UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

# SR4-Fit: INTERPRETABLE RULE-BASED LEARNING FRAMEWORK FOR INFORMATIVE AND TRUSTWORTHY DECISION-MAKING

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

BY

Shyam Sundar Murali Krishnan
Norman, Oklahoma
2026

Downloaded from SHAREOK. Persistent Handle URL: https://hdl.handle.net/11244/342649

SR4-Fit: INTERPRETABLE RULE-BASED LEARNING FRAMEWORK FOR INFORMATIVE AND TRUSTWORTHY DECISION-MAKING

A DISSERTATION APPROVED FOR THE
SCHOOL OF COMPUTER SCIENCE

BY THE COMMITTEE CONSISTING OF

Dr. Dean Hougen, chair

Dr. Sridhar Radhakrishnan

Dr. Charles Nicholson

Dr. Chao Lan

© Copyright by Shyam Sundar Murali Krishnan 2026
All Rights Reserved.

# Acknowledgements

Foremost, I would like to express my deepest gratitude to my advisor, Dr. Dean Hougen, for his unwavering guidance, support, and encouragement throughout my doctoral journey. His profound expertise, insightful feedback, and exceptional patience have been instrumental in shaping this research and enabling me to achieve my academic goals. His commitment to excellence and mentorship has not only strengthened my scholarly development but has also greatly enriched my overall academic experience. Beyond his professional guidance, I am sincerely grateful for his kindness and genuine support, which have made this journey both meaningful and rewarding.

I would also like to express my sincere gratitude to the esteemed members of my research committee, Dr. Sridhar Radhakrishnan, Dr. Charles Nicholson, and Dr. Chao Lan. Their distinguished scholarship, thoughtful guidance, and supportive presence have greatly enriched my academic journey. Their insights and mentorship have provided invaluable direction and have significantly contributed to the development of this work. I deeply appreciate their time, encouragement, and the intellectual foundation they have helped foster throughout my doctoral studies.

I am deeply grateful to my extended family, friends, and colleagues for their enduring patience, understanding, and unwavering support throughout this jour-

ney. During times when my focus was divided and my presence was limited, your encouragement and belief in me served as a constant source of strength. Your support has been a cornerstone of my resilience and success. I look forward with great eagerness to reconnect and cherish many more shared moments together.

In closing, I reserve my deepest and most heartfelt gratitude for my parents and my brother. Your boundless love, unwavering belief, and constant encouragement have been my greatest source of strength and comfort during the most challenging times. Your support has been my foundation, and this achievement would not have been possible without you.

# Abstract

In many high-stakes applications, machine learning is still dominated by black-box models that require post hoc explanations to justify their predictions. These explanations are often unreliable, since they do not reflect what the model is computing, which limits accountability and trust. A natural alternative is to use models that are interpretable by design. However, existing rule-based approaches, such as RuleFit and decision trees, while transparent, often lack stability and predictive strength, reinforcing the perception of a trade-off between accuracy and interpretability. To address this, we propose Sparse Relaxed Regularized Regression Rule-Fit (SR4-Fit), an algorithm for both classification and regression that produces compact and stable rule sets without sacrificing performance. Using demographic data from the U.S. Census Bureau's American Community Survey, SR4-Fit predicts U.S. House election outcomes with high accuracy and interpretability while uncovering demographic interactions missed by black-box models. We further validate SR4-Fit across multiple commonly used machine learning datasets, six in classification and eight in regression, where it consistently demonstrates strong performance in terms of accuracy, stability, and compactness. Across experiments, the SR4-Fit was comparable in terms of accuracy to black box models and surpassed them in terms of robustness and compactness of the model. The algorithm surpassed existing interpretable rule-based algorithms

such as RuleFit and decision trees in terms of accuracy and robustness and was comparable in terms of compactness, thereby generating stable and interpretable rule sets while maintaining better predictive performance, thus addressing the traditional trade-off between model interpretability and predictive capability.

# Contents

**Acknowledgements** **iii**

**Abstract** **vi**

**List of Figures** **ix**

**List of Tables** **xiv**

**1 Introduction** **1**
1.1 Overview . . . . . 2
1.2 Research Objective . . . . . 5
1.3 Overview of Results . . . . . 6
1.4 Contributions to the Research Community . . . . . 7
1.5 Organization of the Dissertation . . . . . 9

**2 Related Work** **10**
2.1 Classification and Regression Tree (CART) . . . . . 11
2.2 Random Forests . . . . . 12
2.3 Least Absolute Shrinkage and Selection Operator (LASSO) . . . . 14
2.4 Sparse Relaxed Regularized Regression (SR3) . . . . . 15
2.5 RuleFit . . . . . 16
2.6 Dice-Sorenson Index . . . . . 18
2.7 Interpretability Score . . . . . 19

**3 Methodology and Experimental setup** **22**
3.1 SR4-Fit Methodology . . . . . 22
3.1.1 Rule Extraction . . . . . 23
3.1.2 Feature Construction and Optimization Objective . . . . . 24
3.1.3 Model Pruning and Rule Selection . . . . . 25
3.1.4 Classification and Regression Prediction . . . . . 26
3.2 Experimental Setup . . . . . 27

**4 Results** **35**
4.1 Results of U.S House of Representative elections classification . . 36
4.2 Results of U.S House of Representative elections regression (DEM) 45
4.3 Results of U.S House of Representative elections regression (REP) 50
4.4 Results on standard datasets classification . . . . . 56
4.5 Results on standard datasets regression . . . . . 62

**5 Discussion** **94**
5.1 Including party percentages—Classification . . . . . 94
5.2 Without party percentages—Classification . . . . . 97
5.3 Including REP percentage and DEM percentage as label—Regression 100
5.4 Without REP percentage and DEM percentage as label—Regression 102
5.5 Including DEM percentage and REP percentage as label—Regression 104
5.6 Without DEM percentage and REP percentage as label—Regression 106
5.7 Interpretation of SR4-Fit rules for public classification datasets . . 108
5.8 Interpretation of SR4-Fit rules for public regression datasets . . . 111

**6 Conclusions and Future Work** **115**
6.1 Conclusions . . . . . 115
6.2 Future Work . . . . . 118

**Bibliography** **120**

# List of Figures


4.1 Line plots comparing model performance across 30 trials on minimum data, with party percentage features included (left) and excluded (right) . . . 36
4.2 Line plots comparing performance of logistic regression, decision tree, and XGBoost models across 30 trials on minimum data, with party percentage features included (left) and excluded (right) . . . 37
4.3 Line plots comparing model performance across 30 trials on standard data, with party percentage features included (left) and excluded (right) . . . 38
4.4 Line plots comparing performance of logistic regression, decision tree, and XGBoost models across 30 trials on standard data, with party percentage features included (left) and excluded (right) . . . 39
4.5 Line plots comparing model performance across 30 trials on expanded data, with party percentage features included (left) and excluded (right) . . . 40
4.6 Line plots comparing performance of logistic regression, decision tree, and XGBoost models across 30 trials on expanded data, with party percentage features included (left) and excluded (right) . . . 41
4.7 Line plots comparing model performance across 30 trials on previous data, with party percentage features included (left) and excluded (right) . . . 42
4.8 Line plots comparing performance of logistic regression, decision tree, and XGBoost models across 30 trials on previous data, with party percentage features included (left) and excluded (right) . . . 43
4.9 Violin plots comparing model performance distributions across multiple datasets for election classification with party percentage features included . . . 44
4.10 Violin plots comparing performance distributions of logistic regression, decision tree, and XGBoost across multiple datasets for election classification with party percentage features included . . . 45
4.11 Violin plots comparing model performance distributions across multiple datasets for election classification with party percentage features excluded . . . 46

4.12 Violin plots comparing performance distributions of logistic regression, decision tree, and XGBoost across multiple datasets for election classification with party percentage features excluded . . 47
4.13 Distribution of interpretability scores (IPS) across election classification datasets, with party percentage features included (left) and excluded (right) . . . . . 48
4.14 Line plots comparing model performance across 30 trials on minimum data with DEM percentage as the label, with REP percentage included (left) and excluded (right) . . . . . 53
4.15 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on minimum data, with REP percentage included (left) and excluded (right) . . . . . 54
4.16 Line plots comparing model performance across 30 trials on standard data with DEM percentage as the label, with REP percentage included (left) and excluded (right) . . . . . 55
4.17 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on standard data, with REP percentage included (left) and excluded (right) . . . . . 56
4.18 Line plots comparing model performance across 30 trials on expanded data with DEM percentage as the label, with REP percentage included (left) and excluded (right) . . . . . 57
4.19 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on expanded data, with REP percentage included (left) and excluded (right) . . . . . 58
4.20 Line plots comparing model performance across 30 trials on previous data with DEM percentage as the label, with REP percentage included (left) and excluded (right) . . . . . 59
4.21 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on previous data, with REP percentage included (left) and excluded (right) . . . . . 60
4.22 Violin plots comparing performance distributions of models for election regression tasks with DEM percentage as label and REP percentage included . . . . . 61
4.23 Violin plots comparing performance distributions of LASSO regression, decision tree, and XGBoost for election regression tasks with DEM percentage as label and REP percentage included . . . 62
4.24 Violin plots comparing performance distributions of models for election regression tasks with DEM percentage as label and REP percentage excluded . . . . . 63
4.25 Violin plots comparing performance distributions of LASSO regression, decision tree, and XGBoost for election regression tasks with DEM percentage as label and REP percentage excluded . . . 64

4.26 Distribution of interpretability scores (IPS) across election regression datasets for predicting DEM percentage, with REP percentage included versus excluding REP percentage . . . . . 65
4.27 Line plots comparing model performance across 30 trials on minimum data with REP percentage as the label, with DEM percentage included (left) and excluded (right) . . . . . 69
4.28 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on minimum data, with DEM percentage included (left) and excluded (right) . . . . . 69
4.29 Line plots comparing model performance across 30 trials on standard data with REP percentage as the label, with DEM percentage included (left) and excluded (right) . . . . . 70
4.30 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on standard data, with DEM percentage included (left) and excluded (right) . . . . . 70
4.31 Line plots comparing model performance across 30 trials on expanded data with REP percentage as the label, with DEM percentage included (left) and excluded (right) . . . . . 71
4.32 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on expanded data, with DEM percentage included (left) and excluded (right) . . . . . 71
4.33 Line plots comparing model performance across 30 trials on previous data with REP percentage as the label, with DEM percentage included (left) and excluded (right) . . . . . 72
4.34 Line plots comparing performance of LASSO regression, decision tree, and XGBoost across 30 trials on previous data, with DEM percentage included (left) and excluded (right) . . . . . 72
4.35 Violin plots comparing performance distributions of models for election regression tasks with REP percentage as label and DEM percentage included . . . . . 73
4.36 Violin plots comparing performance distributions of LASSO regression, decision tree, and XGBoost for election regression tasks with REP percentage as label and DEM percentage included . . . 74
4.37 Violin plots comparing performance distributions of models for election regression tasks with REP percentage as label and DEM percentage excluded . . . . . 75
4.38 Violin plots comparing performance distributions of LASSO regression, decision tree, and XGBoost for election regression tasks with REP percentage as label and DEM percentage excluded . . . 76
4.39 Distribution of interpretability scores (IPS) across election regression datasets for predicting REP percentage, with DEM percentage included versus excluding DEM percentage . . . . . 76

4.40 Line plots comparing model performance for breast cancer data across 30 trials . . . 78
4.41 Line plots comparing model performance for E. coli data across 30 trials . . . 79
4.42 Line plots comparing model performance for page blocks data across 30 trials . . . 80
4.43 Line plots comparing model performance for Pima Indians data across 30 trials . . . 81
4.44 Line plots comparing model performance for vehicle data across 30 trials . . . 82
4.45 Line plots comparing model performance for yeast data across 30 trials . . . 83
4.46 Violin plots comparing performance distributions of random forest, SVM, RuleFit, and SR4-Fit across multiple benchmark classification datasets . . . 84
4.47 Violin plots comparing performance distributions of logistic regression, decision tree, and XGBoost across multiple benchmark classification datasets . . . 85
4.48 Violin plots comparing distribution of interpretability scores across standard public classification datasets. . . . 86
4.49 Line plots comparing model performance for abalone data across 30 trials . . . 86
4.50 Line plots comparing model performance for bone data across 30 trials . . . 87
4.51 Line plots comparing model performance for diabetes data across 30 trials . . . 87
4.52 Line plots comparing model performance for housing data across 30 trials . . . 88
4.53 Line plots comparing model performance for machine data across 30 trials . . . 88
4.54 Line plots comparing model performance for MPG data across 30 trials . . . 89
4.55 Line plots comparing model performance for ozone data across 30 trials . . . 89
4.56 Line plots comparing model performance for prostate data across 30 trials . . . 90
4.57 Violin plots comparing performance distributions of random forest, SVM, RuleFit, and SR4-Fit across multiple benchmark regression datasets . . . 91
4.58 Violin plots comparing performance distributions of LASSO regression, decision tree, and XGBoost across multiple benchmark regression datasets . . . 92

4.59 Violin plot comparing distribution of interpretability scores across standard public regression datasets . . . . . 93

# List of Tables


4.1 Average Dice–Sorensen Index ± standard deviation across 30 trials for election classification datasets . . . . . 49
4.2 Average number of rules ± standard deviation across 30 trials for election classification datasets . . . . . 50
4.3 Average rule complexity ± standard deviation across 30 trials for election classification datasets. . . . . 51
4.4 Average interpretability score (IPS) ± standard deviation across 30 trials for election classification datasets . . . . . 52
4.5 Average Dice–Sorensen Index ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label) . . . 66
4.6 Average number of rules ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label) . . . . . 67
4.7 Average rule complexity ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label) . . . . . 68
4.8 Average interpretability score (IPS) ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label) . . . . . 68
4.9 Average Dice–Sorensen Index ± standard deviation across 30 trials for the election regression dataset (REP percentage as label) . . . 73
4.10 Average number of rules ± standard deviation across 30 trials for the election regression dataset (REP percentage as label) . . . . . 75
4.11 Average rule complexity ± standard deviation across 30 trials for the election regression dataset (REP percentage as label) . . . . . 77
4.12 Average interpretability score (IPS) ± standard deviation across 30 trials for the election regression dataset (REP percentage as label) 77
4.13 Average Dice–Sorensen Index ± standard deviation across 30 trials for public classification datasets . . . . . 78
4.14 Average number of rules ± standard deviation across 30 trials for public classification datasets . . . . . 79
4.15 Average rule complexity ± standard deviation across 30 trials for public classification datasets . . . . . 80
4.16 Average interpretability score (IPS) ± standard deviation across 30 trials for public classification datasets . . . . . 81

4.17 Average Dice–Sorensen Index ± standard deviation across 30 trials for public regression datasets . . . . . 82
4.18 Average number of rules ± standard deviation across 30 trials for public regression datasets . . . . . 83
4.19 Average rule complexity ± standard deviation across 30 trials for public regression datasets . . . . . 90
4.20 Average interpretability score (IPS) ± standard deviation across 30 trials for public regression datasets . . . . . 93

# Chapter 1

# Introduction

The growing need for transparent and reliable machine learning systems has motivated the development of methods that can balance predictive performance with interpretability. In real-world applications that require critical decision-making, models must not only generate accurate predictions but also provide explanations that allow researchers and practitioners to understand the underlying reasoning behind those predictions. This dissertation addresses this challenge by introducing *Sparse Relaxed Regularized Regression Rule-Fit (SR4-Fit)*, an intrinsically interpretable rule-based learning algorithm designed to produce compact and stable rule sets while maintaining strong predictive performance. The proposed approach is evaluated using demographic data related to U.S. House elections as well as a collection of publicly available benchmark datasets for classification and regression. The following sections present an overview of the research problem, the objectives of the study, a summary of the experimental results, and the contributions of this work to the field of interpretable machine learning.

## 1.1 Overview

Machine learning (ML) models such as random forests (Breiman, 2001), Support Vector Machines (SVMs) (Hearst et al., 1998), XGBoost(Friedman, 2001), and neural networks (McCulloch and Pitts, 1943) have transformed how data-driven problems are approached across science and society. Despite their success, these models are often criticized as black boxes: they capture complex, nonlinear patterns but provide little clarity on how inputs influence predictions. In many domains, this opacity limits the ability to verify, explain, or trust automated decisions, especially when those decisions have real-world consequences (Caruana et al., 2015). These concerns have motivated a growing interest in interpretable machine learning, which aims to design models that not only perform well but can also be understood and validated by people (Molnar, 2020).

Interpretability becomes particularly important in areas where *transparency* and *accountability* are essential, such as the study of U.S. House of Representatives elections. Understanding how demographic and structural factors shape political outcomes is central to evaluating fair representation in a democracy. Electoral behavior depends not only on individual choices but also on how districts are drawn, which can amplify or diminish the influence of certain groups. Demographic characteristics, such as race, education, income, and age, strongly correlate with voting patterns and party preference (Arrington, 2010). Modeling these relationships helps estimate district security, or the likelihood that a political party will retain control in future elections, and can also provide insight into practices such as gerrymandering (Friedman and Holden, 2009). In this setting, models that balance predictive accuracy with interpretability can offer more than just forecasts; they can help explain why elections turn out the way they do.

Black-box models have proven capable of predicting congressional election results with high accuracy (Richardson and Hougen, 2020). Yet the inner workings of these models often remain obscure, making it difficult to identify which demographic patterns drive the predictions. This reflects a well-known tension between accuracy and interpretability (Murdoch et al., 2019): complex models tend to predict well but are hard to explain, whereas simpler models are easier to interpret but may lose predictive strength. Finding methods that achieve both remains an active and important challenge.

One common strategy has been to use *post-hoc explanation* tools such as SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017) and Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016). These approaches provide approximate insights into a model's behavior, such as feature importance or local decision boundaries. While valuable for interpretive analysis, post-hoc explanations can diverge from what the model is actually computing, particularly for nonlinear systems. As a result, they can aid understanding but do not fully solve the transparency problem.

An alternative direction focuses on *intrinsically interpretable model* algorithms whose structure makes their reasoning transparent by design (Rudin, 2019). These include decision trees (Quinlan, 1986), sparse linear models, and rule-based systems. Because the logic behind each prediction is directly encoded in the model, such approaches make it easier to verify results and reason about causal patterns. While intrinsic interpretability does not automatically guarantee higher accuracy, it offers a solid basis for building models that are both reliable and explainable. In contexts like electoral forecasting, this combination supports both empirical rigor and public accountability.

Among intrinsically interpretable methods, rule-based models are particularly

appealing because they express decisions as if–then statements that can be read and understood directly. These models align closely with human reasoning and lend themselves naturally to policy and social analysis. However, existing rule-based techniques vary widely in their design and scope. Some focus only on binary classification (Obregon and Jung, 2023), (Letham et al., 2015), and (Angelino et al., 2018) while others prioritize simplicity to the point of losing predictive depth. In addition, many are tailored to a single task type, either classification or regression, but not both (Bénard et al., 2021). These limitations reinforce the perception that interpretability often comes at the expense of performance.

The *RuleFit* algorithm (Friedman and Popescu, 2008) represented a key attempt to balance this trade-off by combining rule extraction from tree ensembles with linear modeling. While RuleFit offers a useful middle ground, it can struggle with stability, consistency, and predictivity, especially when applied to high-dimensional or noisy data (Yang et al., 2017). The rules it generates can overlap or vary across runs, complicating interpretation and limiting trust in the results. Further refinements are therefore needed to make rule-based modeling more robust and versatile.

To address these issues, we introduce Sparse Relaxed Regularized Regression Rule Fit (SR4-Fit), an intrinsically interpretable algorithm that extends the RuleFit framework using the sparse optimization principles of *Sparse Relaxed Regularized Regression (SR3)* (Zheng et al., 2018). SR4-Fit is designed to produce compact and stable rule sets while maintaining competitive predictive accuracy. Its formulation allows fine-grained control over sparsity, offering a single framework that works for both classification (binary and multi-class) and regression tasks.

## 1.2 Research Objective

The goal of this research is to develop and evaluate an intrinsically interpretable rule-based learning algorithm that balances predictive performance with model transparency. The proposed SR4-Fit algorithm is designed to generate compact and stable rule sets while maintaining competitive predictive accuracy.

A central motivation of this research is to address the widely discussed *interpretability–predictivity trade-off* in the area of interpretable machine learning, where highly accurate models are often difficult to interpret, while interpretable models may sacrifice predictive strength. By integrating rule-based modeling with sparse optimization techniques, this work aims to demonstrate that interpretable models can achieve strong predictive performance without relying on black-box architectures.

Specifically, this research aims to achieve the following objectives:

1. To develop the SR4-Fit algorithm by integrating rule-based modeling with sparse optimization techniques derived from Sparse Relaxed Regularized Regression (SR3).

2. To investigate the ability of SR4-Fit to produce interpretable rule sets that capture meaningful relationships between input variables and predicted outcomes.

3. To evaluate the predictive performance and interpretability of SR4-Fit across multiple machine learning tasks, including binary classification, multiclass classification, and regression.

4. To apply the proposed algorithm to real-world electoral data in order to

analyze demographic patterns that influence U.S. House election outcomes.

5. To compare the performance of SR4-Fit with both black-box models and existing interpretable methods using publicly available benchmark datasets.

Through these objectives, this research seeks to demonstrate that it is possible to develop machine learning models that provide both strong predictive capabilities and transparent explanations.

## 1.3 Overview of Results

The experimental evaluation demonstrates that SR4-Fit achieves a strong balance between predictive performance and interpretability. Across multiple datasets, the proposed algorithm generates compact rule sets that are easier to interpret while maintaining predictive accuracy comparable to more complex machine learning models.

In the application to U.S. House election data, SR4-Fit identifies meaningful demographic relationships that influence partisan outcomes across congressional districts. The rules generated by the model highlight interactions between demographic variables such as education levels, racial composition, and age distribution. These interpretable patterns provide insights into how demographic structures shape electoral behavior and voting outcomes.

The regression experiments further demonstrate the ability of SR4-Fit to model continuous electoral outcomes, such as district-level Democratic and Republican vote percentages. By analyzing these continuous outcomes, the model captures how demographic gradients correspond to shifts in partisan support across districts, providing a more detailed understanding of electoral dynamics.

To evaluate the general applicability of the proposed method, SR4-Fit is also tested on a collection of publicly available benchmark datasets, consisting of six classification datasets and eight regression datasets. Across these datasets, the algorithm consistently produces stable rule sets with fewer rules and lower rule complexity while maintaining competitive predictive performance.

In addition to predictive performance, the results highlight improvements in interpretability-related characteristics such as rule stability, sparsity, and compactness of the generated rule sets. These results suggest that the proposed SR4-Fit framework successfully bridges the gap between interpretability and predictivity, demonstrating that intrinsically interpretable models can achieve strong empirical performance across different machine learning tasks.

## 1.4 Contributions to the Research Community

This dissertation makes several contributions to the field of interpretable machine learning. The primary contribution is the development of the SR4-Fit algorithm, an intrinsically interpretable rule-based learning framework designed to produce compact and stable rule sets while maintaining competitive predictive performance.

First, this work proposes SR4-Fit, a novel extension of the RuleFit framework that integrates sparse optimization principles from Sparse Relaxed Regularized Regression (SR3). By incorporating sparsity-driven optimization into the rule-learning process, the proposed approach generates more stable and compact rule sets while preserving predictive strength. This formulation allows the model to control rule complexity and sparsity in a principled manner.

Second, this research demonstrates that intrinsically interpretable models can

achieve strong predictive performance without relying on black-box architectures. By producing transparent rule sets that directly encode the reasoning behind predictions, SR4-Fit enables researchers and practitioners to examine the relationships captured by the model and validate them using domain knowledge.

Third, the dissertation also presents an interpretable analysis of U.S. House election outcomes using demographic data from the American Community Survey. To better evaluate interpretability beyond accuracy alone, it extends the Interpretability Score (IPS) by adding rule complexity to the existing components of predictive performance, stability, and rule count. This addition captures an important aspect that, even if two models have the same number of rules, simpler rules are much easier to understand than complex, multi-condition ones. By accounting for this, the enhanced IPS provides a more realistic measure of a model's interpretability. Using this framework, the resulting rule sets uncover meaningful relationships between demographic factors, such as race, education, age, and income, and voting patterns, offering clear, interpretable insights that black-box models fail to provide.

