	gilgais	0.170 $\pm$ 0.083	0.273 $\pm$ 0.058	0.830	0.165 $\pm$ 0.063	0.276 $\pm$ 0.05	0.835
	cpus	0.392 $\pm$ 0.053	0.126 $\pm$ 0.036	0.609	0.377 $\pm$ 0.078	0.196 $\pm$ 0.025	0.624
0.9	engel	0.219 $\pm$ 0.150	0.308 $\pm$ 0.073	0.781	0.258 $\pm$ 0.197	0.338 $\pm$ 0.101	0.741
	crabs	0.005 $\pm$ 0.002	0.051 $\pm$ 0.009	0.995	0.007 $\pm$ 0.003	0.055 $\pm$ 0.007	0.994
	ufc	0.422 $\pm$ 0.125	0.490 $\pm$ 0.077	0.485	0.446 $\pm$ 0.151	0.530 $\pm$ 0.093	0.565
	gilgais	0.173 $\pm$ 0.059	0.304 $\pm$ 0.077	0.826	0.173 $\pm$ 0.08	0.278 $\pm$ 0.068	0.827
	cpus	0.379 $\pm$ 0.088	0.170 $\pm$ 0.141	0.623	0.376 $\pm$ 0.062	0.214 $\pm$ 0.128	0.625

According to Table 4.12, the ϵ -SVR models fitted with the RBF kernel under the proposed loss function demonstrate strong predictive performance across the five real-world datasets and generally achieve lower MSE and MAE values while maintaining competitive R^2 values. Although ϵ -SVR models under the ϵ -insensitive pinball loss perform marginally better for a few datasets and parameter settings, the overall results indicate that the models fitted under the proposed loss function remain highly effective with the RBF kernel. We now extend the comparison to the θ^* -SVR, beginning with the linear kernel results.

Table 4.13: Prediction performance of θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the linear kernel for $\tau \in \{0.1, 0.5, 0.9\}$.

τ	Dataset	New Loss			Pinball Loss		
		MSE	MAE	R^2	MSE	MAE	R^2
0.1	engel	0.171 ± 0.033	0.328 ± 0.040	0.723	0.174 ± 0.08	0.326 ± 0.094	0.719
	crabs	0.020 ± 0.006	0.053 ± 0.012	0.995	0.045 ± 0.002	0.031 ± 0.015	0.979
	ufc	0.492 ± 0.054	0.572 ± 0.036	0.482	0.425 ± 0.034	0.530 ± 0.015	0.552
	gilgais	0.459 ± 0.108	0.421 ± 0.051	0.501	0.418 ± 0.04	0.422 ± 0.046	0.508
	cpus	0.137 ± 0.047	0.228 ± 0.018	0.864	0.138 ± 0.094	0.152 ± 0.091	0.840
0.5	engel	0.084 ± 0.012	0.231 ± 0.019	0.865	0.127 ± 0.055	0.285 ± 0.074	0.795
	crabs	0.024 ± 0.004	0.048 ± 0.013	0.975	0.037 ± 0.002	0.097 ± 0.009	0.996
	ufc	0.291 ± 0.032	0.424 ± 0.027	0.694	0.334 ± 0.066	0.467 ± 0.050	0.648
	gilgais	0.105 ± 0.016	0.250 ± 0.025	0.876	0.109 ± 0.013	0.259 ± 0.017	0.872
	cpus	0.125 ± 0.068	0.166 ± 0.072	0.817	0.107 ± 0.056	0.144 ± 0.062	0.848
0.9	engel	0.172 ± 0.025	0.329 ± 0.032	0.721	0.202 ± 0.082	0.361 ± 0.102	0.673
	crabs	0.018 ± 0.001	0.062 ± 0.011	0.994	0.056 ± 0.002	0.127 ± 0.004	0.980
	ufc	0.515 ± 0.082	0.579 ± 0.051	0.457	0.577 ± 0.17	0.613 ± 0.089	0.392
	gilgais	0.269 ± 0.047	0.372 ± 0.029	0.689	0.212 ± 0.44	0.367 ± 0.045	0.751
	cpus	0.130 ± 0.092	0.219 ± 0.070	0.804	0.140 ± 0.102	0.215 ± 0.048	0.910

Here also we observe very similar behavior across the five real-world datasets under the θ^* -SVR with the linear kernel. Overall, results under our proposed loss achieve lower MSE while maintaining comparable MAE and R^2 across most datasets and τ levels. The θ^* -SVR models fitted under the proposed loss demonstrate particularly strong performance for the *engel* and *crabs* datasets, where it generally produces lower prediction errors than those under the ϵ -insensitive pinball loss across all values of τ . Although θ^* -SVR models under the ϵ -insensitive pinball loss perform slightly better for some datasets, such as *cpus* and *gilgais* under certain parameter settings, these improvements are often marginal. Moreover, in many of these cases, results under the proposed loss exhibit substantially smaller standard deviations for the MSE and MAE, indicating more consistent predictive performance.