Finally, the proposed framework is evaluated across a diverse collection of publicly available benchmark datasets consisting of both classification and regression tasks. These experiments demonstrate that SR4-Fit consistently produces stable and compact rule sets while maintaining competitive predictive accuracy, highlighting its applicability across different machine learning problems.

Together, these contributions advance the development of interpretable machine learning methods and demonstrate that it is possible to bridge the gap between interpretability and predictive performance in rule-based learning systems.

## 1.5 Organization of the Dissertation

The remainder of this dissertation is organized into five chapters. Chapter 2 presents the background concepts and foundational ideas required to understand the proposed algorithm. Chapter 3 describes the methodology, datasets, and experimental setup used in this research. Chapter 4 presents the experimental results and observations obtained from applying the proposed method. Chapter 5 provides an analysis and interpretation of the rule sets produced by SR4-Fit. Finally, Chapter 6 summarizes the conclusions of this work and outlines potential directions for future research.

# Chapter 2

# Related Work

This chapter provides the foundational concepts and methodologies that underpin the development of the proposed SR4-Fit algorithm. It begins with classical tree-based methods, including Classification and Regression Trees (CART) and Random Forests, which form the basis for rule extraction and modeling nonlinear relationships in data. The chapter then introduces sparse linear modeling techniques such as LASSO and Sparse Relaxed Regularized Regression (SR3), which are essential for achieving model sparsity and stability. Building on these ideas, the RuleFit algorithm is discussed as a hybrid approach that combines tree-based rule generation with sparse linear modeling. Finally, the chapter presents key concepts related to interpretability, including stability measurement using the Dice–Sorensen index and the formulation of an interpretability score, which collectively establish the framework for evaluating interpretable machine learning models. These concepts provide the necessary theoretical and methodological foundation for the design and evaluation of SR4-Fit.

## 2.1 Classification and Regression Tree (CART)

The tree-based methods partition the feature space into a set of rectangles and fit them to a simple model (mostly it will be a constant) in each partitioned rectangle space (Hastie et al., 2009). Though they are conceptually simple, they can be powerful because of their ability to partition the feature space, leading to more accurate predictions. One such method is the Classification and Regression Tree (CART) (Breiman et al., 1984).

Let us consider data that has inputs and labels, which in this case, since it is classification data, the labels will be 1, 2, 3, ..., $K$ where $K$ is the number of classes for $N$ observations, which is $(x_i, y_i)$ where the $x$ are features and $y$ are labels and $i$ range from 1, 2, ... $N$. The algorithm automatically decides the splitting variables and splitting points. Suppose there is node $m$ which represents the region $R_m$ for $N_m$ observations, then

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k)$$

where $\hat{p}_{mk}$ is the proportion of class $k$ for observations in node $m$. $I$ is the indicator function which counts the number of observations in the node $R_m$ that belong to class $k$. The reason for using the equation is that it helps in identifying the proportion of a class or label, or how much a class impacts a node in a particular region. This value helps in identifying how impure the node is, which is calculated via the Gini index. The equation for *Gini index* is

$$Q_m(T) = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

where $Q_m(T)$ is the Gini index for node $m$. The reason for using the Gini index

is because it helps in understanding if the node is impure or not. That is, if all classes or labels have the same frequency, then the node is impure, or if only one class is present for the node, then it is maximally pure. It attains a value of zero when all observations in the node belong to a single class (maximal purity) and reaches its maximum when all classes are equally represented (maximal impurity). Intermediate values correspond to varying degrees of class heterogeneity $0 \leq Q_m \leq 1 - \frac{1}{K}$, with lower values indicating greater homogeneity within the node. Because of this, the best splitting points make the regions or the subsets look very different with respect to the target. The algorithm continues this search and splits recursively, creating many new nodes until a stop criterion is reached. The possible criteria that can be considered include a minimum number of data points that have to be in a node before the split or the minimum number of data points that have to be in a leaf node or the terminal node.

## 2.2 Random Forests

Random Forests (Breiman, 2001), is an ensemble learning method designed to improve predictive performance and robustness by combining multiple decision trees. The central idea is to construct a large collection of decision trees during training and aggregate their predictions to produce a final output. For classification tasks, the model outputs the class that receives the majority vote across all trees, while for regression tasks, it computes the average of the individual tree predictions.

A key characteristic of random forests is the use of bootstrap aggregation (bagging). Each tree in the ensemble is trained on a randomly sampled subset of the data drawn with replacement from the original dataset. This sampling process

ensures diversity among the trees, reducing variance and improving generalization performance. In addition to sampling data points, Random Forests introduce further randomness by selecting a random subset of features at each split when constructing individual trees. This random feature selection helps to decorrelate the trees, making the ensemble more robust and less prone to overfitting compared to a single decision tree.

Another important property of random forests is their ability to handle high-dimensional data and complex nonlinear relationships. Since each tree partitions the feature space differently, the ensemble collectively captures intricate patterns that may not be identifiable using simpler models. Furthermore, Random Forests provide an internal mechanism for estimating generalization error through out-of-bag (OOB) samples, which are data points not included in the bootstrap sample for a given tree. This allows for an unbiased estimate of model performance without requiring a separate validation set.

Despite their strong predictive performance, Random Forests are often considered black-box models due to the difficulty in interpreting the combined decision-making process of many trees. While individual trees are interpretable, the aggregation of numerous trees obscures the global structure of the model. This limitation motivates the development of intrinsically interpretable methods, such as rule-based models, which aim to retain predictive strength while providing transparent and human-understandable explanations.

## 2.3 Least Absolute Shrinkage and Selection Operator (LASSO)

The LASSO (Least Absolute Shrinkage and Selection Operator) (Tibshirani, 1996), is a regression technique that performs both coefficient shrinkage and variable selection within a unified framework. Unlike traditional ordinary least squares (OLS), which may produce unstable estimates in the presence of high-dimensional or correlated predictors, LASSO introduces an $\ell_1$-norm constraint that encourages sparsity in the model coefficients. This results in some coefficients being exactly zero, thereby selecting a subset of relevant predictors and improving model interpretability.

Formally, given a dataset $(x_i, y_i)$ for $i = 1, 2, \ldots, n$, the LASSO estimator is defined as the solution to the following optimization problem

$$(\hat{\beta}) = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left( y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} + \lambda \sum_{j=1}^{p} |\beta_j| \qquad \sum_{j=1}^{p} |\beta_j| \leq t$$

where $t \geq 0$ is a tuning parameter that controls the amount of shrinkage applied to the coefficients. $\alpha$ is the intercept term capturing the baseline response, whereas $\beta$ are the regression coefficients associated with the input features. Smaller values of $t$ enforce stronger regularization, shrinking more coefficients toward zero and increasing sparsity in the model.

The key advantage of LASSO lies in its ability to produce sparse models that retain predictive performance while enhancing interpretability. By shrinking some coefficients exactly to zero, LASSO effectively performs feature selection,

making it particularly useful in high-dimensional settings where interpretability and stability are important considerations.

## 2.4 Sparse Relaxed Regularized Regression (SR3)

Sparse Relaxed Regularized Regression (SR3) (Zheng et al., 2018), is a flexible optimization framework designed to improve sparse model estimation by decoupling the data-fitting term from the regularization term. Unlike traditional formulations such as LASSO, SR3 introduces an auxiliary variable to relax the coupling between sparsity and prediction.

The SR3 optimization problem is defined as:

$$\arg\min_{\beta,w} \frac{1}{2}\|y - X\beta\|_2^2 + \lambda R(w) + \frac{\kappa}{2}\|\beta - w\|_2^2$$

where $\beta$ represents the predictive coefficients, $w$ is an auxiliary variable enforcing sparsity, $R(w)$ is a regularization function (typically $\ell_1$-norm), $\lambda$ controls sparsity, and $\kappa$ controls the coupling between $\beta$ and $w$. This relaxation improves optimization behavior by allowing the sparsity constraint to act on $w$ rather than directly on $\beta$, resulting in improved convergence and more stable solutions.

The fundamental difference between LASSO and SR3 lies in how sparsity is enforced within the optimization framework. In LASSO, the sparsity constraint is applied directly to the model coefficients, which tightly couples the objectives of data fitting and regularization. This direct coupling can make the optimization problem more difficult to solve, particularly in high-dimensional or complex settings. In contrast, SR3 introduces an auxiliary variable to separate these two objectives. The predictive coefficients are optimized for data fitting, while spar-

sity is enforced on a separate variable, allowing greater flexibility in balancing accuracy and interpretability.

Compared to LASSO, the SR3 framework provides several key advantages arising from its relaxed formulation. The primary distinction lies in the introduction of an auxiliary variable, which allows the regularization term to act on a separate variable rather than directly on the model coefficients. This relaxation leads to improved optimization behavior and greater flexibility in sparse model construction. SR3 improves the conditioning of the optimization problem. The relaxation introduced through the quadratic penalty term $|\beta - w|_2^2$ results in a better-conditioned problem compared to the standard LASSO formulation. This improved conditioning leads to faster convergence and more stable numerical performance during optimization. Also, the SR3 formulation allows for more stable solutions across different datasets and sampling variations, as the relaxation reduces sensitivity to small perturbations in the data. This is particularly beneficial in applications where consistency of selected features or rules is important for interpretability.

## 2.5 RuleFit

The RuleFit algorithm (Friedman and Popescu, 2008) is composed of two components. The first component is the generation of rules by growing an ensemble of trees, which can be random forests or boosting techniques. The intuition behind growing trees is that each node in the tree can contribute to an elementary rule. The second component of the algorithm is to fit a sparse linear model, which is LASSO, with the input features and the new rules generated from the previous step. The intuition behind this step is to aggregate the rules into a smaller rule

set because the previous step produces lots of rules. The inclusion of raw features alongside extracted rules allows the model to capture both simple linear relationships and complex interaction effects, resulting in a more expressive and compact representation (Molnar, 2020).

The first component is the rule generation component, which is generated by decomposing the ensemble of trees. A path in a tree to a particular node can be converted into a decision rule. The trees used for generating the rules are used to predict the target outcome. Mostly ensemble techniques like gradient boosting will be used for generating the trees. The general equation of a tree ensemble can be given by

$$\hat{f}(x) = a_0 + \sum_{m=1}^{M} a_m \hat{f}_m(x)$$

where M is the number of trees and $\hat{f}_m(x)$ is the prediction function of the $m^{th}$ tree. The $a$ are the weights or the scaling factor. The intuition behind the scaling factor is that the scaling factor acts as the ratio of the measurement of the tree $m$ and the actual measurement. Bagged ensembles, random forest, and AdaBoost are the different ensemble techniques that can produce tree ensembles and can be used for RuleFit. The idea behind the above equation is that this equation grows the trees, and from this the rules can be extracted, and this rule extraction is done by the equation below.

$$r_m = \prod_{j \in T_m} I(x_j \in s_{jm})$$

where $r_m$ is the rules for the $m^{th}$ tree, $T_m$ is the set of observations used in the $m^{th}$ tree, $I$ is the indicator function that is 1 when the feature $x_j$ is in the specified

subset of values $s$ for the $j^{th}$ feature or it is also specified as the tree splits and 0 otherwise. This means that if the specific feature is within the condition of the tree splits it will be considered for making a rule or not. The RuleFit learns trees with random depth so that many diverse rules with different lengths are generated.

In the first step, there will be many rules because the rules are generated from the ensemble, and the ensemble has lots of nodes, resulting in many rules. So in order to limit the number of rules, it is fitted to a sparse linear model. In this step, in addition to the rules from the first step, all the raw features from the original dataset are used as the input for fitting in the sparse linear model. The equation, which combines both rules and features in the sparse linear model, is given as

$$\hat{f}(x) = \hat{\beta}_0 + \sum_{k=1}^{K} \hat{\alpha}_k r_k(x) + \sum_{j=1}^{P} \hat{\beta}_j f_j(x_j)$$

where $\hat{\alpha}$ is the estimated weight vector for the rule features and $\hat{\beta}$ the weight vector for the original features. Both weights are updated by minimizing the loss of the model. The result is a linear model that has linear effects for all of the original features and for the rules. The interpretation is the same as for linear models, the only difference is that some features are rules.

## 2.6 Dice-Sorenson Index

The Dice–Sorensen index (Chao et al., 2006) is used to quantify the stability of the rule sets generated by the model. Stability, in this context, refers to the consistency of the rules produced by the algorithm when trained on different

samples of the same dataset (Bénard et al., 2019). Let $\hat{P}_{M,n,p_0}$ and $\hat{P}'_{M,n,p_0}$ denote the sets of rules obtained from two independently trained models on different samples. The Dice–Sorensen index measures the proportion of rules that are common between these two sets and is defined as

$$\hat{S}_{M,n,p_0} = \frac{2|\hat{P}_{M,n,p_0} \cap \hat{P}'_{M,n,p_0}|}{|\hat{P}_{M,n,p_0}| + |\hat{P}'_{M,n,p_0}|}.$$

This metric takes values in the range $[0, 1]$, where a value of 1 indicates that both models produce identical rule sets, reflecting high stability, while a value of 0 indicates no overlap between the rule sets, reflecting low stability. In this dissertation, independent samples are generated using cross-validation, and the stability is computed as the average Dice–Sorensen index across different folds. This provides a robust measure of how consistently the algorithm selects rules under variations in the training data.

## 2.7 Interpretability Score

In this dissertation, interpretability is understood as a combination of three fundamental components: predictivity, stability, and simplicity (Margot and Luta, 2021). Rather than treating interpretability as a vague or purely qualitative notion, this work adopts a quantitative perspective in which these three components collectively define how understandable and reliable a model is. Predictivity reflects how well the model performs in terms of accuracy or error, ensuring that interpretability does not come at the cost of poor predictive performance. Stability captures the consistency of the rules generated by the model across different samples of the data, which is measured using the Dice–Sorensen index. This en-

sures that the model does not produce highly variable or unreliable explanations. Simplicity, on the other hand, represents the structural complexity of the model and is quantified based on the total length of the rule set, where shorter and fewer rules correspond to more interpretable models.

Based on this formulation, the interpretability score is defined as a weighted combination of these three components:

$$\text{IPS}(A) = \gamma_P P_n(A) + \gamma_S S_n(A) + \gamma_{simp}(1 - R_n^{\text{simp}}(A))$$

where $P_n(A)$ denotes the predictivity score, $S_n(A)$ represents the stability score, and $R_n^{\text{simp}}(A)$ corresponds to the simplicity score of the model. $R_n^{\text{simp}}(A)$ is inverted to account for lower values, which would contribute to higher interpretability. The subscript $n$ indicates that each component is computed based on $n$ samples or observations used in the evaluation process, typically through resampling techniques such as cross-validation.

The weights $\gamma_P$, $\gamma_S$, and $\gamma_{simp}$ allow flexibility in emphasizing different aspects of interpretability, although equal weighting is commonly adopted for balanced evaluation. In this dissertation, this formulation is used as a guiding framework to evaluate and compare interpretable models, particularly to analyze how SR4-Fit balances predictive performance with stability and structural simplicity. In addition to these components, rule complexity is also considered an important factor in evaluating interpretability. While the simplicity score captures the overall size of the rule set, it does not fully account for the internal structure of individual rules. Rule complexity, defined as the number of conditions within a rule, provides a more granular measure of how easily a rule can be understood. Even a small number of rules can become difficult to interpret if each rule contains

many conditions. Therefore, incorporating rule complexity allows for a more comprehensive assessment of interpretability by ensuring that both the number of rules and their structural depth are taken into account. The equation with rule complexity can be seen in section 3.2.

# Chapter 3

# Methodology and Experimental setup

This chapter presents the proposed Sparse Relaxed Regularized Regression Rule Fit (SR4-Fit) algorithm along with the experimental framework used to evaluate its performance. The chapter details the formulation of the algorithm, its underlying design principles, and how it integrates rule-based learning with sparse optimization techniques. It also outlines the datasets, evaluation metrics, and comparative models used to assess the effectiveness of SR4-Fit in balancing interpretability and predictive accuracy across both classification and regression tasks.

## 3.1 SR4-Fit Methodology

The SR4-Fit algorithm is developed based on the interpretability approach of the RuleFit algorithm with the sparse optimization capabilities of SR3 (Murali Krishnan and Hougen, 2025). SR4-Fit extracts decision rules from decision trees to

capture non-linear patterns similar to RuleFit and then applies SR3 optimization to fit a sparse linear model over both rules and raw features. To adapt this setup for classification tasks, the SR3 objective is modified by replacing the squared loss with logistic loss, allowing the model to predict class probabilities rather than continuous outputs. For regression tasks, SR4-Fit retains the same optimization framework but employs a squared-error loss instead of the logistic loss. This enables the model to predict continuous target variables. This results in sparse, interpretable classification and regression models that leverage rule-based logic and SR3's structured regularization to balance accuracy in the case of classification and lower square loss in the case of regression with explainability.

### 3.1.1 Rule Extraction

The SR4-Fit algorithm starts by extracting interpretable decision rules from a collection of decision trees, which are trained on the input data. In a classification setting, for each class in the dataset, a binary one-versus-rest formulation (Rifkin and Klautau, 2004) is used, where the forest is trained to distinguish the class of interest from all others. For the regression setting, the trees are trained to predict the continuous target variable directly, enabling the model to capture meaningful nonlinear dependencies between features and the output variable. From every tree in the forest, rules are extracted from each path from the root to internal or terminal nodes. Each path defines a rule composed of logical conditions on the input features. These rules are encoded as binary indicators, returning 1 if the rule evaluates to true for a particular input vector and 0 otherwise. For example, the rule

$$X_1 \leq 2.7 \text{ and } X_3 > 1.4$$

asserts that input feature $X_1$ is less than 2.7 and input feature $X_3$ is greater than 1.4. If this is true, the rule evaluates to 1; otherwise, it evaluates to 0. Because rules are collected from each tree throughout, the resulting set captures diverse and robust patterns throughout the input space. To ensure interpretability, the number of rules retained is limited by the maximum rules $r_{\max}$ parameter.

### 3.1.2 Feature Construction and Optimization Objective

Following rule extraction, SR4-Fit builds an extended feature matrix by combining the original input features with the binary rule indicators. For each data point, the rule evaluations are concatenated with the original feature values to form a comprehensive representation $Z = [X \mid R]$ where $X \in \mathbb{R}^{n \times d}$ and $R \in \{0, 1\}^{n \times m}$. This expanded feature matrix allows the model to utilize both raw features and data-driven patterns captured by the trees. To learn a sparse and accurate classifier, the algorithm minimizes the loss function, which consists of two parameters $\beta$, the model's predictive weight vector, and $w$, the vector enforcing sparsity:

$$L(\beta, w) = \sum_{i=1}^{n} \log\left(1 + \exp\left(-y_i Z_i^\top \beta\right)\right) + \lambda \|w\|_1 + \frac{\kappa}{2} \|\beta - w\|_2^2.$$

.

For regression tasks, the same structure is retained, but the logistic loss is replaced by the squared-error loss:

$$L(\beta, w) = \frac{1}{2} \|y - Z\beta\|_2^2 + \lambda \|w\|_1 + \frac{\kappa}{2} \|\beta - w\|_2^2.$$

.

In these objective functions, $\lambda$ controls the sparsity and $\kappa$ controls how closely $\beta$ must follow $w$. The algorithm minimizes these objective functions by alternating between the two update steps. First, with $w$ fixed, $\beta$ is updated using gradient-based optimization on the logistic loss for classification and a closed-form ridge-type solution on the squared error loss for regression, both balancing fitting accuracy with quadratic regularization that enforces $\beta \approx w$. Second, with $\beta$ fixed, $w$ is updated via soft-thresholding, which is key for achieving sparsity (Tibshirani, 1996), using

$$w_j = \text{sign}(\beta_j) \cdot \max\left(|\beta_j| - \frac{\lambda}{\kappa}, 0\right)$$

where

$$\text{sign}(z) = \begin{cases} +1, & \text{if } z > 0 \\ 0, & \text{if } z = 0 \\ -1, & \text{if } z < 0. \end{cases}$$

.

This iterative process continues until the relative change in the objective function falls below a small tolerance $\varepsilon$ (typically $10^{-4}$). The result is a compact, interpretable model in which the nonzero components of $w$ identify the features and rules that contribute most strongly to prediction accuracy.

### 3.1.3 Model Pruning and Rule Selection

After optimization, SR4-Fit performs model pruning to retain only the most informative features and rules. The pruning step relies on the sparse auxiliary vector $w$, which reflects the importance of each feature and rule learned during training. Any component $w_j = 0$ indicates that the corresponding predictor

contributes negligibly to the model and can be safely removed. The remaining nonzero elements of $w$ identify the subset of features and rules that form the final interpretable model.

This separation of roles between $\beta$ and $w$, where $\beta$ captures predictive strength and $w$ enforces sparsity, provides a more stable and interpretable selection process than methods that apply sparsity constraints directly to the predictive coefficients. Because $w$ is updated through soft-thresholding, irrelevant or weakly correlated rules are automatically pruned, leading to a compact and transparent rule set.

The pruning mechanism is identical for both classification and regression. By discarding inactive predictors and retaining only those associated with nonzero $w_j$ values, SR4-Fit yields a sparse model that is both easy to interpret and efficient to evaluate. This final rule set highlights the dominant feature interactions uncovered during training, offering clear and human-readable insights into how the model forms its predictions.

### 3.1.4 Classification and Regression Prediction

In the final stage, SR4-Fit uses the optimized coefficients $\beta$ to make predictions for both classification and regression tasks. For a new input instance $x$, the model evaluates the active rules and features and computes a linear score

$$s(x) = \beta_0 + \sum_{j=1}^{d+m} \beta_j Z_j(x),$$

where $Z_j(x)$ represents the $j^{th}$ component of the extended feature vector that combines both raw features and rule indicators. The value $s(x)$ serves as the aggregated evidence derived from all selected predictors. For classification tasks,

SR4-Fit transforms this score into a probability through the sigmoid function

$$\hat{y}(x) = \sigma(s(x)) = \frac{1}{1 + \exp(-s(x))},$$

mapping the real-valued score to the interval $(0, 1)$ and providing a probabilistic interpretation of the model's confidence in the positive class. This probabilistic formulation not only supports threshold-based classification but also conveys uncertainty in predictions. For multiclass problems, SR4-Fit employs a one-vs-rest approach, training one binary classifier per class; the final class prediction corresponds to the one with the highest estimated probability. Because each classifier is constructed from sparse and interpretable rules, the decision path for every class prediction can be directly traced through the contributing rules and coefficients.

For regression tasks, SR4-Fit uses the same linear prediction structure without the sigmoid transformation, which provides a direct estimate of the target variable. Each nonzero coefficient $\beta_j$ quantifies the contribution of the corresponding rule or feature to the predicted outcome, allowing clear interpretation of how specific variables influence the continuous target. Overall, SR4-Fit delivers a unified prediction framework that supports both probabilistic classification and continuous regression while preserving interpretability through its sparse, rule-based structure.

## 3.2 Experimental Setup

The demographic data, which was used in a previous study covering U.S. House of Representatives elections from 2006 to 2013 for training and 2014 to 2016

**Algorithm** SR4-Fit Classification Algorithm

**Input:** Dataset $(X, y)$, parameters $\lambda$, $\kappa$ and $r_{\max}$.
**Output:** Predictive weights $\beta$.
**for** each class $c = 0, 1, \ldots, C$ **do**
    Train a random forest using one-vs-rest labeling for class $c$ (Breiman, 2001).
    Extract decision rules from each path in each tree until $r_{\max}$ (Friedman, 2001).
    Encode rules as binary indicators $R \in \{0, 1\}^{n \times m}$ (Cohen et al., 1995).
    Form an extended feature matrix $Z = [X \mid R]$.
    Initialize $\beta$, $w$.
    **while** not converged **do**
        Minimize $L(\beta, w) =$
        $$\sum_{i=1}^{n} \log(1 + \exp(-y_i Z_i^\top \beta)) + \lambda \|w\|_1 + \frac{\kappa}{2} \|\beta - w\|_2^2$$
        Update $w$, $w_j = \text{sign}(\beta_j) \cdot \max\left(|\beta_j| - \frac{\lambda}{\kappa}, 0\right)$
    **end while**
    Prune rules where $w_j = 0$
**end for**
**Prediction:**
Compute $s(x) = \beta_0 + \sum_{j=1}^{d+m} \beta_j Z_j(x)$
Compute predicted probability $\hat{y}(x) = \frac{1}{1+\exp(-s(x))}$

**Algorithm** SR4-Fit Regression Algorithm

**Input:** Dataset $(X, y)$, parameters $\lambda$, $\kappa$, maximum rules $r_{\max}$, and tolerance $\varepsilon$.
**Output:** Regression coefficients $\beta$.
Train a random forest of depth $d$ on $(X, y)$.
Extract decision rules from all root-to-leaf paths up to $r_{\max}$.
Encode each rule as a binary feature; form an extended matrix $Z = [X \mid R]$.
Initialize coefficients $\beta$, $w$.
**while** not converged **do**
    Update $\beta$ by solving

$$\beta = \arg\min_{\beta} \frac{1}{2}\|y - Z\beta\|_2^2 + \frac{\kappa}{2}\|\beta - w\|_2^2$$

    Update $w$ using soft-thresholding:

$$w_j = \text{sign}(\beta_j) \cdot \max(|\beta_j| - \tfrac{\lambda}{\kappa}, 0)$$

    Check convergence: $\frac{|L_{t-1} - L_t|}{L_{t-1}} < \varepsilon$, where

$$L(\beta, w) = \tfrac{1}{2}\|y - Z\beta\|_2^2 + \lambda\|w\|_1 + \tfrac{\kappa}{2}\|\beta - w\|_2^2$$

**end while**
prune rules with $|w_j| = 0$.
**Prediction:** For a new instance $x$, compute $\hat{y}(x) = \beta_0 + \sum_{j=1}^{d+m} \beta_j Z_j(x)$.

for testing, (U.S. Census Bureau, 2020) has been used for these experiments. The election results were obtained from the MIT Election Data and Science Lab's "U.S. House 1976–2018" dataset (MIT Election Data and Science Lab, 2017), which provides district-level outcomes. Of the 2,610 seats contested during this period, seven (0.2%) were won by candidates who were neither Democrats nor Republicans. For the experiments, four datasets were used based on the demographic characteristics included. The *minimum dataset* consists of median age, male percentage, white percentage, and bachelor's degree attainment. The *standard dataset* adds breakdowns of age, race, income, and education categories to the minimum set. The *expanded dataset* further includes language spoken at home, marital status, and poverty indicators. The *previous dataset* supplements the standard attributes with information about the party that won the district in the prior election.

For the classification tasks, experiments were conducted under two configurations: (i) including both Democratic (DEM) and Republican (REP) vote percentages as features, and (ii) excluding both party-percentage variables to analyze model behavior under purely demographic inputs. For the regression tasks, two prediction settings were considered: predicting DEM vote percentage and predicting REP vote percentage. Each regression setting was further evaluated under two feature conditions: when DEM percentage served as the label, experiments were conducted both with REP percentage included as a feature and with it removed; similarly, when REP percentage served as the label, experiments were run with and without DEM percentage as a feature.

These multi-condition experiments were designed to systematically assess model robustness, predictivity, and interpretability under varying degrees of political information. Party-percentage variables are highly predictive and can dom-

inate model behavior; removing them allows us to evaluate how well the models rely on demographic structure alone. By analyzing this case, we obtain a more complete understanding of SR4-Fit's performance along with its competitors, its sensitivity to feature availability, its capacity to generalize under reduced information, and its ability to produce stable and interpretable rules across both politically rich and politically limited scenarios.

Previous research on U.S. demographic data had shown that Random Forest and Support Vector Machine (SVM) models are strong predictors. Along with those models, XGBoost is also used for experimental comparison with SR4-Fit to evaluate performance on demographic data, enabling a direct comparison between black-box models and the interpretable SR4-Fit rule-based model. Additionally, SR4-Fit was compared with the traditional RuleFit algorithm, as SR4-Fit was designed with a similar interpretability objective, ensuring a fair evaluation setting. Along with RuleFit, other interpretable and compact models like decision trees and linear regression were also included to provide an extensive comparison with respect to interpretable and compact models. The hyperparameters ($r_{\max}$, $\lambda$ , $\kappa$) were selected via grid search (Pedregosa et al., 2011) across a range of values. Each setting was evaluated over multiple trials, and the combination yielding higher accuracy was chosen.

All experiments on the U.S. demographic datasets were conducted over 30 independent trials for each model to provide statistically meaningful results. For classification tasks, the predictive performance was assessed using standard classification metrics, including accuracy, precision, recall (Swets, 1969), and F1-score (Rijsbergen, 1979), to provide a quantitative evaluation. For regression tasks, the predictive performance was assessed using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) (Hodson, 2022), and Coefficient of Determination

($R^2$). The Dice–Sorensen index (Chao et al., 2006) was also employed to analyze the robustness of the models.

To quantify the interpretability of the model, the *Interpretability Score* (IPS) is calculated based on the equally weighted combination of the accuracy of the model for classification and RMSE for regression; the consistency or robustness of the model (Dice-Sorensen); the simplicity of the model (number of rules); and the rule complexity (number of components in a rule) (Margot and Luta, 2021). The interpretability is compared only between rule-based models like RuleFit, SR4-Fit, random forest, and decision trees.

$$\text{IPS}(A) = \gamma_P P_n(A) + \gamma_S S_n(A) + \gamma_{simp}(1 - R_n^{\text{simp}}(A)) + \gamma_{comp}(1 - R_n^{\text{comp}}(A))$$

where $P_n(A)$ denotes the predictivity score, $S_n(A)$ represents the stability score, $R_n^{\text{simp}}(A)$ corresponds to the simplicity score of the model, and $R_n^{\text{comp}}(A)$ corresponds to the rule complexity of the model. The weights $\gamma_P$, $\gamma_S$, $\gamma_{simp}$ and $\gamma_{comp}$ allow flexibility in emphasizing different aspects of interpretability. $R_n^{\text{simp}}(A)$ and $R_n^{\text{comp}}(A)$ are inverted to account for lower values, which would contribute to higher interpretability. The normalization of the components is done using MinMaxScaler in Python (Hao and Ho, 2019), which converts each component to a range of 0 to 1.

$$v_{\text{scaled}} = \frac{v - v_{\text{min}}}{v_{\text{max}} - v_{\text{min}}}$$

where $v$ is the original value of the component, $v_{\text{min}}$ is the minimum value of the component across trails, and $v_{\text{max}}$ is the maximum value of the component across

trails.

To facilitate visual comparative analysis, violin plots and line plots were used (Tukey, 1977). These visualizations help to illustrate the variability, central tendency, and distribution of each model's performance in aggregate and across trials, making it easier to assess both consistency and predictive strength (Suh et al., 2023).

To demonstrate the broad applicability of the SR4-Fit algorithm in classification settings, six publicly available datasets were also selected for experimentation. These include both binary and multiclass classification tasks, as well as imbalanced datasets, ensuring comprehensive evaluation. The Wisconsin Breast Cancer dataset (Wolberg, 1990) provides diagnostic features from digitized images of breast masses to classify tumors as benign or malignant. The E. coli dataset (Nakai, 1996) is comprised of protein localization sites across different cellular compartments. The Page Blocks dataset (Malerba, 1994) includes layout and geometric features of document blocks from scanned pages for classifying each block's type. The Pima Indians Diabetes dataset (Smith, J. W. and Everhart, J. E. and Dickson, W. C. and Knowler, W. C. and Johannes, R. S., 1988) contains medical and demographic attributes of female Pima Indian individuals to predict the onset of type 2 diabetes. The vehicle silhouette dataset (Mowforth and Shepherd) involves shape descriptors extracted from vehicle silhouettes to classify them into categories based on geometric properties. Lastly, the yeast dataset (Nakai, 1991) includes sequence-based features of yeast proteins to predict their cellular localization. In this set of datasets, except for those of breast cancer and Pima Indians diabetes, all other datasets are multi-class datasets.

For regression settings, eight distinct public datasets were used for experiments. The Ozone dataset predicts the daily average ozone measurement in Los

Angeles, 1976 (Friedman et al., 2001a). The MPG data predicts fuel consumption for city cycles (Quinlan, 1993a). The machine data predicts relative CPU performance (Quinlan, 1993b). The abalone data carries out age predictions of abalone (Nash et al., 1994). The diabetes data predicts how the disease will progress one year from the baseline (Efron et al., 2004). The bone mineral density of 261 adolescents in North America is predicted by the bones data (Friedman et al., 2001b). The housing data predicts Boston home prices (Asuncion and Newman, 2003). Prostate-specific antigen levels in males slated for radical prostatectomy are predicted by the prostate data (Friedman et al., 2001c).

# Chapter 4

# Results

Across all experimental settings, including U.S. House election classification, Democratic and Republican regression tasks, and a broad collection of public benchmark datasets, SR4-Fit is evaluated against linear regression, SVM, decision tree, XGBoost, random forest, and RuleFit to assess its predictive performance and compared with only rule-based methods (decision tree, random forest, and RuleFit) to assess interpretability through an interpretability score. These experiments span multiple data modalities and difficulty levels, ranging from politically structured demographic datasets to heterogeneous real-world regression and classification benchmarks, enabling a comprehensive assessment of how SR4-Fit behaves under varying feature conditions and domain characteristics. By examining performance across both full-information settings (with party-percentage features) and reduced-information settings (with key predictors removed), the results provide insight into the model's robustness, generalization, and capacity to generate compact and stable rule sets.

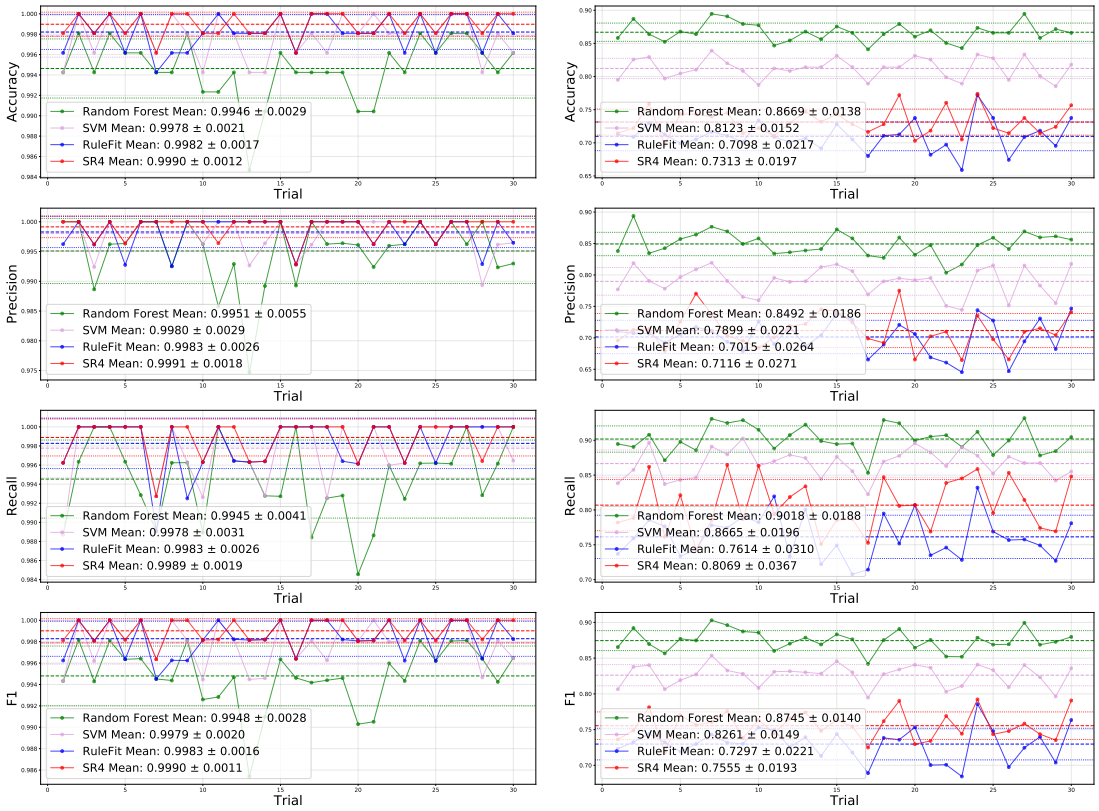



Figure 4.1: Line plot comparison of model performance metrics for minimum data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

## 4.1 Results of U.S House of Representative elections classification

Across all configurations of the U.S. House of Representatives election classification task, SR4-Fit consistently emerges as the combination of a stable, accurate, and compact model, even when compared with all the baselines. The combined evidence from line plots, violin plots, stability tables, and rule-complexity analyses provides a comprehensive view of model behavior both with and without the party-percentage features.

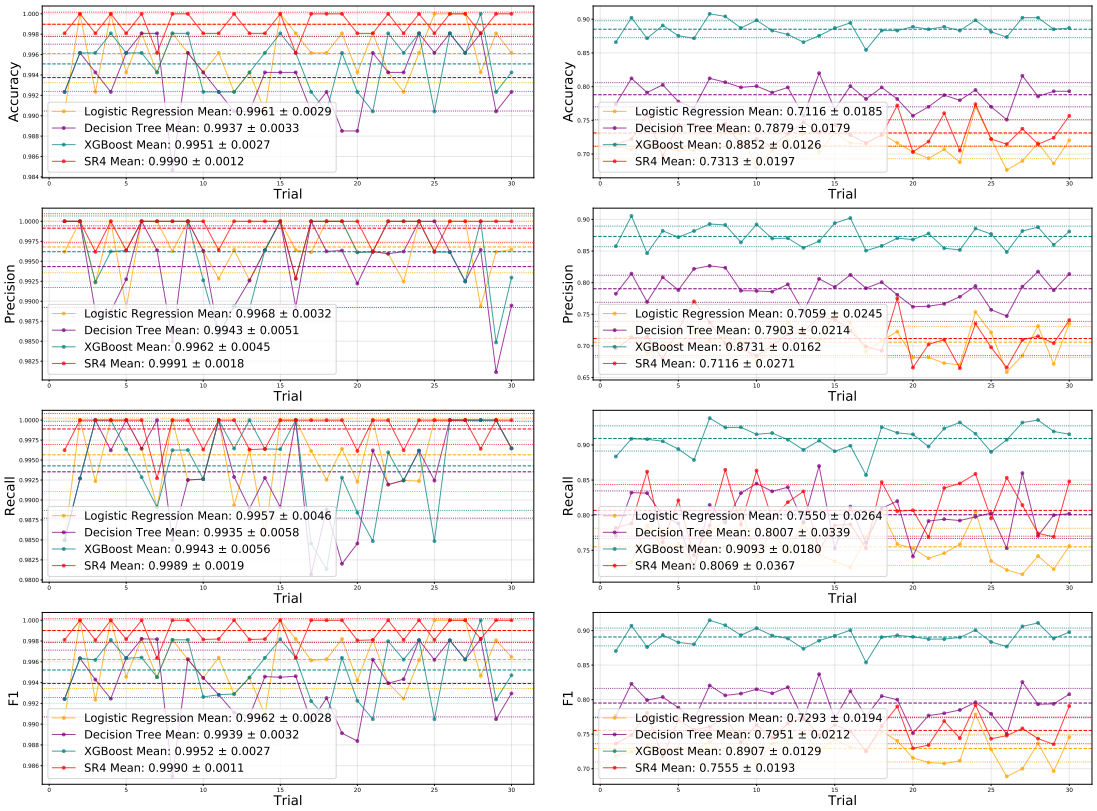



Figure 4.2: Line plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics for minimum data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

In the minimum dataset, the line plots in Figure 4.1 and Figure 4.2 show that when party percentage is included, all models achieve high performance; however, SR4-Fit exhibits the smoothest trajectories across accuracy, precision, recall, and F1 with small variance, outperforming logistic regression, decision tree, and XGBoost. The distributional comparisons through violin plots in Figure 4.9 and Figure 4.10 confirm that SR4-Fit produces tighter metric distributions than other models, which display broader spreads and heavier fluctuations. When party percentage is removed, performance declines across models, with XGBoost maintaining the highest predictive scores, while SR4-Fit maintains comparatively

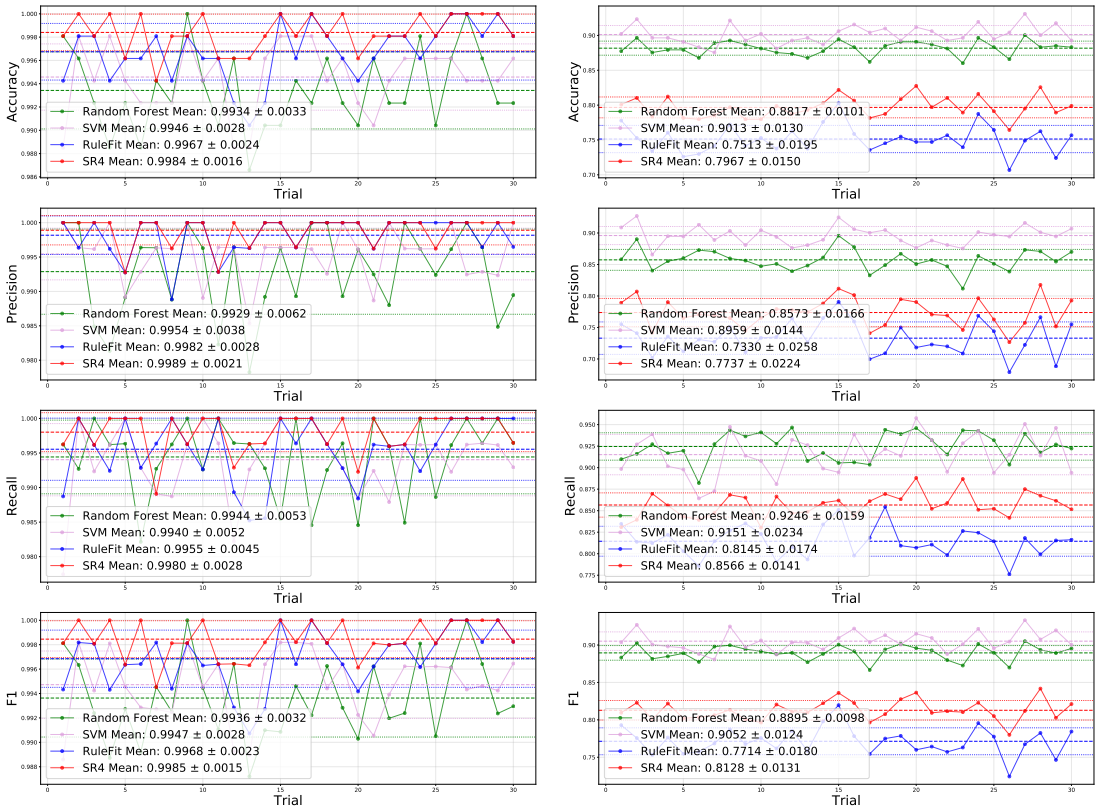



Figure 4.3: Line plot comparison of model performance metrics for standard data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

smoother trends despite lower accuracy, which is further confirmed through the violin plots in Figure 4.11 and Figure 4.12 where the distributions are tighter for SR4-Fit despite lower accuracy. Stability analysis in Table 4.1 shows that with and without party percentages, SR4-Fit achieves the highest Dice–Sorensen Index when compared with all rule-based models, showing its structural consistency and robustness. Table 4.2 shows that in the case of party percentages as a feature, the decision tree has the most compact structure, which is preceded by both SR4-Fit and RuleFit, whereas without including party percentages, SR4-Fit has a more compact structure with respect to the number of rules. Table 4.3 shows

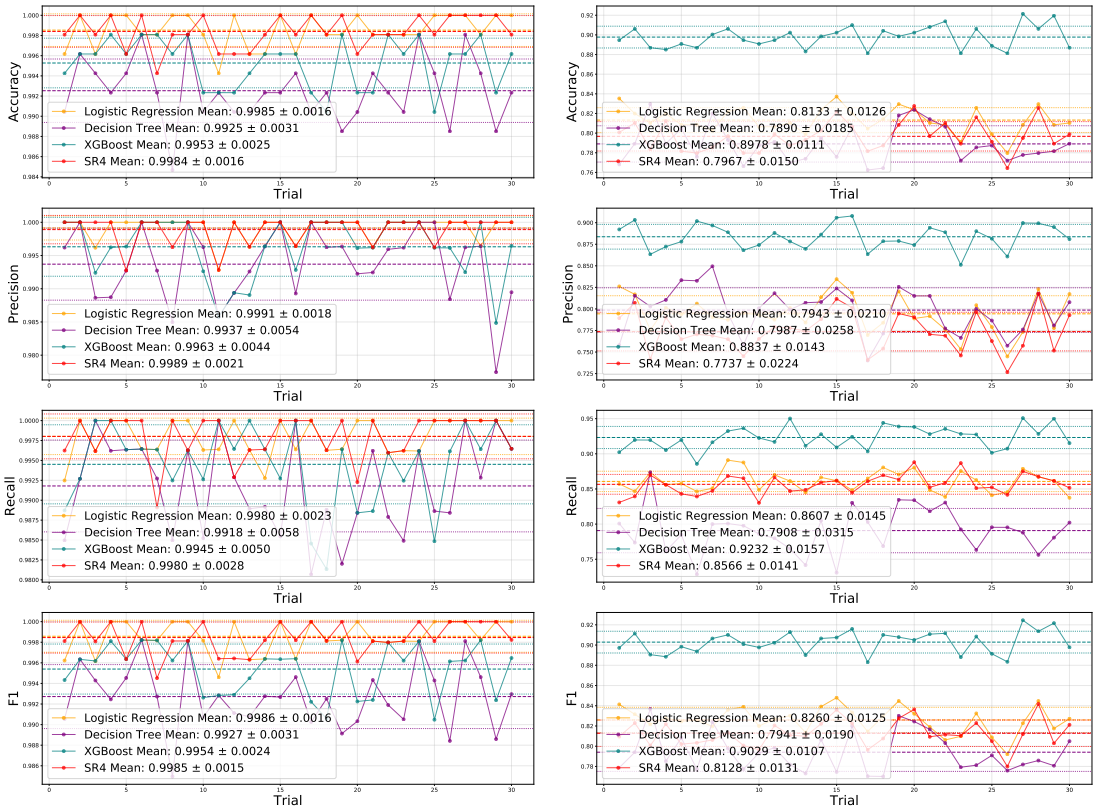



Figure 4.4: Line plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics for standard data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

that random forest has lower average rule complexity compared to all the other models.