Table 4.14: Prediction performance of θ^* -SVR models under the proposed loss function and the ϵ -insensitive pinball loss with the RBF kernel for $\tau \in \{0.1, 0.5, 0.9\}$.

τ	Dataset	New Loss			Pinball Loss		
		MSE	MAE	R^2	MSE	MAE	R^2
0.1	engel	0.191 $\pm$ 0.058	0.337 $\pm$ 0.029	0.692	0.285 $\pm$ 0.222	0.359 $\pm$ 0.104	0.539
	crabs	0.005 $\pm$ 0.001	0.026 $\pm$ 0.016	0.990	0.009 $\pm$ 0.004	0.062 $\pm$ 0.015	0.945
	ufc	0.698 $\pm$ 0.139	0.857 $\pm$ 0.077	0.597	0.690 $\pm$ 0.057	0.786 $\pm$ 0.325	0.628
	gilgais	0.206 $\pm$ 0.06	0.321 $\pm$ 0.053	0.758	0.433 $\pm$ 0.058	0.423 $\pm$ 0.046	0.704
	cpus	0.521 $\pm$ 0.039	0.276 $\pm$ 0.158	0.649	0.492 $\pm$ 0.089	0.137 $\pm$ 0.052	0.698
0.5	engel	0.089 $\pm$ 0.022	0.236 $\pm$ 0.020	0.855	0.145 $\pm$ 0.015	0.264 $\pm$ 0.058	0.765
	crabs	0.003 $\pm$ 0.002	0.052 $\pm$ 0.012	0.993	0.006 $\pm$ 0.005	0.094 $\pm$ 0.032	0.967
	ufc	0.383 $\pm$ 0.044	0.481 $\pm$ 0.019	0.636	0.565 $\pm$ 0.184	0.579 $\pm$ 0.098	0.511
	gilgais	0.094 $\pm$ 0.011	0.228 $\pm$ 0.019	0.852	0.163 $\pm$ 0.036	0.288 $\pm$ 0.042	0.808
	cpus	0.589 $\pm$ 0.077	0.213 $\pm$ 0.18	0.641	0.410 $\pm$ 0.054	0.129 $\pm$ 0.013	0.652
0.9	engel	0.173 $\pm$ 0.032	0.325 $\pm$ 0.045	0.720	0.253 $\pm$ 0.084	0.426 $\pm$ 0.09	0.591
	crabs	0.009 $\pm$ 0.003	0.062 $\pm$ 0.012	0.990	0.005 $\pm$ 0.001	0.021 $\pm$ 0.007	0.994
	ufc	0.953 $\pm$ 0.178	0.728 $\pm$ 0.079	0.584	0.516 $\pm$ 0.155	0.568 $\pm$ 0.093	0.681
	gilgais	0.163 $\pm$ 0.047	0.311 $\pm$ 0.042	0.795	0.275 $\pm$ 0.034	0.455 $\pm$ 0.039	0.677
	cpus	0.539 $\pm$ 0.062	0.209 $\pm$ 0.157	0.683	0.403 $\pm$ 0.039	0.149 $\pm$ 0.133	0.637

Under the proposed loss function, the θ^* -SVR framework with the RBF kernel demonstrates strong predictive performance across the five real-world datasets. For the *engel*, *crabs*, and *gilgais* datasets, the models under the proposed loss generally achieve lower prediction errors across most τ levels. However, for the *ufc* and *cpus* datasets, performance is mixed, with the ϵ -insensitive pinball loss showing lower MSE for certain τ settings. Thus, the results under the proposed loss remain competitive across all datasets, indicating that it is capable of helping to fit more robust models and provide accurate predictions while effectively accommodating nonlinear relationships under the θ^* -SVR. The simulation studies and real data applications together demonstrate the practical applicability of the proposed methodology and motivate the concluding remarks presented in the next chapter.

Chapter 5

Conclusion and Future Work

This chapter provides a summary of the research presented in this thesis. It also discusses the limitations of the study and outlines potential directions for future research to address these limitations and build upon the findings.