In the standard dataset, the line plots in Figure 4.3 and Figure 4.4 show that when party percentage is included, SR4-Fit exhibits the smoothest trajectories with small variance, outperforming all models. The distributional comparisons through violin plots in Figure 4.9 and Figure 4.10 confirm that SR4-Fit produces tighter metric distributions than other models. When the party percentages are removed, performance declines across models, with SVM maintaining the highest predictive scores. Stability analysis in Table 4.1 shows that with and

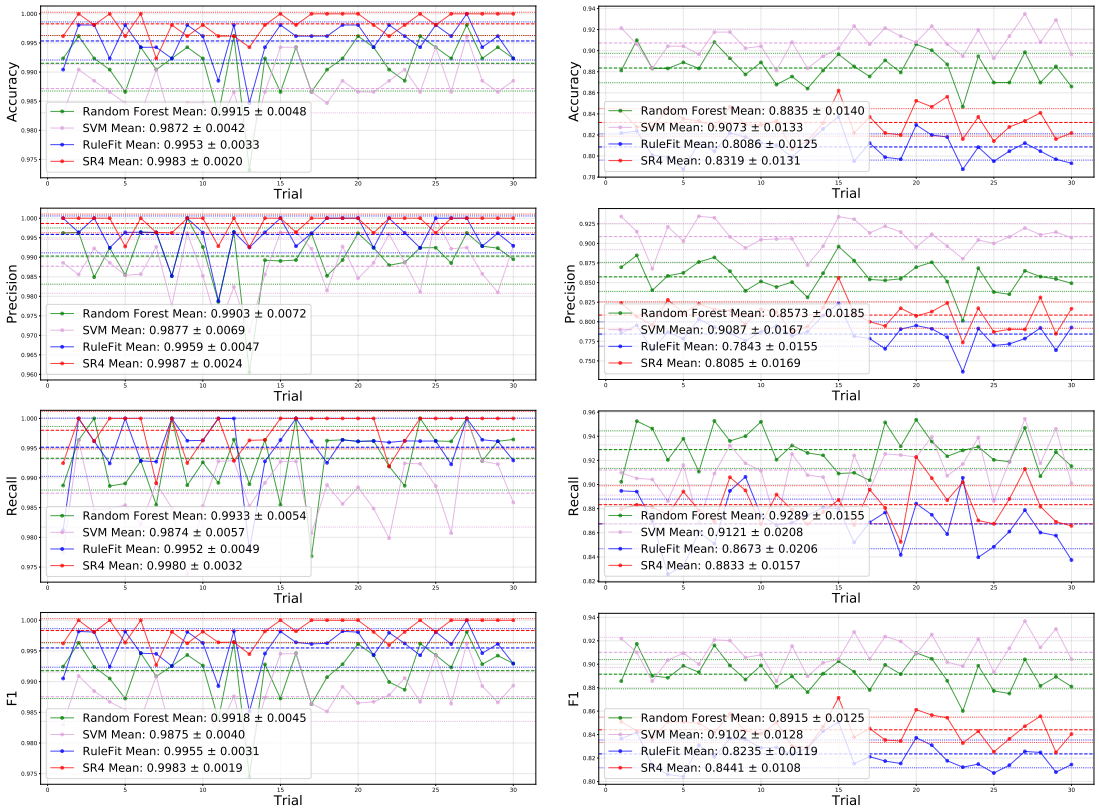



Figure 4.5: Line plot comparison of model performance metrics for expanded data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

without party percentages, RuleFit achieves the highest Dice–Sorensen Index when compared with all rule-based models, which is closely preceded by SR4-Fit. Table 4.2 shows that in the case of party percentages as a feature, the decision tree has the most compact structure, whereas, without including party percentages, Random Forest has a more compact structure with respect to the number of rules. Table 4.3 shows that Random Forest has lower average rule complexity when party percentages are not included, whereas RuleFit has lower average rule complexity when party percentages are included.

In the expanded dataset, the line plots in Figure 4.5 and Figure 4.6 show

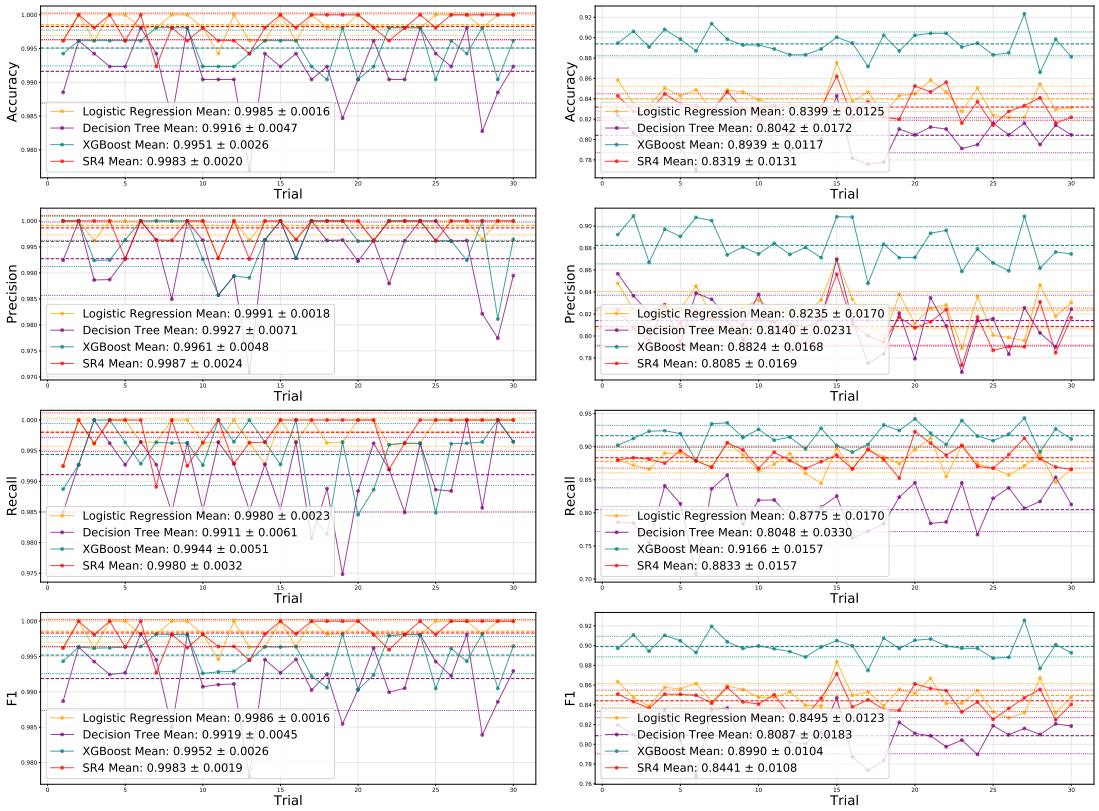



Figure 4.6: Line plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics for expanded data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

that when party percentage is included, SR4-Fit exhibits the smoothest trajectories with small variance, outperforming all models. When the party percentages are removed, performance declines across models, with SVM maintaining the highest predictive scores. Stability analysis in Table 4.1 shows that with and without party percentages, SR4-Fit achieves the highest Dice–Sorensen Index when compared with other rule-based models. Table 4.2 shows that in the case of party percentages as a feature, the decision tree has the most compact structure, whereas, without including party percentages, Random Forest has a more compact structure with respect to the number of rules. Table 4.3 shows that

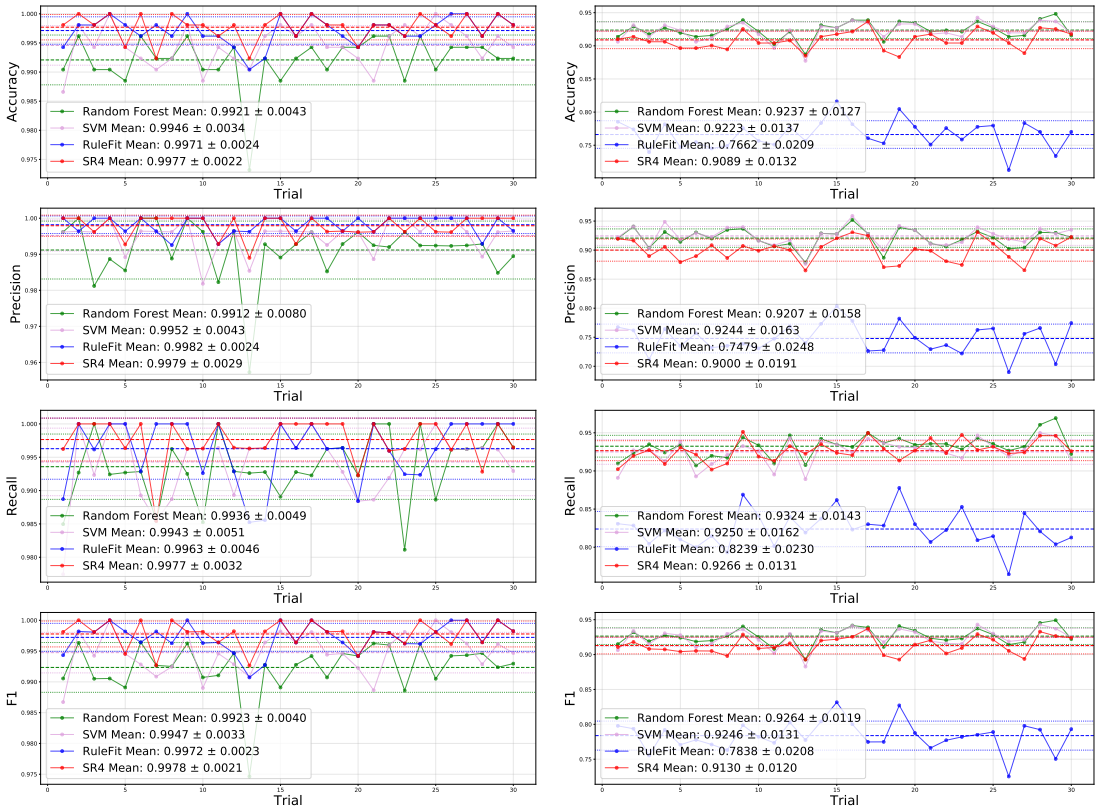



Figure 4.7: Line plot comparison of model performance metrics for previous data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

random forest has lower average rule complexity when party percentages are not included, whereas both SR4-Fit and RuleFit have lower average rule complexity when party percentages are included.

In the previous dataset, the line plots in Figure 4.7 and Figure 4.8 show that when party percentage is included, SR4-Fit exhibits the smoothest trajectories with small variance, outperforming all models. When the party percentages are removed, performance declines across models, with Random Forest maintaining the highest predictive scores. Stability analysis in Table 4.1 shows that with and without party percentages, SR4-Fit achieves the highest Dice–Sorensen Index

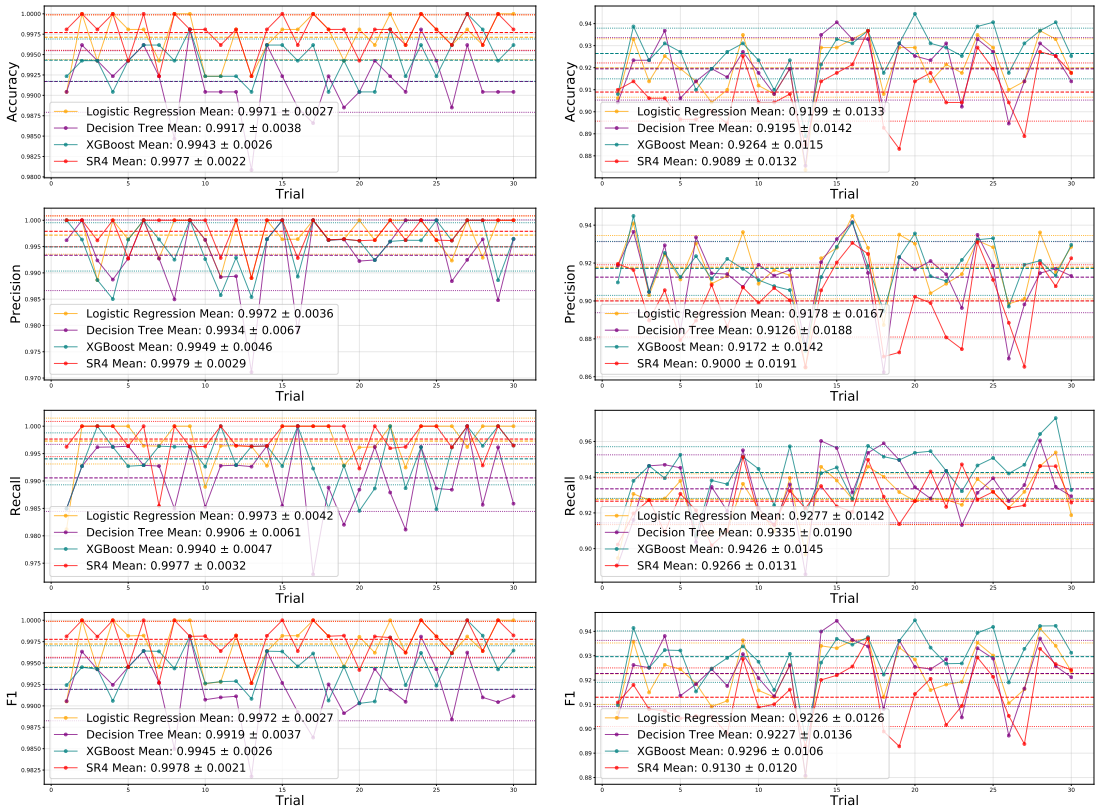



Figure 4.8: Line plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics for previous data across 30 trials. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

when compared with other rule-based models. Table 4.2 shows that in both cases, the decision tree has the most compact structure. Table 4.3 shows that random forest has lower average rule complexity when party percentages are not included, whereas SR4-Fit has lower average rule complexity when party percentages are included.

Across all four demographic datasets, SR4-Fit consistently provides stronger interpretability than both RuleFit and Random Forest in the classification setting. As shown in Table 4.4, SR4-Fit achieves the highest IPS values when party percentages are included, outperforming the alternatives in the minimum,

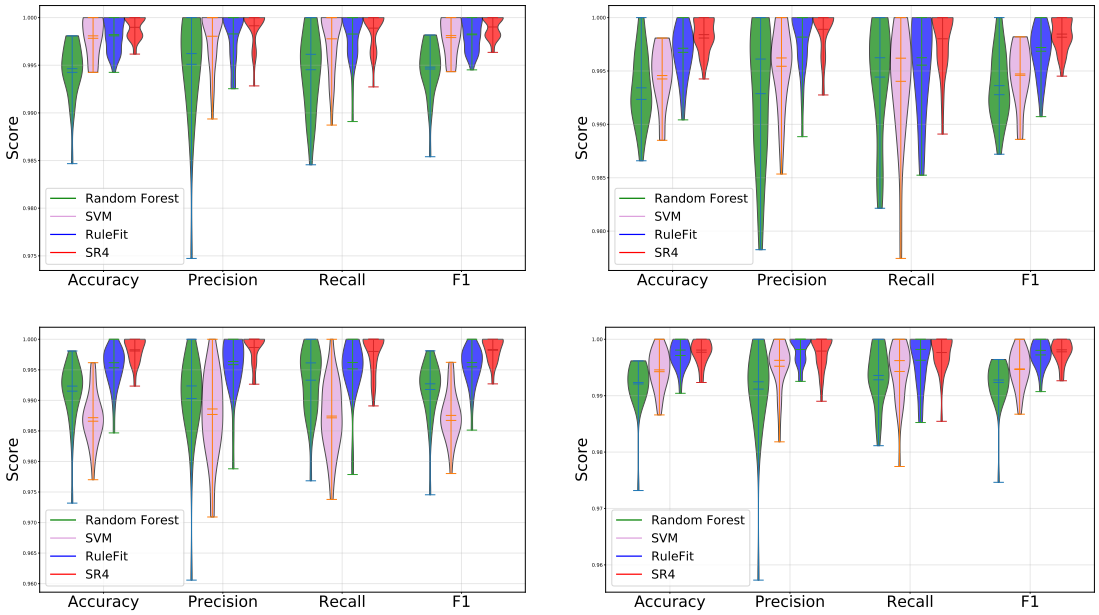



Figure 4.9: Violin plot comparison of model performance metrics across multiple datasets for the election classification task that includes party percentage information. Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

standard, expanded, and previous datasets (0.6432, 0.6594, 0.6738, and 0.6547, respectively). When party percentages are removed, all models experience reductions in interpretability. SR4-Fit produces the best IPS for the expanded dataset, whereas the decision tree produced the best IPS for the minimum and previous datasets but produces very unstable rules across the trails, just like the random forests. The combined violin plots in Figure 4.13 further illustrate that SR4-Fit and RuleFit maintain wider distributions and more stable interpretability distributions, while in the case without including party percentages, the scores are lower and more tightly grouped across models.

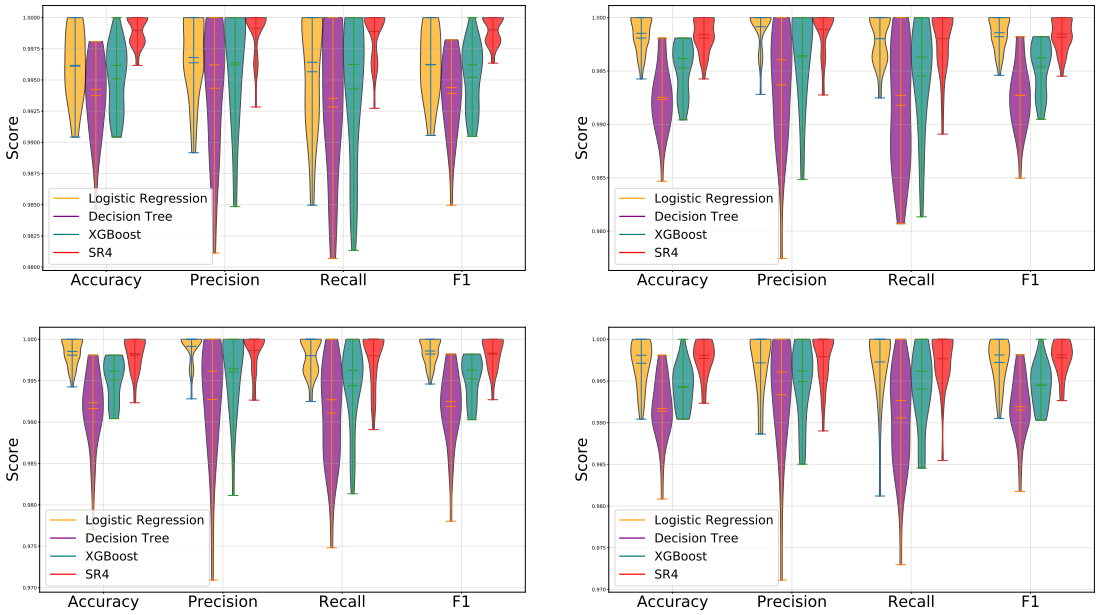



Figure 4.10: Violin plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics across multiple datasets for the election classification task that includes party percentage information. Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

## 4.2 Results of U.S House of Representative elections regression (DEM)

In the minimum dataset, the line plots in Figure 4.14 and Figure 4.15 show that when REP percentage is included as a feature, all models achieve relatively low error values, whereas random forest attains slightly lower average error values compared to other models. The distributional comparisons through violin plots in Figure 4.22 and Figure 4.23 reinforce these findings. When REP percentage is removed, performance declines across all models, with XGBoost attaining slightly lower average error values compared to other models. In contrast, SR4-Fit maintains comparatively smoother trends despite a modest increase in error,

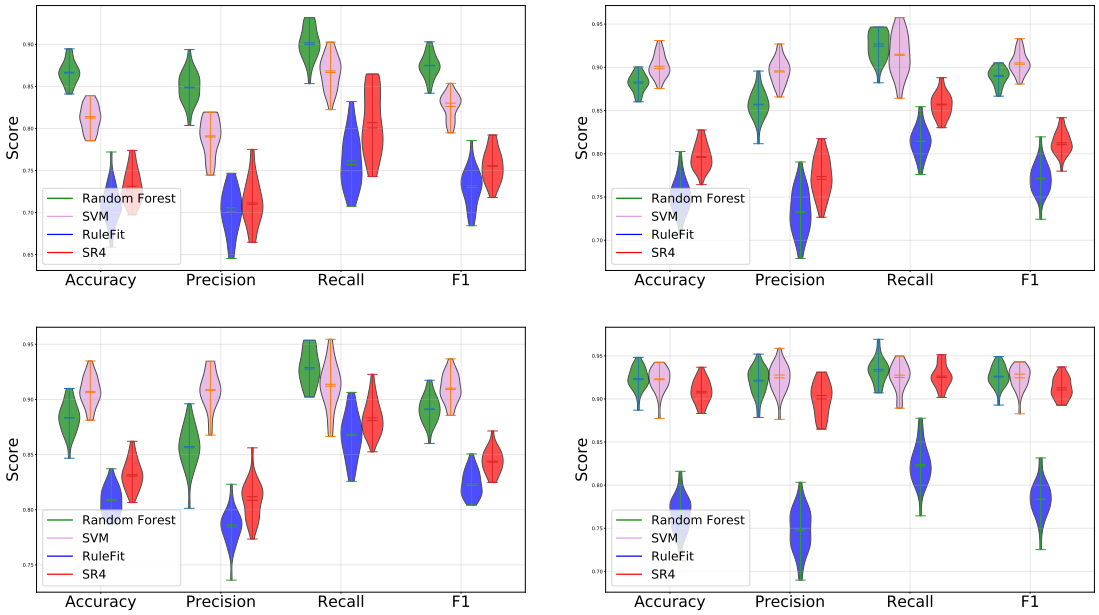



Figure 4.11: Violin plot comparison of model performance metrics across multiple datasets for the election classification task that does not include party percentage information. Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

which is further confirmed in Figure 4.24 and Figure 4.25, where SR4-Fit retains more concentrated distributions. Stability analysis in Table 4.5 shows that with both cases, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models, showing its structural consistency and robustness across different experimental scenarios. Table 4.6 shows that in both cases, the decision tree has the most compact structure, but the structure is too compact, which may lower robustness and may lower the ability to identify interactions properly. Table 4.7 shows that in the case of including REP percentage as a feature, SR4-Fit has the lower average rule complexity, whereas in the case of removing REP percentage as a feature, the decision tree has the overall lower average rule complexity.

In the standard dataset, the line plots in Figure 4.16 and Figure 4.17 show that when REP percentage is included as a feature, all models achieve lower error

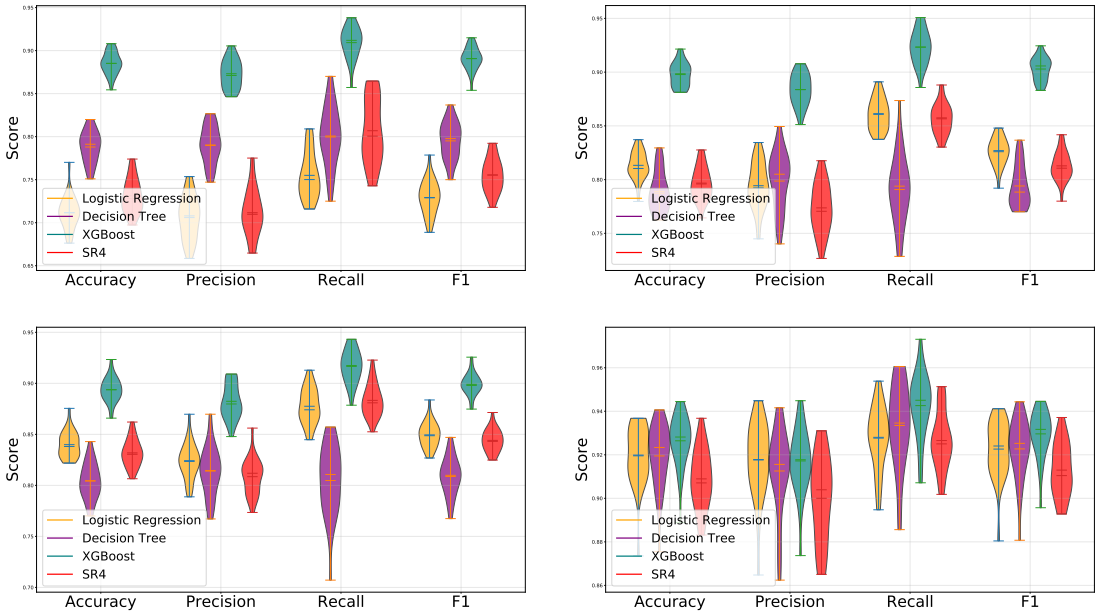



Figure 4.12: Violin plot comparison of model (logistic regression, decision tree, and XGBoost) performance metrics across multiple datasets for the election classification task that does not include party percentage information. Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

values, whereas XGBoost attains slightly lower average error values compared to other models. When the REP percentage is removed, performance declines across all models, with XGBoost still attaining lower average error values compared to other models. Stability analysis in Table 4.5 shows that in the case of including REP percentage as a feature, RuleFit achieves the highest Dice–Sorensen Index among rule-based models, whereas when REP percentage is removed, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.6 shows that in both cases, the decision tree has the most compact structure, trailed closely by SR4-Fit. Table 4.7 shows that in the case of including REP percentage as a feature, RuleFit has the lower average rule complexity, whereas in the case of removing REP percentage as a feature, the SR4-Fit has the overall lower average rule complexity.

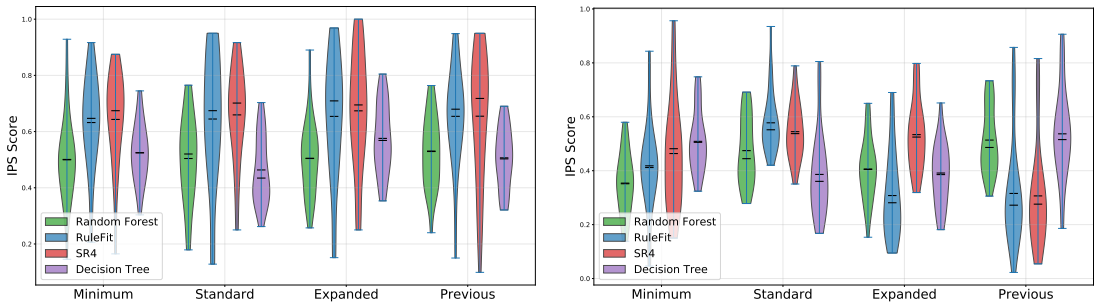



Figure 4.13: Distribution of interpretability scores (IPS) across election classification datasets. The left panel reports results obtained using feature sets that include party percentage information, whereas the right panel shows results with party percentages removed.

In the expanded dataset, the line plots in Figure 4.18 and Figure 4.19 show that when REP percentage is included as a feature, all models achieve lower error values, whereas XGBoost again attains slightly lower average error values compared to other models. When the REP percentage is removed, performance declines across all models, with XGBoost still attaining lower average error values compared to other models. Stability analysis in Table 4.5 shows that in the case of including REP percentage as a feature, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models, whereas when REP percentage is removed, RuleFit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.6 shows that in both cases, the decision tree has the most compact structure. Table 4.7 shows that in the case of including REP percentage as a feature, SR4-Fit has the lower average rule complexity, whereas in the case of removing REP percentage as a feature, the decision tree has the overall lower average rule complexity.

In the previous dataset, the line plots in Figure 4.20 and Figure 4.8 show that when REP percentage is included as a feature, all models achieve lower error values, whereas XGBoost attains slightly lower average error values compared to

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.05±0.01 | **0.48±0.02** | **0.48±0.02** | 0.05±0.03 |
| Standard | 0.03±0.01 | **0.78±0.01** | 0.74±0.01 | 0.04±0.03 |
| Expanded | 0.03±0.01 | **0.85±0.01** | **0.85±0.01** | 0.04±0.02 |
| Previous | 0.06±0.01 | 0.74±0.01 | **0.78±0.01** | 0.04±0.03 |

(a) **With party percentage as a feature.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.02±0.01 | 0.04±0.02 | **0.20±0.01** | 0.01±0.01 |
| Standard | 0.03±0.01 | **0.34±0.00** | 0.21±0.00 | 0.01±0.00 |
| Expanded | 0.01±0.01 | 0.18±0.01 | **0.31±0.01** | 0.01±0.00 |
| Previous | 0.06±0.01 | **0.64±0.01** | **0.64±0.01** | 0.03±0.02 |

(b) **Without party percentage as a feature.**

Table 4.1: Average Dice–Sorensen Index ± standard deviation across 30 trials for election classification datasets. Bold indicates the best-performing method per dataset.

other models. When the REP percentage is removed, performance declines across all models, with XGBoost slightly achieving lower average error values compared to other models. Stability analysis in Table 4.5 shows that with both cases, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.6 shows that in both cases, the decision tree has the most compact structure. Table 4.7 shows that in the case of including REP percentage as a feature, RuleFit has the lower average rule complexity, whereas in the case of removing REP percentage as a feature, the SR4-Fit has the overall lower average rule complexity.

When the Democratic percentage serves as the regression target and when the REP percentage is included as a feature, RuleFit has the highest IPS scores for the standard and expanded datasets, whereas SR4-Fit has high IPS scores across previous datasets Table 4.8, but RuleFit has lower robustness and higher rule complexity when compared to SR4-Fit, showing higher stability and lower

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 86.43±8.70 | 19.33±1.06 | 19.33±1.06 | **9.97±0.76** |
| Standard | 68.00±5.19 | 36.13±0.94 | 38.30±0.99 | **10.03±0.85** |
| Expanded | 82.07±8.99 | 54.13±0.94 | 54.13±0.94 | **10.03±0.93** |
| Previous | 153.93±11.25 | 39.30±0.99 | 37.13±0.94 | **8.77±1.01** |

(a) **With party percentage as a feature.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 89.70±5.16 | 177.80±13.47 | **34.80±1.54** | 253.87±15.9 |
| Standard | **32.50±7.23** | 76.00±0.00 | 126.00±0.00 | 104.00±3.301 |
| Expanded | **23.27±6.69** | 242.73±2.98 | 144.00±0.00 | 90.56±5.67 |
| Previous | 30.03±6.71 | 42.87±0.35 | 42.87±0.35 | **22.03±1.62** |

(b) **Without party percentage as a feature.**

Table 4.2: Average number of rules ± standard deviation across 30 trials for election classification datasets. Bold indicates the most compact model.

rule complexity for SR4-Fit across most dataset configurations. When removing REP percentage, SR4-Fit remains the top performer in two datasets: Expanded (0.6255) and Previous (0.8213), outperforming both Random Forest and Decision Tree and RuleFit. The violin plots in Figure 4.26 show that SR4-Fit maintains more stable interpretability distributions than the baselines, while in the case without including the REP percentage, SR4-Fit maintains compact interpretability distributions.

## 4.3 Results of U.S House of Representative elections regression (REP)

In the minimum dataset, the line plots in Figure 4.27 and Figure 4.28 show that when DEM percentage is included as a feature, all models achieve relatively

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | **1.90±0.03** | 3.00±0.15 | 3.00±0.15 | 4.41±0.20 |
| Standard | 1.91±0.03 | **1.84±0.11** | 2.21±0.11 | 4.42±0.23 |
| Expanded | 1.90±0.02 | **1.60±0.09** | **1.60±0.09** | 4.44±0.30 |
| Previous | 1.92±0.02 | 2.19±0.11 | **1.83±0.11** | 4.03±0.20 |

(a) **With party percentage as a feature.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | **1.99±0.01** | 8.11±0.14 | 4.43±0.07 | 10.95±0.19 |
| Standard | **1.97±0.03** | 5.61±0.32 | 6.79±0.10 | 8.36±0.31 |
| Expanded | **1.98±0.03** | 8.05±0.21 | 7.41±0.32 | 7.55±0.14 |
| Previous | **1.97±0.03** | 2.62±0.03 | 2.62±0.03 | 4.64±0.08 |

(b) **Without party percentage as a feature.**

Table 4.3: Average rule complexity (number of conditions per rule) ± standard deviation across 30 trials. Bold indicates less complex rules.

low error values, whereas XGBoost attains slightly lower average error values compared to other models. The distributional comparisons through violin plots in Figure 4.35 and Figure 4.36 reinforce these findings. When the DEM percentage is removed, performance declines across all models, with XGBoost still achieving lower average error values compared to other models, which is further confirmed in Figure 4.37 and Figure 4.38. Stability analysis in Table 4.9 shows that in the case of including DEM percentage as a feature, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models, whereas when DEM percentage is removed, RuleFit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.10 shows that in both cases, the decision tree has the most compact structure, but the structure is too compact, which may lower robustness and may lower the ability to identify interactions properly. Table 4.11 shows that in the case of including DEM percentage as a feature, the decision tree has the

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.50±0.14 | 0.63±0.16 | **0.64±0.16** | 0.52±0.10 |
| Standard | 0.50±0.15 | 0.64±0.23 | **0.66±0.16** | 0.46±0.11 |
| Expanded | 0.50±0.13 | 0.65±0.21 | **0.67±0.20** | 0.57±0.12 |
| Previous | 0.52±0.12 | 0.65±0.17 | **0.66±0.22** | 0.50±0.10 |

(a) **With party percentages.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.35±0.11 | 0.41±0.16 | 0.46±0.19 | **0.50±0.11** |
| Standard | 0.47±0.12 | **0.57±0.11** | 0.54±0.10 | 0.38±0.15 |
| Expanded | 0.40±0.11 | 0.30±0.15 | **0.53±0.13** | 0.38±0.11 |
| Previous | 0.51±0.12 | 0.31±0.19 | 0.31±0.18 | **0.53±0.16** |

(b) **Without party percentages.**

Table 4.4: Average interpretability score (IPS) ± standard deviation across 30 trials for election classification datasets. Bold indicates the best-performing method per dataset.

lower average rule complexity, whereas in the case of removing DEM percentage as a feature, RuleFit has the overall lower average rule complexity.

In the standard dataset, the line plots in Figure 4.29 and Figure 4.30 show that when DEM percentage is included as a feature, all models achieve lower error values, whereas XGBoost attains slightly lower average error values compared to other models. When the DEM percentage is removed, performance declines across all models, with XGBoost still attaining lower average error values compared to other models. Stability analysis in Table 4.9 shows that in the case of including DEM percentage as a feature, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models, whereas when DEM percentage is removed, RuleFit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.10 shows that in both cases, the decision tree has the most compact structure. Table 4.11 shows that in the case of including DEM percentage as a feature,

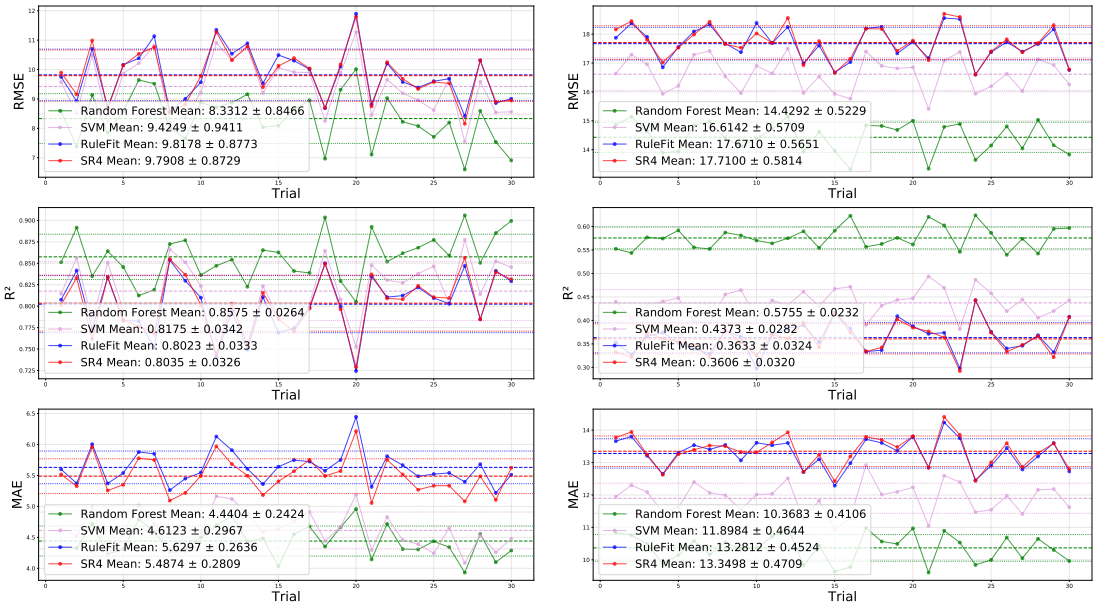



Figure 4.14: Line plot comparison of model performance metrics for minimum data across 30 trials with DEM percentage as label. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

the decision tree has the lower average rule complexity, whereas in the case of removing DEM percentage as a feature, RuleFit has the overall lower average rule complexity.

In the expanded dataset, the line plots in Figure 4.31 and Figure 4.32 show that when DEM percentage is included as a feature, all models achieve lower error values, whereas XGBoost again attains slightly lower average error values compared to other models. When the DEM percentage is removed, performance declines across all models, with XGBoost still attaining lower average error values compared to other models. Stability analysis in Table 4.9 shows that with both cases, SR4-Fit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.10 shows that in both cases, the decision tree has the most compact structure. Table 4.7 shows that in both cases, the decision tree has lower average

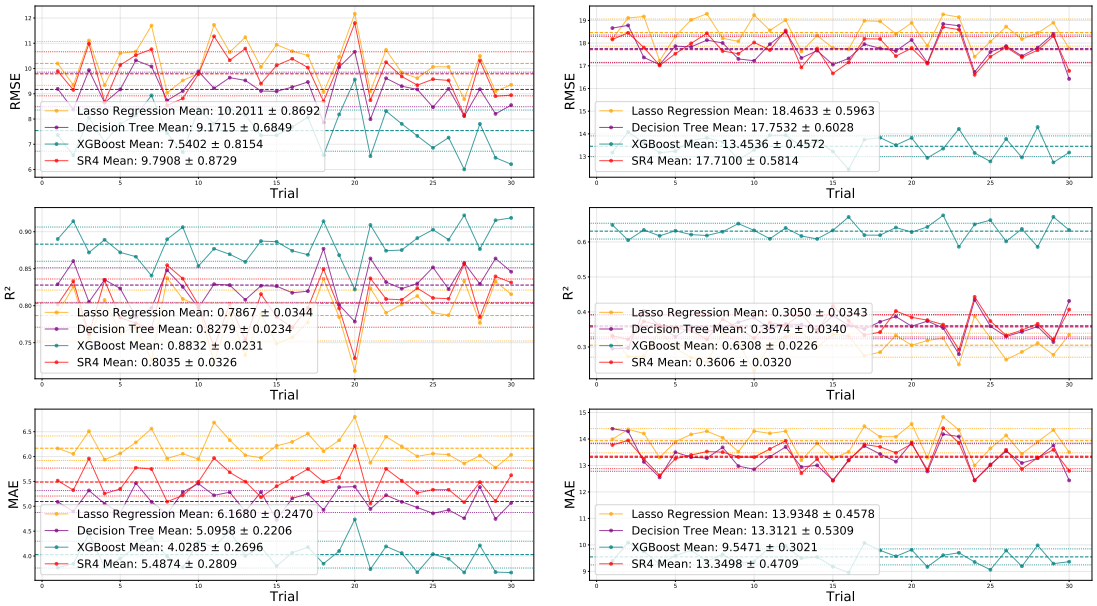



Figure 4.15: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for minimum data across 30 trials. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

rule complexity compared to all models.

In the previous dataset, the line plots in Figure 4.33 and Figure 4.34 show that when DEM percentage is included as a feature, all models achieve lower error values, whereas XGBoost attains lower average error values compared to other models. When the DEM percentage is removed, performance declines across all models, with random forest slightly achieving lower average error values compared to other models. Stability analysis in Table 4.9 shows that with both cases, RuleFit achieves the highest Dice–Sorensen Index among rule-based models. Table 4.10 shows that in both cases, the decision tree has the most compact structure. Table 4.11 shows that in the case of including DEM percentage as a feature, the decision tree has the lower average rule complexity, whereas in the case of removing DEM percentage as a feature, RuleFit has the overall lower

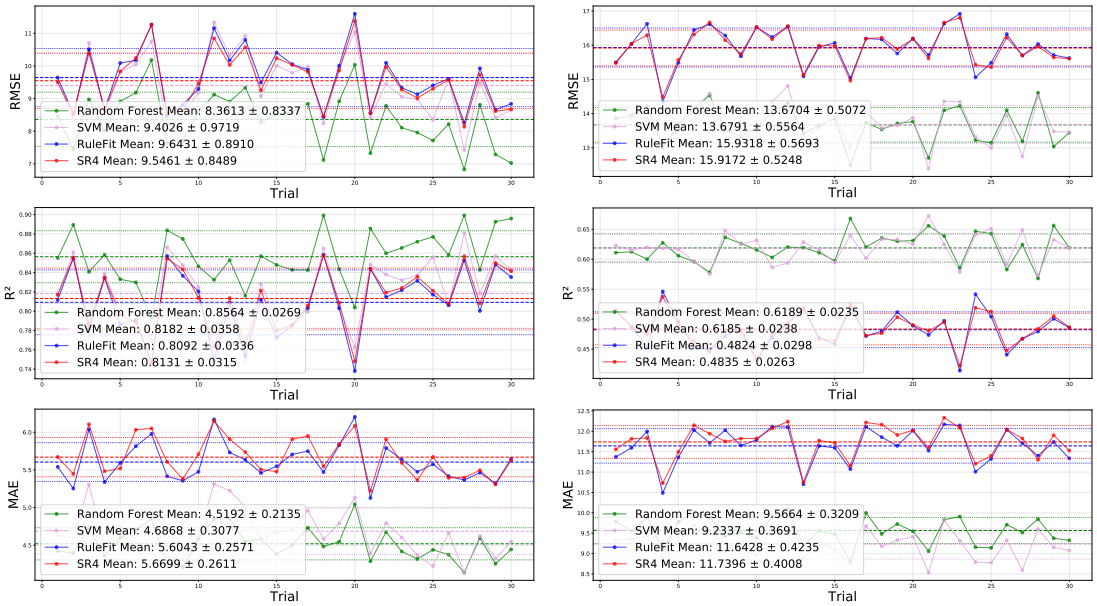



Figure 4.16: Line plot comparison of model performance metrics for standard data across 30 trials with DEM percentage as label. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

average rule complexity.

For the regression task predicting Republican vote percentage, SR4-Fit continues to outperform the baselines across most datasets and feature configurations. As shown in Table 4.12, SR4-Fit achieves the highest IPS in the minimum, standard, and previous datasets when DEM percentage is included (0.6179, 0.5628, and 0.5661, respectively), while RuleFit performs strongest in the expanded dataset. When DEM percentage is removed, interpretability decreases for all models, but SR4-Fit retains superiority in the previous dataset (0.6606). The combined violin plots in Figure 4.39 confirm that SR4-Fit maintains compact interpretability distributions, whereas RuleFit exhibits widening variance and Random Forest performs poorly throughout. These results highlight SR4-Fit's capacity to generate stable and interpretable rule sets for REP vote-share

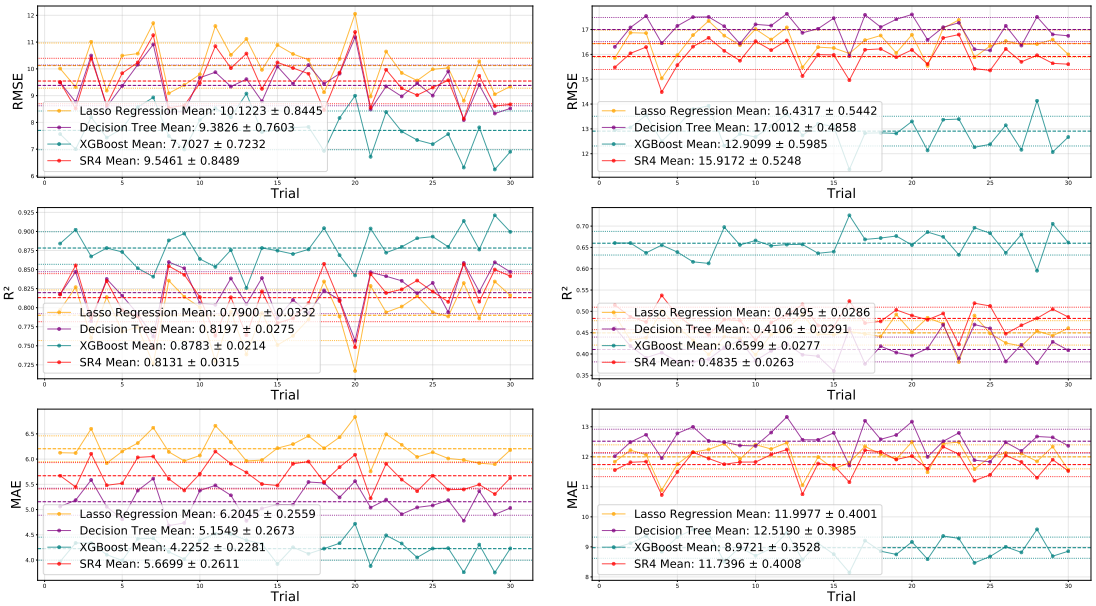



Figure 4.17: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for standard data across 30 trials. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

modeling, even in the absence of politically informative features.

## 4.4 Results on standard datasets classification

Across all six classification datasets and 30 trials, SR4-Fit demonstrates a strong balance of predictive performance, rule stability, and low variance, making it the most robust and interpretable method overall. Although individual models occasionally outperform it on accuracy, SR4-Fit repeatedly delivers the most consistent and stable behavior across trials, as reflected in the line plots and violin plots.

In the Breast Cancer dataset, XGBoost has the highest performance with respect to prediction metrics, but SR4-Fit trails very closely behind XGBoost, as seen in Figure 4.40. The violin plots for breast cancer in Figure 4.46 and Fig-

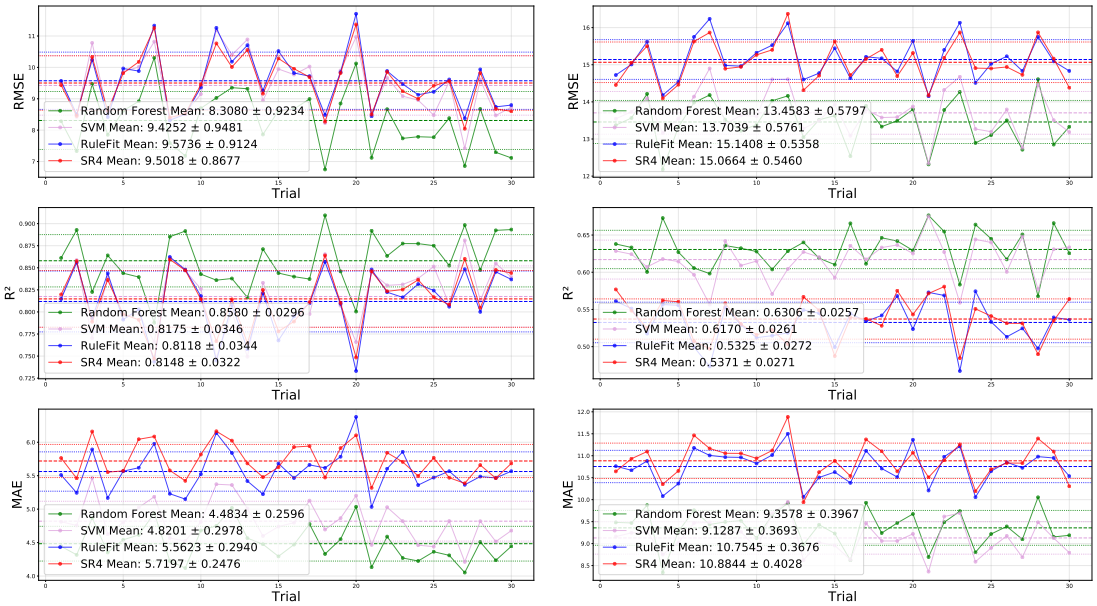



Figure 4.18: Line plot comparison of model performance metrics for expanded data across 30 trials with DEM percentage as label. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

ure 4.47 reinforce these findings. Despite this small difference in accuracy, SR4-Fit significantly outperforms in stability, achieving the highest Dice–Sorensen Index, along with RuleFit, as seen in Table 4.13. Table 4.14 shows that the decision tree has a compact structure, whereas with respect to average rule complexity in Table 4.15 random forest has lower average rule complexity compared to all rule-based models.