5.1 Summary

To begin, this thesis focused on developing a general mathematical framework for SVR that accommodates a broad class of convex loss functions. The proposed framework provides a unified approach for deriving the primal and dual optimization problems associated with both ϵ -SVR and θ^* -SVR using convex analysis and Legendre-Fenchel conjugates. It also shows that many existing SVR loss functions can be obtained as special cases (1). Building on this framework, a novel smooth asymmetric ϵ -insensitive loss function was introduced for SVR. Although the classical ϵ -insensitive loss has been successfully applied in many SVR problems, it treats overestimation and underestimation equally and can be sensitive to extreme observations. These limitations motivated the development of a general class of loss functions for SVR analysis while allowing different penalties for positive and negative prediction errors.

The first chapter reviewed the fundamental concepts of statistical learning and regression methods, with particular attention given to SVR and the role of loss functions. Existing symmetric and asymmetric ϵ -insensitive loss functions were discussed to highlight their advantages and limitations and to motivate the proposed methodology. Also, we introduced the kernel-based extension of SVR, which enables the model to capture nonlinear relationships between predictors and the response.

In Chapter 2, a general mathematical framework for deriving SVR optimization problems using asymmetric but convex loss functions was then developed. Starting from the primal formulation, the corresponding Lagrangian, Karush-Kuhn-Tucker (KKT) conditions, and dual optimization problem were obtained using the Legendre-Fenchel conjugate. This framework provides a unified approach for deriving dual formulations for a broad class of loss functions, rather than treating each loss separately.

Based on the developed framework in Chapter 3, we introduced a new smooth asymmetric ϵ -insensitive loss function. The proposed loss preserves the ϵ -insensitive region while introducing smooth transitions outside the tube through a smoothing parameter k . It also incorporates an asymmetry parameter τ , which allows different penalties for overestimation and underestimation, depending on the application. Several theoretical properties of the proposed loss, including convexity, non-negativity, smoothness, asymptotic behavior, and its relationship with the classical ϵ -insensitive pinball loss, were established. Optimization formulations were then developed for both ϵ -SVR and θ^* -SVR under this loss function. Finally, a visual comparison between the proposed loss and the ϵ -insensitive pinball loss was presented to illustrate their behavior under different outlier contamination scenarios for both the ϵ -SVR and θ^* -SVR frameworks.

Moving to Chapter 4, the performance of the proposed method was evaluated through

extensive simulation studies under both linear and nonlinear settings. Different noise structures, contamination levels, and parameter configurations were considered to compare the performance of ϵ -SVR and θ^* -SVR methods under the proposed loss with their counterparts under the ϵ -insensitive pinball loss. Model performance was assessed using Mean Squared Error (MSE) and Mean Absolute Error (MAE).

Finally, the practical usefulness of the proposed approach was demonstrated using five real-world datasets from different application areas. The results from both the simulation studies and the real data analyses showed that SVR models under the proposed loss function provide competitive predictive performance while offering greater robustness to outliers and increased flexibility through asymmetric error modelling. Furthermore, the proposed optimization framework remained convex and could be efficiently implemented for both ϵ -SVR and θ^* -SVR using kernel methods.

Overall, this study contributes to the SVR literature by developing a unified and general framework for SVR analysis under asymmetric loss functions and introducing a new loss function as an alternative to ϵ -insensitive pinball loss. The proposed methodology provides an effective alternative for regression problems where robustness and asymmetric prediction errors are important considerations.

5.2 Limitations and Future Work

Although the proposed methodology performs well in both simulation studies and real data applications, several limitations remain and provide opportunities for future research.

The proposed method introduces an additional smoothing parameter, k , together with the asymmetry parameter τ and other tuning parameters associated with SVR. While

these parameters increase the flexibility of the model, they also increase the computational effort required for parameter tuning. Therefore, adaptive parameter estimation techniques could reduce the computational cost associated with cross-validation while maintaining good predictive performance.

Furthermore, the proposed loss function provides smooth asymmetric penalties outside the ϵ -insensitive region; its current form uses a single smoothing parameter, k , for both sides of the regression function. A possible extension is to introduce separate smoothing parameters for positive and negative residuals. This would allow the degree of smoothness to differ between underestimation and overestimation and could provide additional flexibility in applications where the two types of prediction errors have different behaviours. Future work could also investigate alternative smooth functions outside the ϵ -insensitive region and compare their properties, robustness, and computational performance with those of the proposed loss.