In the E. coli dataset, SVM has the highest performance with respect to prediction metrics, as seen in Figure 4.41. The violin plots for E. coli in Figure 4.46 and Figure 4.47 reinforce these findings. SR4-Fit and RuleFit significantly outperform in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.13. Table 4.14 shows that the decision tree has a compact structure, whereas with respect to average rule complexity in Table 4.15 random forest has lower

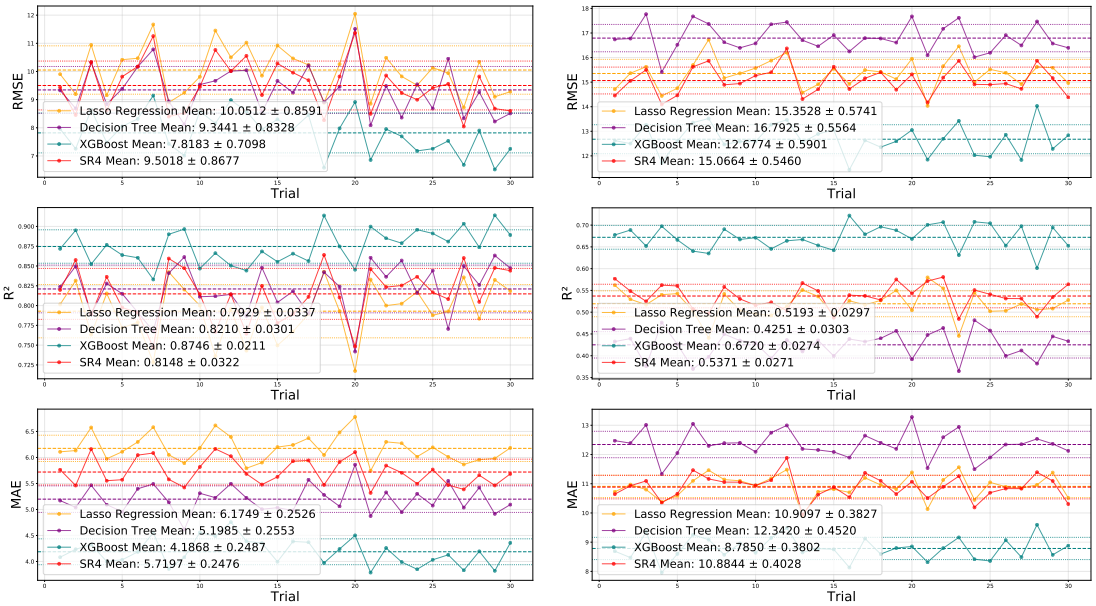



Figure 4.19: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for expanded data across 30 trials. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

average rule complexity compared to all rule-based models.

On the Page blocks dataset, random forest has the highest performance with respect to prediction metrics, but SR4-Fit trails very closely behind random forest, as seen in Figure 4.42. The violin plots for page blocks in Figure 4.46 and Figure 4.47 show the distribution of the prediction metrics across trials. SR4-Fit and RuleFit continue to significantly outperform in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.13. Table 4.14 shows that the decision tree has a compact structure, whereas with respect to average rule complexity in Table 4.15 random forest has lower average rule complexity compared to all rule-based models.

On the Pima Indians dataset, random forest continues to have the highest performance with respect to prediction metrics, but SR4-Fit trails very closely

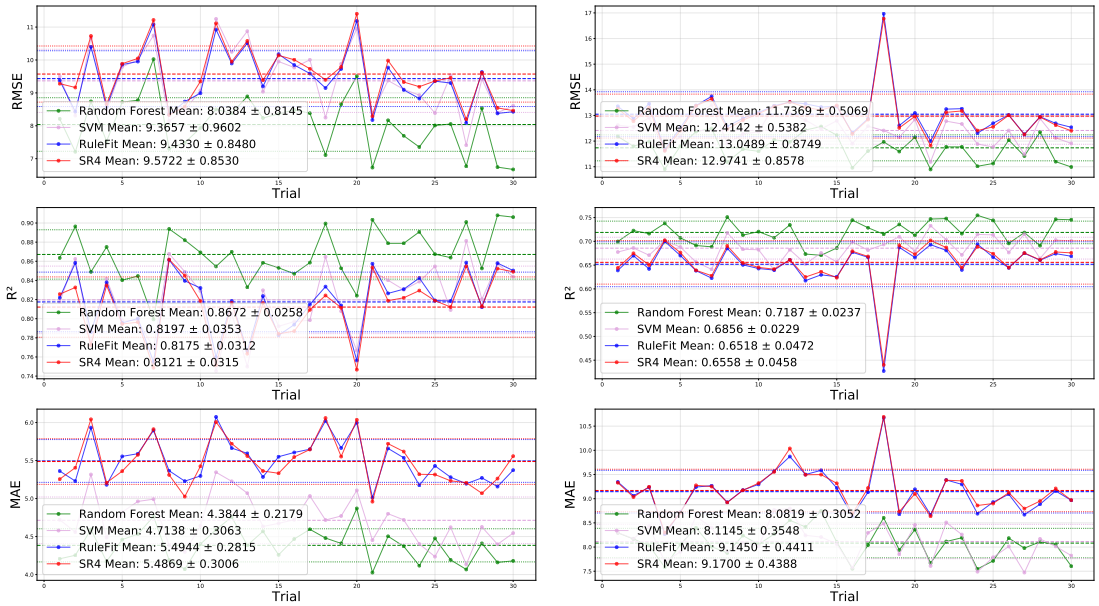



Figure 4.20: Line plot comparison of model performance metrics for previous data across 30 trials with DEM percentage as label. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

behind random forest, as seen in Figure 4.43. The violin plots for Pima Indians in Figure 4.46 and Figure 4.47 show the distribution of the prediction metrics across trials. SR4-Fit and RuleFit continue to significantly outperform in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.13. Table 4.14 shows that the SR4-Fit and RuleFit have a compact structure, whereas, with respect to average rule complexity in Table 4.15 random forests continue to have lower average rule complexity compared to all rule-based models.

In the vehicle dataset, SVM has the highest performance with respect to prediction metrics, but SR4-Fit trails very closely behind SVM, as seen in Figure 4.44. The violin plots for the vehicle in Figure 4.46 and Figure 4.47 show the distribution of the prediction metrics across trials. SR4-Fit and RuleFit continue to significantly outperform in stability, achieving the highest Dice–Sorensen In-

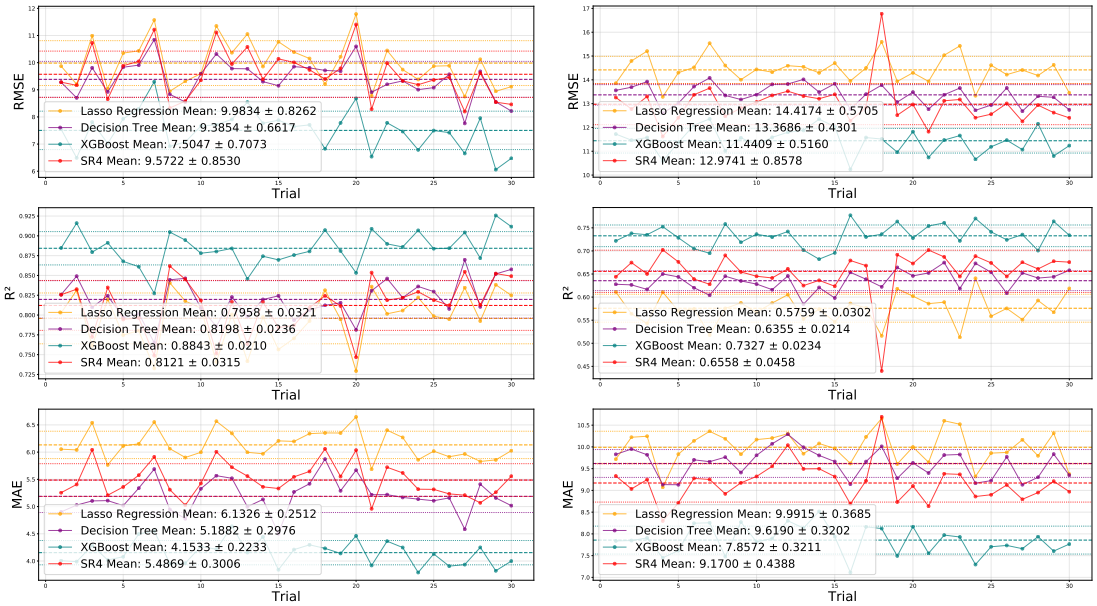



Figure 4.21: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for previous data across 30 trials. The left panel reports results obtained with party percentage (REP) included, whereas the right panel shows results with party percentage (REP) not included.

dex, as seen in Table 4.13. Table 4.14 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.15 random forests continue to have lower average rule complexity compared to all rule-based models.

Finally, in the vehicle dataset, random forest has the highest performance with respect to prediction metrics, as seen in Figure 4.45. The violin plots for the yeast in Figure 4.46 and Figure 4.47 show the distribution of the prediction metrics across trials. SR4-Fit and RuleFit continue to significantly outperform in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.13. Table 4.14 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.15 random forests continue to have lower average rule complexity compared to all rule-based models.

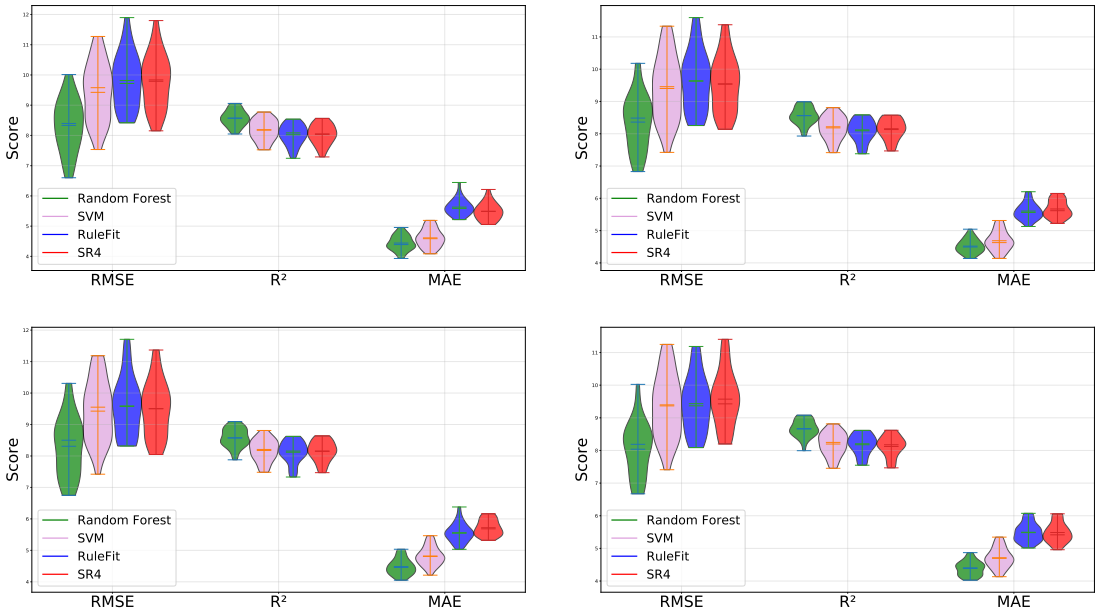



Figure 4.22: Violin plot comparison of model performance metrics across multiple datasets for the election regression task that has DEM percentage as a label and includes party percentage (REP). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

Across all six datasets, the interpretability results in Table 4.16, show that SR4-Fit consistently achieves high interpretability score values across 3 out of the 6 datasets, often outperforming RuleFit, Random Forest, and Decision Tree. Importantly, SR4-Fit attains these interpretability scores while maintaining relatively low variance, indicating consistent interpretability across repeated trials. Figure 4.48 visually reinforces these findings through interpretability score violin plots, which illustrate the distributional behavior of interpretability scores across trials. The violin plots show that SR4-Fit typically has higher medians and broader upper ranges, indicating stronger and more robust interpretability performance.

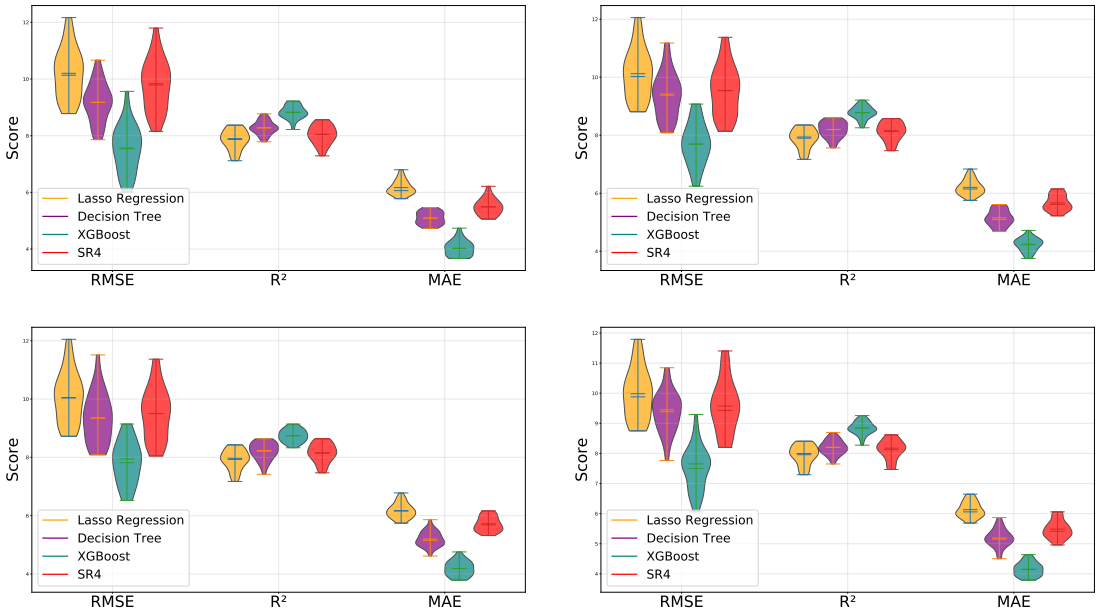



Figure 4.23: Violin plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics across multiple datasets for the election regression task that has DEM percentage as a label and includes party percentage (REP). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

## 4.5 Results on standard datasets regression

Across all eight regression datasets and 30 trials, SR4-Fit demonstrates a strong balance of predictive performance, rule stability, and low variance, making it the most robust and interpretable method overall. Although individual models occasionally outperform it on prediction metrics, SR4-Fit repeatedly delivers the most consistent and stable behavior across trials.

In the abalone dataset, SVM has the lowest error values, but SR4-Fit trails very closely behind SVM, as seen in Figure 4.49. The violin plots for abalone in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit significantly outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure,

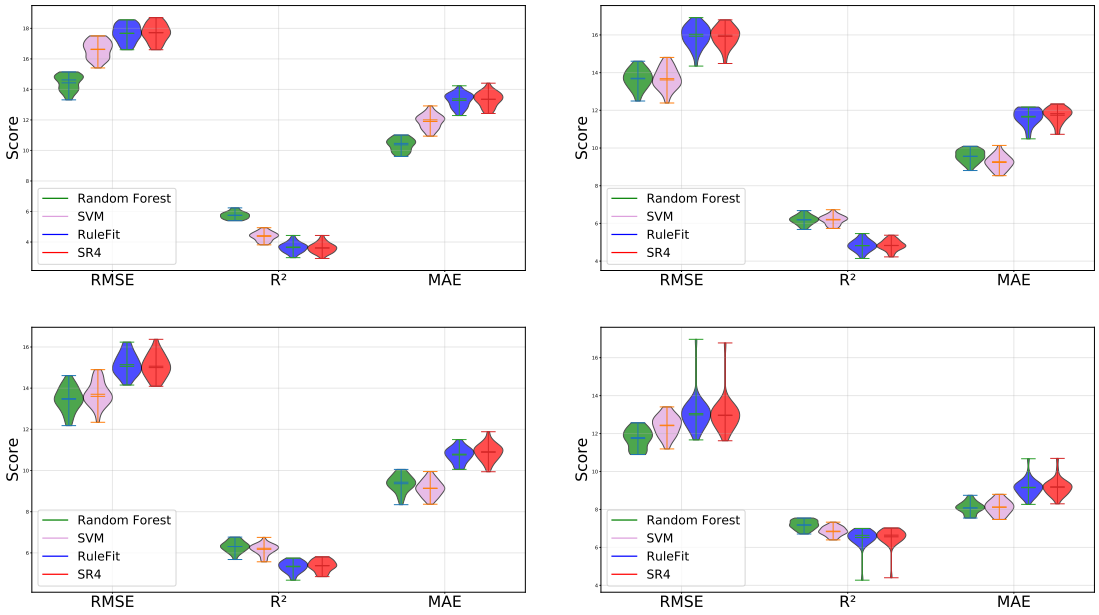



Figure 4.24: Violin plot comparison of model performance metrics across multiple datasets for the election regression task that has DEM percentage as a label and does not include party percentage (REP). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

whereas, with respect to average rule complexity in Table 4.19 decision tree has lower average rule complexity compared to all rule-based models.

In the bone dataset, random forest has the lowest error values, but SR4-Fit trails very closely behind random forest, as seen in Figure 4.50. The violin plots for bone in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit significantly outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 decision tree has lower average rule complexity compared to all rule-based models.

In the diabetes dataset, linear regression has the lowest error values, but SR4-Fit and RuleFit trail very closely behind, as seen in Figure 4.51. The violin plots for diabetes in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit

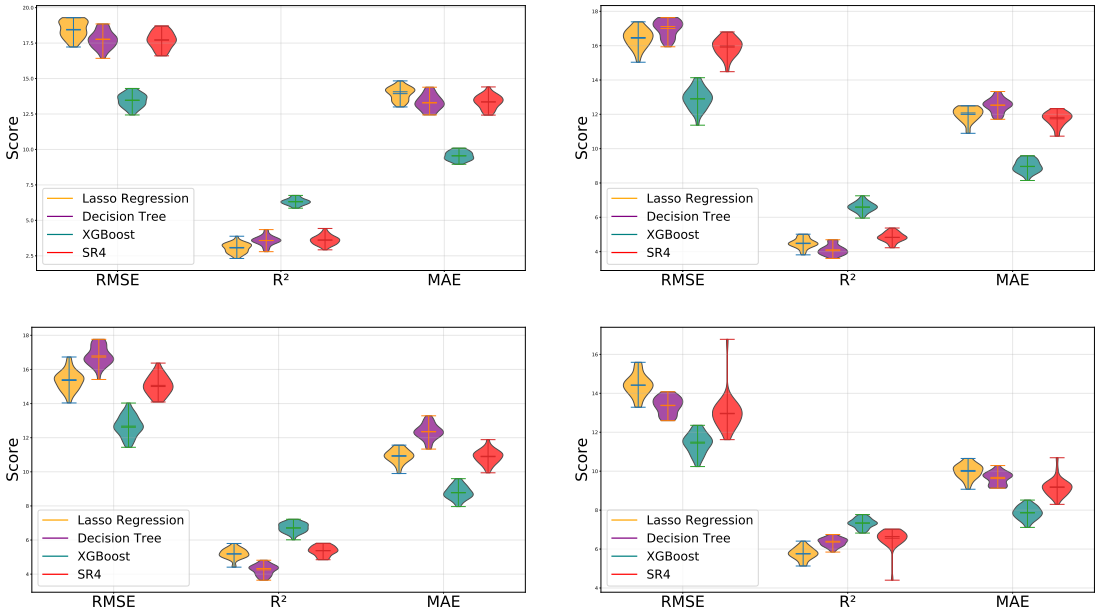



Figure 4.25: Violin plot comparison of model (LASSO regression, decision tree, XG boost) performance metrics across multiple datasets for the election regression task that has DEM percentage as a label and does not include party percentage (REP). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

significantly outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 decision tree has lower average rule complexity compared to all rule-based models.

In the housing dataset, XGBoost has the lowest error values, but SR4-Fit trails very closely behind XGBoost, as seen in Figure 4.52. The violin plots for housing in Figure 4.57 and Figure 4.58 reinforce these findings. RuleFit outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 RuleFit has lower average rule complexity compared to all rule-based models.

In the machine dataset, XGBoost has the lowest error values, but SR4-Fit

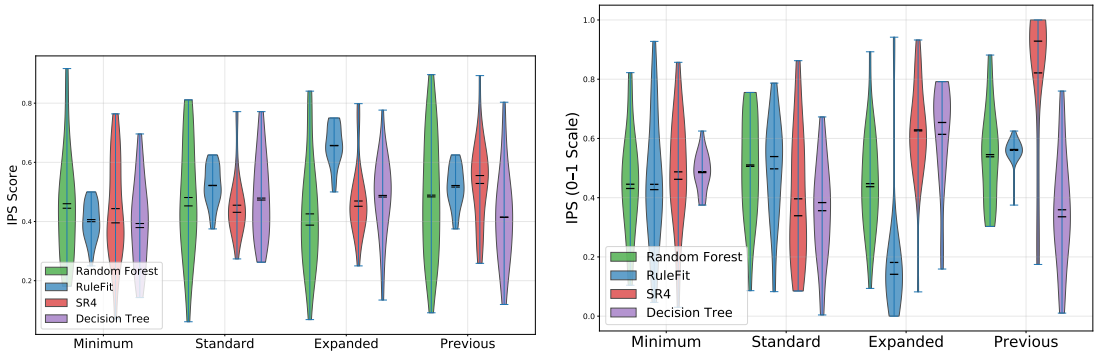



Figure 4.26: Distribution of interpretability scores (IPS) across demographic regression datasets when predicting the Democratic vote percentage. Results are shown for models trained with all party-related features and for models trained with Republican percentage features removed.

trails very closely behind XGBoost, as seen in Figure 4.53. The violin plots for the machine in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 SR4-Fit has lower average rule complexity compared to all rule-based models.

In the MPG dataset, SVM has the lowest error values, but SR4-Fit and RuleFit trail very closely behind, as seen in Figure 4.54. The violin plots for MPG in Figure 4.57 and Figure 4.58 reinforce these findings. The decision tree outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 decision tree has lower average rule complexity compared to all rule-based models.

In the ozone dataset, SVM has the lowest error values, but SR4-Fit trails very closely behind, as seen in Figure 4.55. The violin plots for the ozone in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit outperforms in stability, achiev-

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.01±0.00 | 0.33±0.00 | **0.39±0.01** | 0.01±0.01 |
| Standard | 0.01±0.00 | **0.77±0.00** | 0.75±0.01 | 0.01±0.01 |
| Expanded | 0.01±0.00 | 0.73±0.00 | **0.83±0.01** | 0.01±0.01 |
| Previous | 0.01±0.00 | **0.77±0.00** | 0.68±0.01 | 0.01±0.01 |

(a) **DEM as a label, with REP percentage**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.03±0.01 | 0.10±0.01 | **0.27±0.01** | 0.01±0.00 |
| Standard | 0.02±0.01 | 0.30±0.01 | **0.54±0.02** | 0.01±0.01 |
| Expanded | 0.01±0.01 | **0.73±0.01** | 0.66±0.01 | 0.35±0.23 |
| Previous | 0.02±0.05 | 0.77±0.00 | **0.81±0.01** | 0.09±0.10 |

(b) **DEM as a label, without REP percentage.**

Table 4.5: Average Dice–Sorensen Index ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label). Bold indicates the best-performing method per dataset.

ing the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 SR4-Fit has lower average rule complexity compared to all rule-based models.

In the prostate dataset, SVM has the lowest error values, but SR4-Fit trails very closely behind, as seen in Figure 4.56. The violin plots for the ozone in Figure 4.57 and Figure 4.58 reinforce these findings. SR4-Fit outperforms in stability, achieving the highest Dice–Sorensen Index, as seen in Table 4.17. Table 4.18 shows that the decision tree has a compact structure, whereas, with respect to average rule complexity in Table 4.19 SR4-Fit has lower average rule complexity compared to all rule-based models.

Across all eight datasets, the interpretability results in Table 4.20, show that SR4-Fit and the decision tree achieve high interpretability score values across

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 8086.66±199.12 | 24.00±0.00 | 20.63±1.43 | **9.03±1.45** |
| Standard | 8162.07±233.69 | 35.00±0.00 | 34.13±1.03 | **12.33±1.03** |
| Expanded | 7972.60±224.35 | 61.00±0.00 | 52.10±1.09 | **11.56±1.69** |
| Previous | 7856.70±222.97 | 36.00±0.00 | 38.83±1.32 | **11.10±1.21** |

(a) **DEM as a label, with REP percentage.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 8869.63±179.67 | 67.87±1.82 | 26.13±2.46 | **1.00±0.00** |
| Standard | 9024.40±202.97 | 85.70±1.84 | 46.13±3.35 | **8.22±2.15** |
| Expanded | 8654.03±138.38 | 59.93±0.25 | 61.70±2.31 | **1.00±0.00** |
| Previous | 7946.13±164.93 | 35.00±0.00 | 32.30±0.65 | **1.53±0.50** |

(b) **DEM as a label, without REP percentage.**

Table 4.6: Average number of rules ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label). Bold indicates the most compact model.

3 out of the 8 datasets each, often outperforming RuleFit and Random Forest. Figure 4.59 visually reinforces these findings through interpretability score violin plots, which illustrate the distributional behavior of interpretability scores across trials.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 6.25±0.03 | 3.00±0.00 | **2.84±0.01** | 4.81±0.14 |
| Standard | 6.17±0.03 | **1.46±0.00** | 1.46±0.01 | 4.80±0.08 |
| Expanded | 6.09±0.03 | 1.78±0.00 | **1.30±0.01** | 4.75±0.10 |
| Previous | 6.11±0.03 | **1.44±0.00** | 1.89±0.05 | 4.73±0.01 |

(a) **DEM as a label, with REP percentage.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 6.36±0.02 | 5.44±0.04 | 4.64±0.13 | **1.01±0.00** |
| Standard | 6.39±0.02 | 4.44±0.05 | **3.28±0.21** | 4.73±0.16 |
| Expanded | 6.34±0.014 | 1.79±0.01 | 2.68±0.12 | **1.2±0.61** |
| Previous | 6.25±0.03 | 1.46±0.00 | **1.39±0.03** | 2.80±0.61 |

(b) **DEM as a label, without REP percentage.**

Table 4.7: Average rule complexity ± standard deviation across 30 trials for the election regression dataset (DEM percentage as label). Bold indicates less complex rules.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.44±0.18 | 0.39±0.06 | **0.44±0.16** | 0.39±0.13 |
| Standard | 0.48±0.19 | **0.52±0.06** | 0.45±0.10 | 0.47±0.13 |
| Expanded | 0.42±0.18 | **0.65±0.06** | 0.46±0.11 | 0.48±0.13 |
| Previous | 0.48±0.20 | 0.51±0.06 | **0.53±0.13** | 0.41±0.15 |

(a) **With the Republican percentage as feature.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.44±0.17 | 0.44±0.21 | 0.48±0.18 | **0.48±0.06** |
| Standard | **0.50±0.17** | 0.49±0.16 | 0.39±0.21 | 0.35±0.16 |
| Expanded | 0.44±0.16 | 0.18±0.20 | **0.62±0.16** | 0.61±0.16 |
| Previous | 0.53±0.15 | 0.55±0.04 | **0.82±0.25** | 0.35±0.18 |

(b) **Without the Republican percentage as feature.**

Table 4.8: Average interpretability score (IPS) ± standard deviation across 30 trials for election regression datasets. Results are reported for predicting Democratic vote percentage, with and without Republican percentage included as a feature. Bold indicates the best-performing method per dataset.

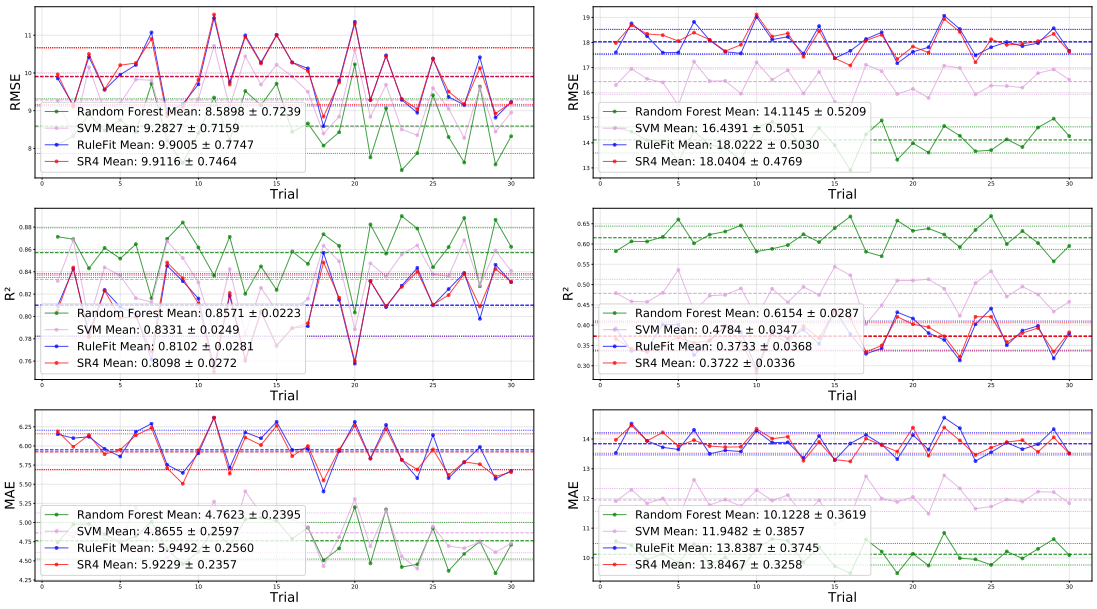



Figure 4.27: Line plot comparison of model performance metrics for minimum data across 30 trials with REP percentage as label. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

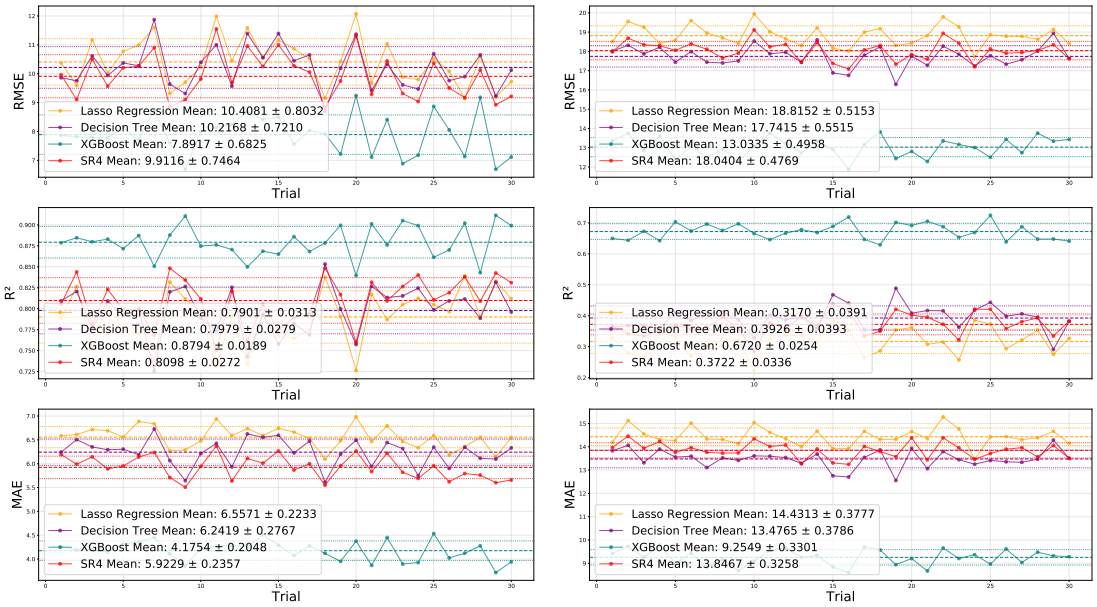



Figure 4.28: Line plot comparison of models (LASSO regression, decision tree, and XGBoost) performance metrics for minimum data across 30 trials. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

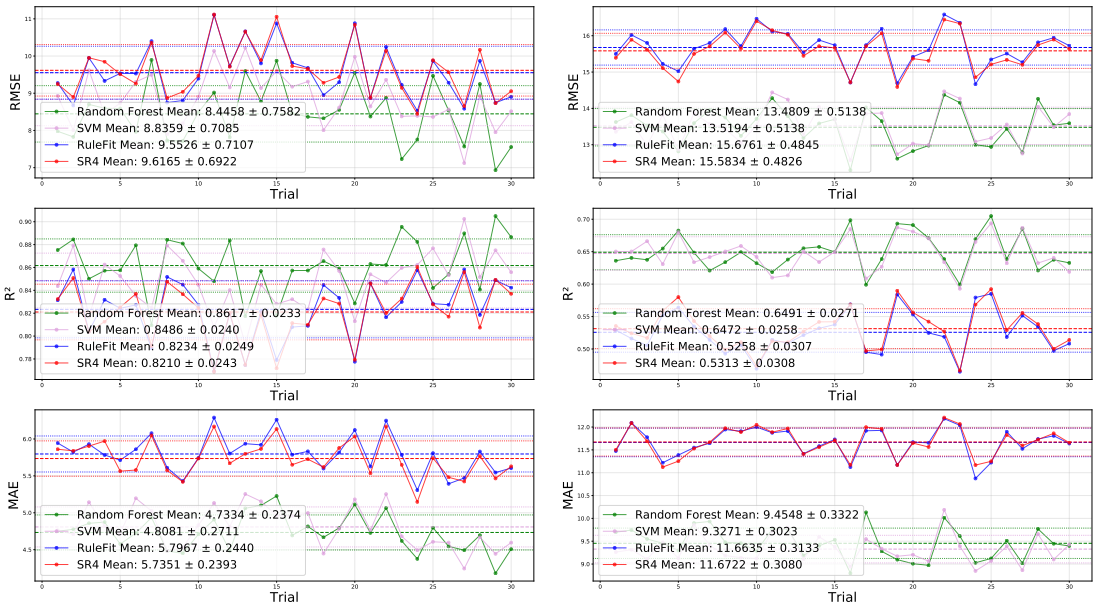



Figure 4.29: Line plot comparison of model performance metrics for standard data across 30 trials with REP percentage as label. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

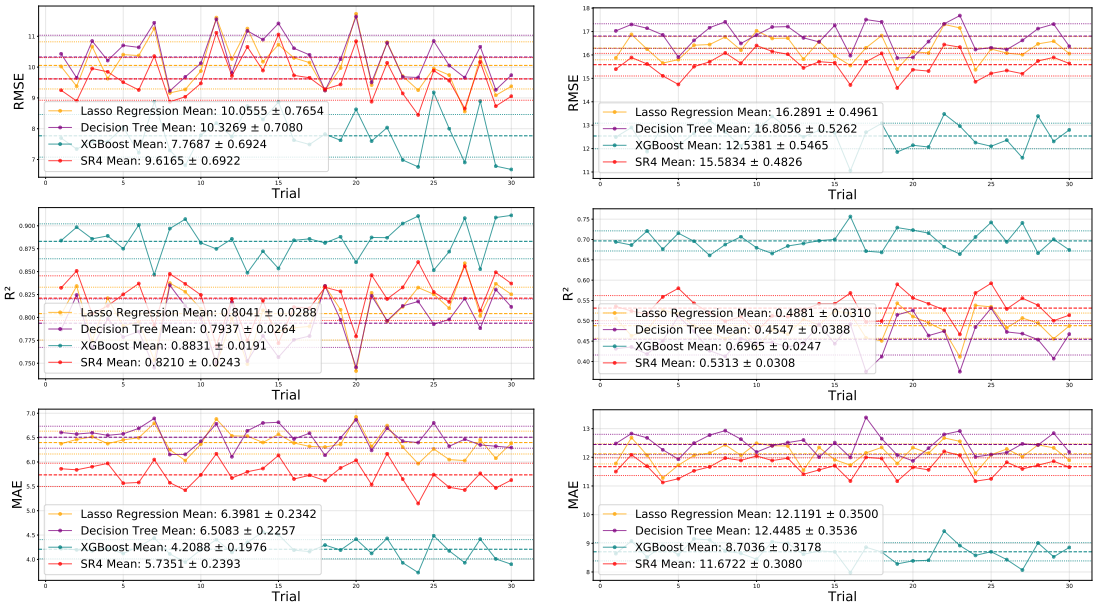



Figure 4.30: Line plot comparison of models (LASSO regression, decision tree, and XGBoost) performance metrics for standard data across 30 trials. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

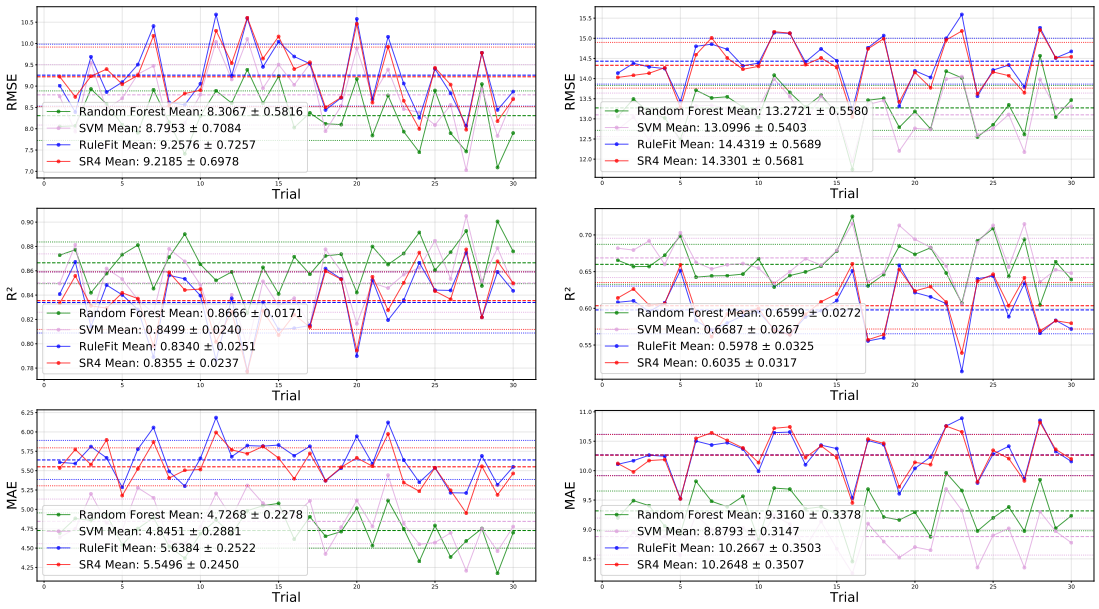



Figure 4.31: Line plot comparison of model performance metrics for expanded data across 30 trials with REP percentage as label. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

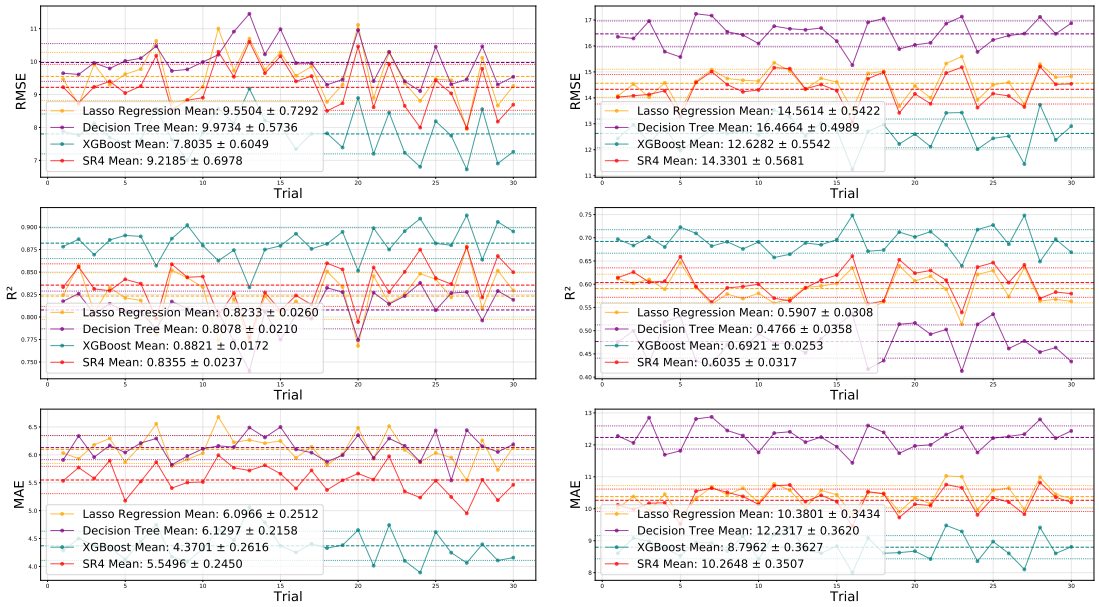



Figure 4.32: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for expanded data across 30 trials. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

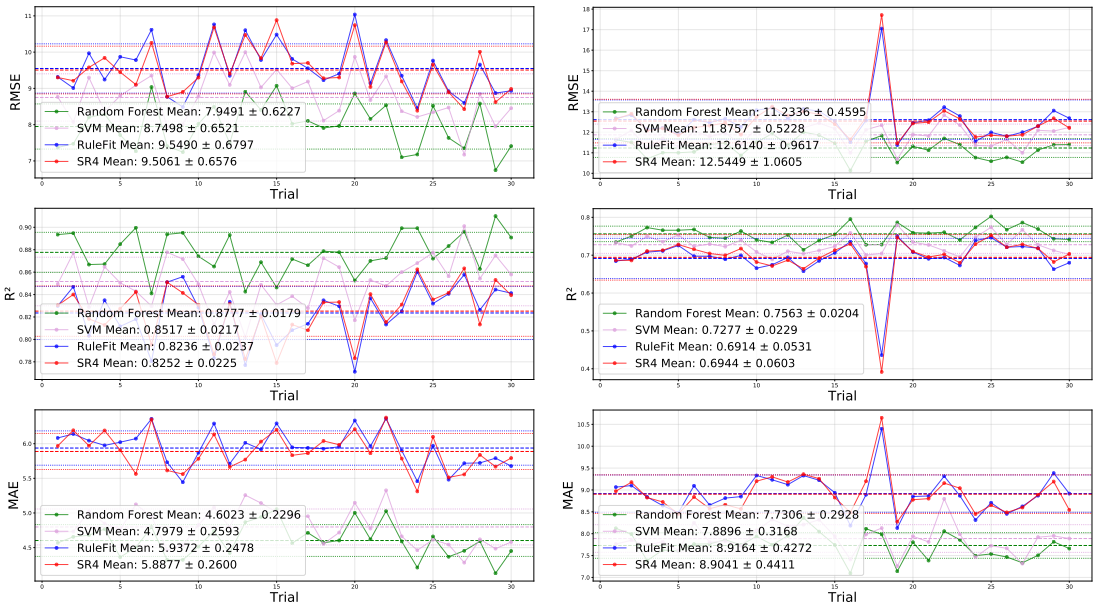



Figure 4.33: Line plot comparison of model performance metrics for previous data across 30 trials with REP percentage as label. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

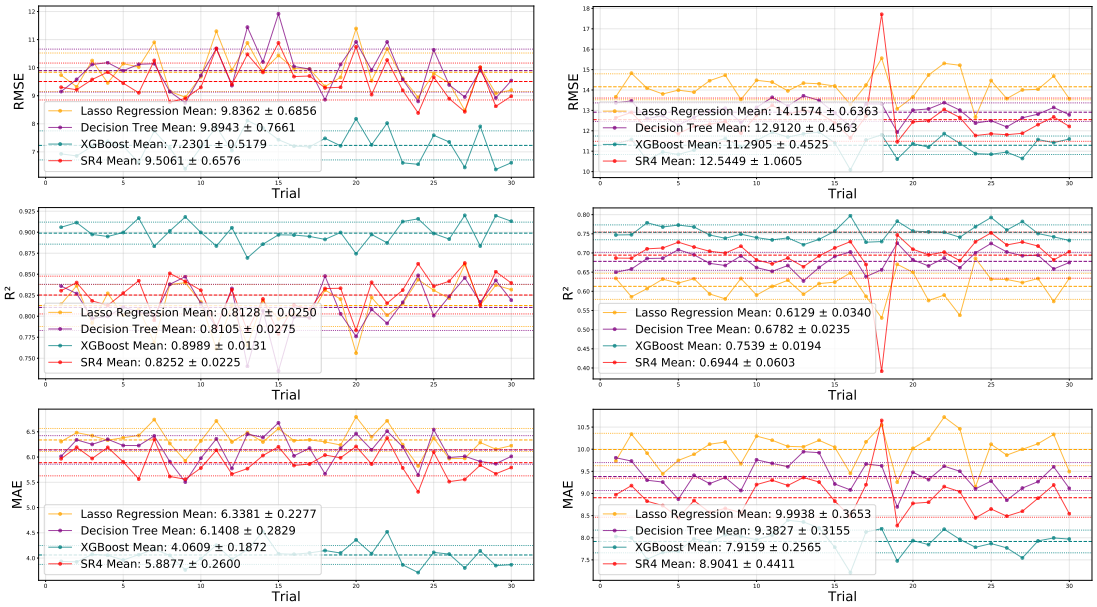



Figure 4.34: Line plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics for previous data across 30 trials. The left panel reports results obtained with party percentage (DEM) included, whereas the right panel shows results with party percentage (DEM) not included.

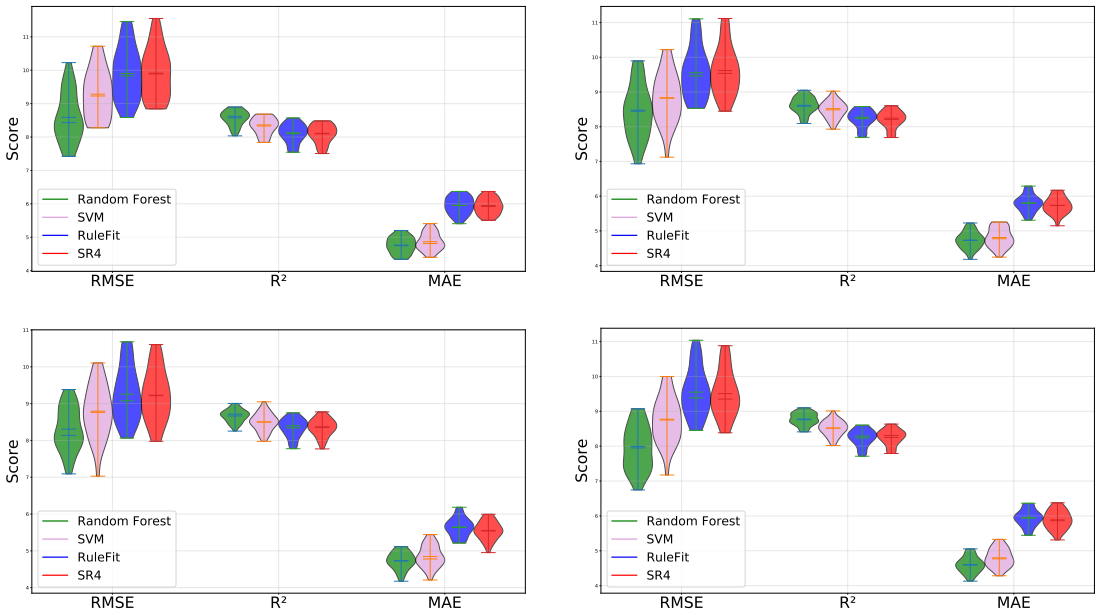



Figure 4.35: Violin plot comparison of model performance metrics across multiple datasets for the election regression task that has REP percentage as a label and includes party percentage (DEM). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.01±0.00 | 0.33±0.00 | **0.49±0.01** | 0.01±0.01 |
| Standard | 0.01±0.00 | 0.62±0.00 | **0.68±0.01** | 0.01±0.01 |
| Expanded | 0.01±0.00 | 0.73±0.00 | **0.78±0.01** | 0.01±0.01 |
| Previous | 0.01±0.00 | **0.77±0.0000** | 0.71±0.01 | 0.01±0.01 |

(a) **REP as a label, with DEM percentage.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.04±0.01 | **0.30±0.01** | 0.27±0.01 | 0.01±0.01 |
| Standard | 0.01±0.00 | **0.76±0.00** | 0.76±0.01 | 0.01±0.01 |
| Expanded | 0.01±0.00 | 0.73±0.00 | **0.75±0.01** | 0.41±0.27 |
| Previous | 0.01±0.05 | **0.77±0.00** | 0.68±0.01 | 0.13±0.11 |

(b) **REP as a label, without DEM percentage.**

Table 4.9: Average Dice–Sorensen Index ± standard deviation across 30 trials for the election regression dataset (REP percentage as label). Bold indicates the best-performing method per dataset.