Also, this study considered linear and radial basis function (RBF) kernels only. Other kernel functions, such as polynomial and sigmoid kernels, may provide better performance for certain applications and deserve further investigation ([Schölkopf and Smola, 2003](#)).

Another limitation of the current study is that it focuses only on SVR analysis under the proposed loss function while all available predictor variables are retained during model fitting. Variable selection was not considered in this research. In many practical applications, especially in high-dimensional datasets, some predictors may be irrelevant or redundant, leading to increased computational cost and reduced model interpretability. Therefore, an interesting direction for future research is to combine the proposed loss function with variable selection techniques such as LASSO, elastic net, or sparse

SVR methods. Such extensions could produce models that are both robust and interpretable while maintaining good prediction accuracy ([Tibshirani, 1996](#))

The current study mainly evaluated prediction accuracy and robustness using simulated data and five real-world datasets. Applying the proposed method to larger datasets from diverse fields such as healthcare, finance, environmental science, and engineering would provide additional evidence of its practical usefulness and generalizability.

Finally, although this thesis focused on SVR, our proposed smooth asymmetric loss function may also be useful in other machine learning methods that rely on convex loss functions. Future studies could investigate its application to other regression methods, such as kernel ridge regression, quantile regression, and neural network-based regression models through appropriate optimization frameworks. Such extensions would help assess the broader applicability of the proposed loss function in predictive modeling problems where robustness and asymmetric prediction errors are important.

Bibliography

- Akay, Ö. and M. Tunçeli (2021). Use of the support vector regression in medical data analysis. *Experimental and Applied Medical Science* 2(4), 242–256. 4
- Aldrich, J. (2005). Fisher and Regression. *Statistical Science* 20(4), 401 – 417. 5
- Allen, M. P. (1997). *The origins and uses of regression analysis*, pp. 1–5. Boston, MA: Springer US. 5
- Anand, P., R. Rastogi, and S. Chandra (2019). A new asymmetric ϵ -insensitive pinball loss function based support vector quantile regression model. 15
- Arlot, S. and A. Celisse (2010, January). A survey of cross-validation procedures for model selection. *Statistics Surveys* 4(none). 78
- Bassett, G., Jr., and R. Koenker (1982). Robust tests for heteroscedasticity based on regression quantiles. *Econometrica* 50(1), 43–62. 96
- Bertsekas, D. P. (1982). *Constrained optimization and Lagrange multiplier methods*. Computer science and applied mathematics. New York: Academic Press. 17
- Bertsekas, D. P. (1999). *Nonlinear programming*. Athena Scientific. 18
- Boş, R. I. and A. Heinrich (2014, November). Regression tasks in machine learning via fenchel duality. *Ann. Oper. Res.* 222(1), 197–211. 27

- Boyd, S. and L. Vandenberghe (2004). *Convex optimization*. Cambridge University Press. 18, 59
- Campbell, N. A. and R. J. Mahon (1974). A multivariate study of variation in two species of rock crab of the genus *leptograpsus*. *Australian Journal of Zoology* 22(3), 417–425. 97
- Chen, C., Y. Li, C. Yan, J. Guo, and G. Liu (2017). Least absolute deviation-based robust support vector regression. *Knowledge-Based Systems* 131, 183–194. 7
- Cherkassky, V. and Y. Ma (2004). Practical selection of svm parameters and noise estimation for svm regression. *Neural networks* 17, 113–126. 46, 47, 91
- Ein-Dor, P. and J. Feldmesser (1987, April). Attributes of the performance of central processing units: a relative performance prediction model. *Commun. ACM* 30(4), 308–317. 98
- Fu, A., B. Narasimhan, and S. Boyd (2020). `jb_cvxri/bj`: An `ijri/ij` package for disciplined convex optimization. *Journal of Statistical Software* 94(14). 69, 77
- Haben, S., S. Arora, G. Giasemidis, M. Voss, and D. Vukadinović Greetham (2021). Review of low voltage load forecasting: Methods, applications, and recommendations. *Applied Energy* 304, 117798. 4
- Hastie, T., R. Tibshirani, and J. H. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). New York, NY: Springer Science & Business Media. 2
- Huber, P. J. (1964). Robust estimation of a location parameter. *Annals of Mathematical Statistics* 35, 492–518. 14, 31

- Hui-zhong, Y., S. Xin-guang, and D. Feng (2005). A new smooth support vector regression based on ϵ -insensitive logistic loss function. In L. Wang, K. Chen, and Y. S. Ong (Eds.), *Advances in Natural Computation*, Berlin, Heidelberg, pp. 25–32. Springer Berlin Heidelberg. 33
- Jain, A. K., M. N. Murty, and P. J. Flynn (1999). Data clustering: a review. *ACM Computing Surveys* 31(3), 264–323. 3
- James, G., D. Witten, T. Hastie, and R. Tibshirani (2021). *An Introduction to Statistical Learning: with Applications in R* (2nd ed.). New York, NY: Springer. 1, 2
- Koenker, R. (2005). *Quantile regression*, Volume 38. Cambridge university press. 7, 8, 10, 31
- Koenker, R. and G. Bassett (1978). Regression quantiles. *Econometrica* 46(1), 33–50. 9, 15
- Künzi, H. P. and W. Krelle (1966). *Nonlinear Programming*. Blaisdell Publishing Company. 18
- Kwok, J. T., K. Hornik, G. Dorffner, and H. Bischof (2001). In *Artificial Neural Networks — ICANN 2001*. Berlin, Heidelberg. 47
- Laslett, G. M. (1989). Kriging and splines: An empirical comparison of their predictive performance in some applications. *Journal of the American Statistical Association* 89, 391–400. 98
- Legendre, A.-M. (1805). *Nouvelles méthodes pour la détermination des orbites des comètes*. Paris: Courcier. Contains the appendix “Sur la méthode des moindres quarrés”. 13

- Loewen, N. and M. Jafari Jozani (2026). Leveraging rank information for robust regression analysis: A nomination sampling approach. *Statistics in Medicine* 45(1-2), e70362. [79](#)
- Mitchell, T. M. (1997). *Machine Learning*. New York, NY: McGraw Hill. [2](#)
- Montgomery, D. C., E. A. Peck, and G. G. Vining (2012). *Introduction to Linear Regression Analysis*. Hoboken, NJ: John Wiley & Sons. [3](#)
- Provost, F. and T. Fawcett (2013). *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. Sebastopol, CA: O'Reilly Media. [3](#)
- Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press. [42](#), [59](#)
- Schölkopf, B. and A. Smola (2003). Kernel methods and support vector machines. *Encyclopedia of Biostatistics* 8(2), 5328–5335. [3](#), [22](#), [107](#)
- Schölkopf, B., A. J. Smola, R. C. Williamson, and P. L. Bartlett (2000). New support vector algorithms. *Neural computation* 12(5), 1207–1245. [48](#), [55](#), [99](#)
- Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Statistics and Computing* 14(3), 199–222. [viii](#), [4](#), [13](#), [22](#), [26](#), [32](#)
- Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Statistics and Computing* 14(3), 199–222. [10](#)
- Takeuchi, I., Q. V. Le, T. D. Sears, and A. J. Smola (2006). Nonparametric quantile estimation. *Journal of Machine Learning Research* 7(45), 1231–1264. [15](#), [85](#), [96](#)
- Tay, F. E. and L. Cao (2001). Application of support vector machines in financial time series forecasting. *Omega* 29(4), 309–317. [4](#)

- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 58(1), 267–288. 108
- Vapnik, V. N. (1995a). *The Nature of Statistical Learning Theory*. New York, NY: Springer. 12, 30, 56
- Vapnik, V. N. (1995b). *The nature of statistical learning theory*. New York: Springer. 46
- Vapnik, V. N. (1998). *Statistical learning theory*. New York: Wiley. 2, 22, 31, 45
- Vapnik, V. N. and A. Y. Chervonenkis (1964). A note on one class of perceptrons. *Automation and Remote Control* 25(3), 774–780. 3
- Varian, H. R. (1975). A bayesian approach to real estate assessment. In S. E. Feinberge and A. Zellner (Eds.), *Studies in Bayesian Econometrics and Statistics*, pp. 195–208. Amsterdam North Holland. 33
- Varma, S. and R. M. Simon (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics* 7, 91 – 91. 78
- Venables, B. and B. Ripley (2002, 01). *Modern Applied Statistics With S*. 98
- Wang, K. and P. Zhong (2014). Robust non-convex least squares loss function for regression with outliers. *Knowledge-Based Systems* 71, 290–302. 16
- Webster, R. (1977, 12). Spectral analysis of gilgai soil. *Australian Journal of Soil Research* 15(3), 191–204. 98
- Weisberg, S. (2014). *Applied linear regression*, Volume 528. John Wiley & Sons. 97