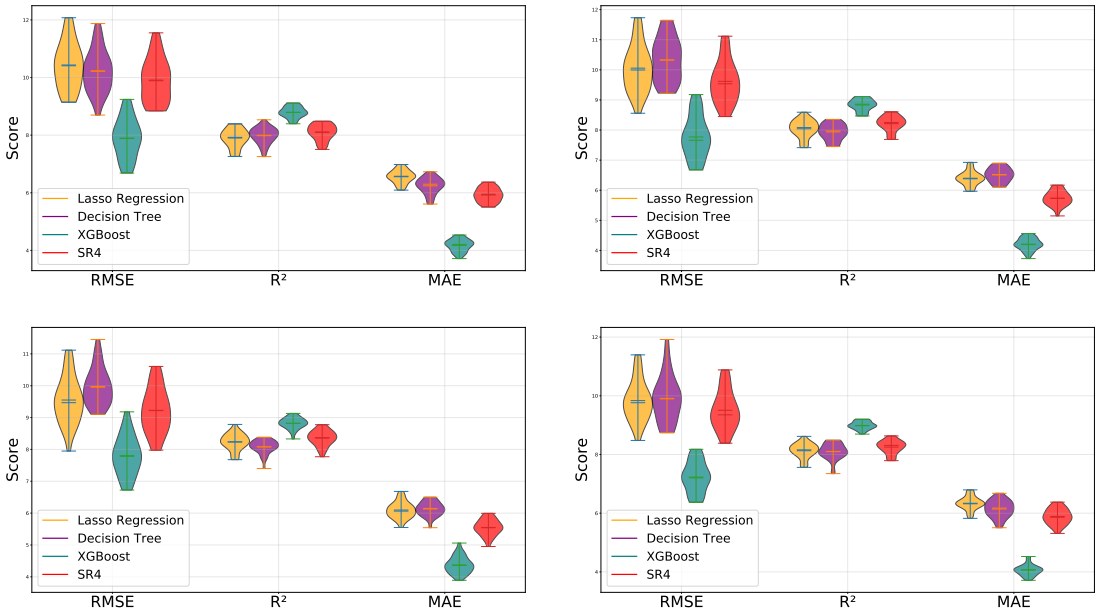



Figure 4.36: Violin plot comparison of model (LASSO regression, decision tree, and XGBoost) performance metrics across multiple datasets for the election regression task that has REP percentage as a label and includes party percentage (DEM). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

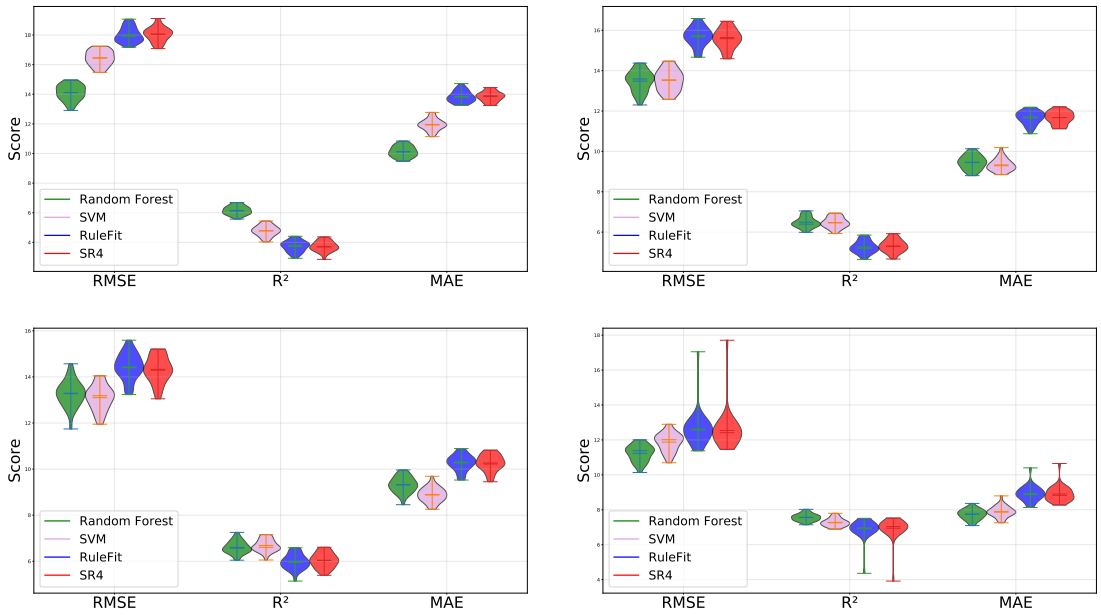



Figure 4.37: Violin plot comparison of model performance metrics across multiple datasets for the election regression task that has REP percentage as a label and does not include party percentage (DEM). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 8161.43±200.06 | 24.00±0.00 | 15.86±0.34 | **1.00±0.00** |
| Standard | 8225.93±225.83 | 43.00±0.00 | 39.00±0.52 | **1.00±0.00** |
| Expanded | 8457.63±232.08 | 61.00±0.00 | 56.96±1.12 | **1.00±0.00** |
| Previous | 8205.06±199.47 | 36.00±0.00 | 39.26±0.78 | **1.00±0.00** |

(a) **REP as a label, with DEM percentage.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 8797.16±179.66 | 22.96±0.18 | 26.36±2.41 | **10.50±2.28** |
| Standard | 9513.73±133.31 | 34.00±0.00 | 32.96±0.18 | **10.13±1.45** |
| Expanded | 9256.36±162.12 | 60.00±0.00 | 55.03±1.15 | **1.00±0.00** |
| Previous | 8044.26±152.79 | 35.00±0.00 | 38.16±0.91 | **1.60±0.49** |

(b) **REP as a label, without DEM percentage.**

Table 4.10: Average number of rules ± standard deviation across 30 trials for the election regression dataset (REP percentage as label). Bold indicates the most compact model.

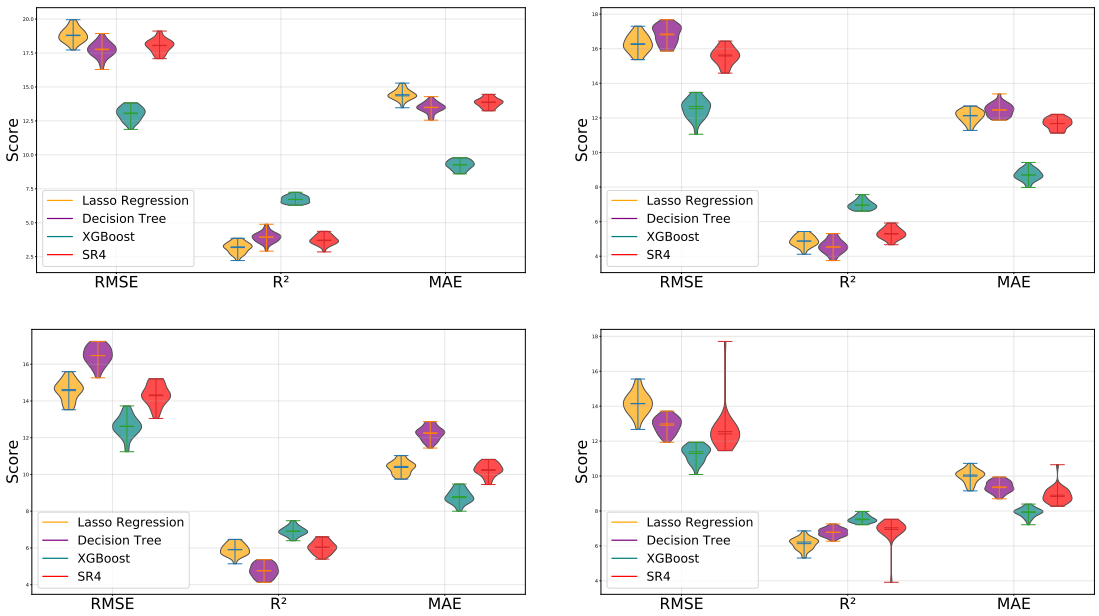



Figure 4.38: Violin plot comparison of model (LASSO regression, decision tree, XG boost) performance metrics across multiple datasets for the election regression task that has REP percentage as a label and does not include party percentage (DEM). Top left: Minimum data; top right: Standard data; bottom left: Expanded data; bottom right: Previous data.

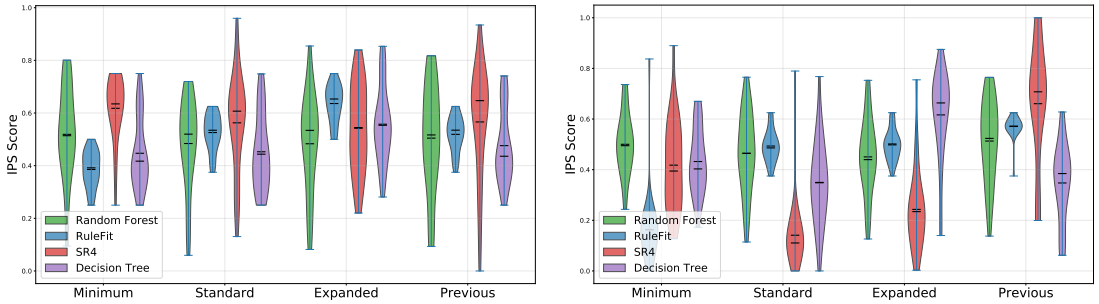



Figure 4.39: Distribution of interpretability scores (IPS) across demographic regression datasets when predicting Republican vote percentage. Results are shown for models trained with all party-related features and for models trained with Democratic percentage features removed.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 6.23±0.03 | 3.00±0.00 | 2.01±0.02 | **1.00±0.00** |
| Standard | 6.17±0.03 | 2.11±0.00 | 1.93±0.03 | **1.00±0.00** |
| Expanded | 6.18±0.03 | 1.78±0.00 | 1.64±0.03 | **1.20±0.61** |
| Previous | 6.16±0.03 | 1.44±0.00 | 1.86±0.04 | **1.00±0.00** |

(a) **REP as a label, with DEM percentage.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 6.32±0.02 | **3.08±0.02** | 4.66±0.13 | 4.59±0.17 |
| Standard | 6.42±0.01 | **1.47±0.00** | 1.48±0.01 | 4.73±0.13 |
| Expanded | 6.39±0.03 | 1.80±0.00 | 1.76±0.05 | **1.20±0.61** |
| Previous | 6.30±0.02 | **1.45±0.00** | 1.95±0.05 | 2.86±0.50 |

(b) **REP as a label, without DEM percentage.**

Table 4.11: Average rule complexity ± standard deviation across 30 trials for the election regression dataset (REP percentage as label). Bold indicates less complex rules.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | 0.51±0.17 | 0.38±0.06 | **0.61±0.12** | 0.44±0.13 |
| Standard | 0.48±0.17 | 0.52±0.06 | **0.56±0.18** | 0.45±0.14 |
| Expanded | 0.48±0.18 | **0.63±0.06** | 0.54±0.16 | 0.55±0.14 |
| Previous | 0.50±0.20 | 0.51±0.06 | **0.56±0.21** | 0.47±0.13 |

(a) **With the Democratic percentage as feature.**

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Minimum | **0.50±0.11** | 0.16±0.14 | 0.41±0.17 | 0.43±0.12 |
| Standard | 0.46±0.15 | **0.49±0.06** | 0.14±0.13 | 0.34±0.16 |
| Expanded | 0.45±0.14 | 0.49±0.06 | 0.24±0.15 | **0.61±0.17** |
| Previous | 0.51±0.16 | 0.57±0.04 | **0.66±0.23** | 0.34±0.15 |

(b) **Without the Democratic percentage as feature.**

Table 4.12: Average interpretability score (IPS) ± standard deviation across 30 trials for demographic regression datasets. Results are reported for predicting Republican vote percentage, with and without Democratic percentage included as a feature. Bold indicates the best-performing method per dataset.

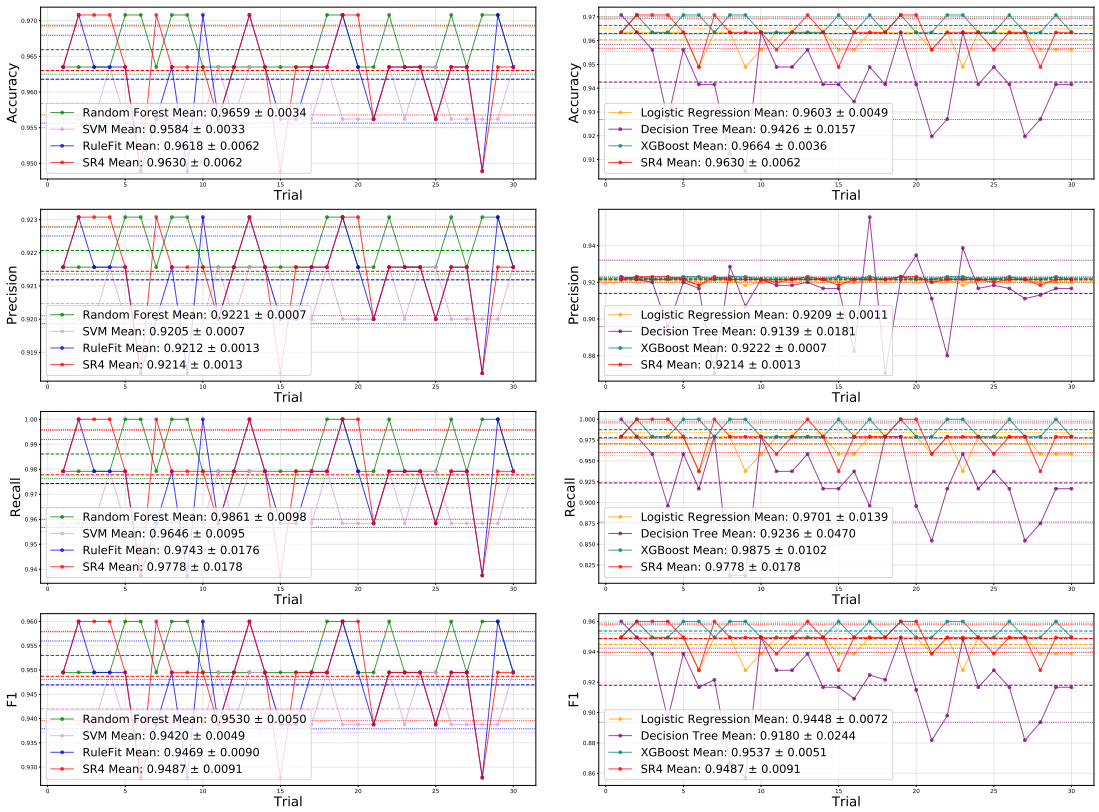



Figure 4.40: Line plot comparison of model performance metrics for breast cancer data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit , whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| **Dataset** | **Random Forest** | **RuleFit** | **SR4-Fit** | **Decision Tree** |
|---|---|---|---|---|
| Breast Cancer | 0.33±0.02 | **0.44±0.02** | **0.44±0.02** | 0.05±0.05 |
| E. coli | 0.11±0.01 | **0.17±0.01** | **0.17±0.01** | 0.04±0.03 |
| Page Blocks | 0.10±0.01 | **0.32±0.02** | **0.32±0.02** | 0.01±0.01 |
| Pima Indians | 0.04±0.01 | **0.51±0.01** | **0.51±0.01** | 0.02±0.01 |
| Vehicle | 0.19±0.01 | **0.52±0.02** | **0.52±0.02** | 0.05±0.03 |
| Yeast | 0.11±0.01 | **0.18±0.01** | **0.18±0.01** | 0.01±0.01 |

Table 4.13: Average Dice–Sorensen Index ± standard deviation across 30 trials for public classification datasets. Bold indicates the best-performing method per dataset.

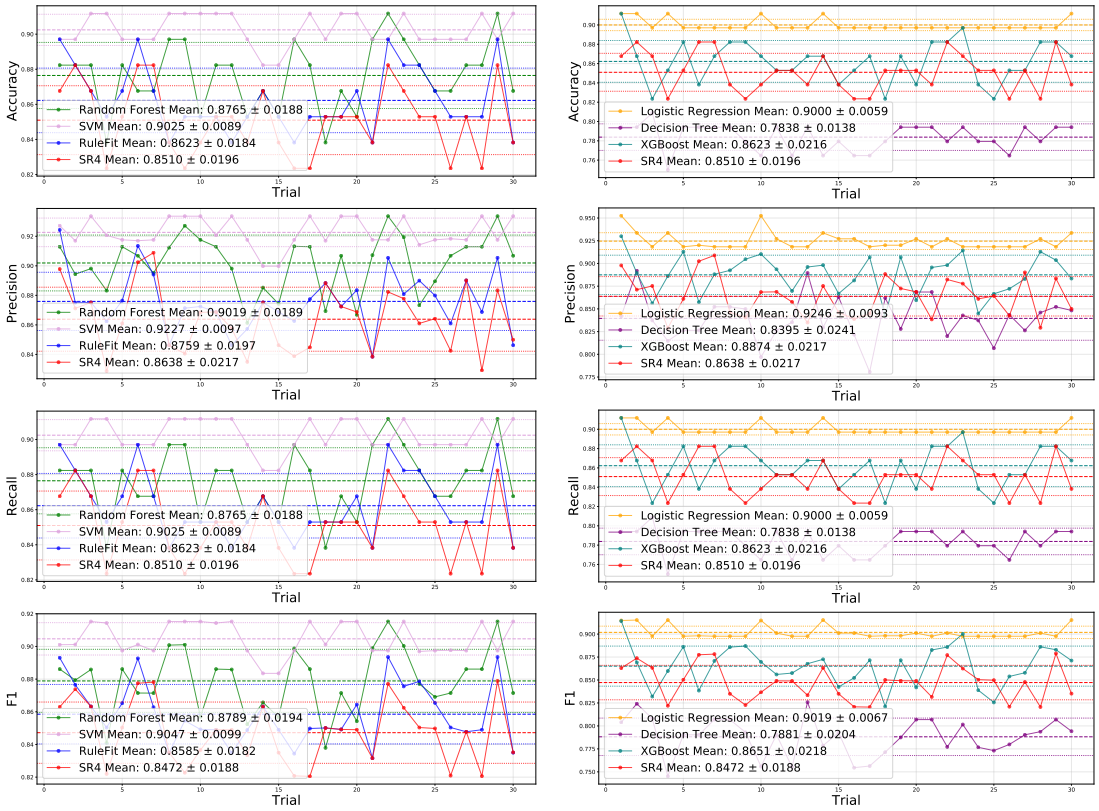



Figure 4.41: Line plot comparison of model performance metrics for E. coli data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit , whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| **Dataset** | **Random Forest** | **RuleFit** | **SR4-Fit** | **Decision Tree** |
|---|---|---|---|---|
| Breast Cancer | 104.43±11.58 | 21.13±1.11 | 21.13±1.11 | **9.10±0.66** |
| E. coli | 117.17±13.48 | 57.00±0.00 | 57.00±0.00 | **11.03±0.67** |
| Page Blocks | 50.27±8.77 | 43.10±2.15 | 43.10±2.15 | **18.23±0.89** |
| Pima Indians | 37.47±8.19 | **15.67±0.48** | **15.67±0.48** | 20.77±1.96 |
| Vehicle | 147.37±11.09 | 40.07±0.78 | 40.07±0.78 | **18.57±1.33** |
| Yeast | 57.00±8.87 | 57.00±0.00 | 57.00±0.00 | **24.07±1.41** |

Table 4.14: Average number of rules ± standard deviation across 30 trials for public classification datasets. Lower values indicate more compact models.

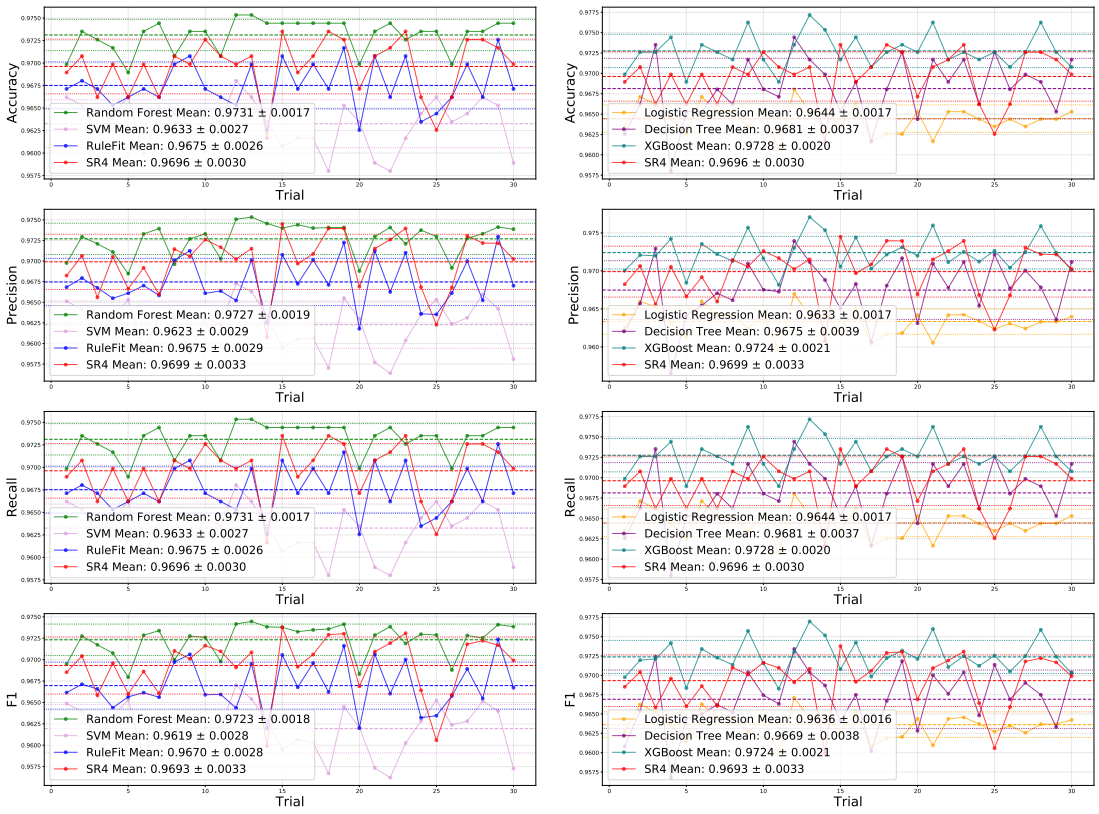



Figure 4.42: Line plot comparison of model performance metrics for page blocks data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit, whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| **Dataset** | **Random Forest** | **RuleFit** | **SR4-Fit** | **Decision Tree** |
|---|---|---|---|---|
| Breast Cancer | **1.99±0.01** | 2.97±0.12 | 2.97±0.12 | 3.42±0.15 |
| E. coli | **1.97±0.01** | 3.55±0.01 | 3.55±0.01 | 3.60±0.08 |
| Page Blocks | **1.97±0.01** | 2.47±0.04 | 2.47±0.04 | 4.41±0.05 |
| Pima Indians | **1.96±0.01** | 2.28±0.07 | 2.28±0.07 | 4.61±0.08 |
| Vehicle | **1.99±0.01** | 2.05±0.04 | 2.05±0.04 | 4.52±0.09 |
| Yeast | **1.96±0.02** | 2.66±0.03 | 2.66±0.03 | 4.69±0.06 |

Table 4.15: Average rule complexity (number of conditions per rule) ± standard deviation across 30 trials for public classification datasets. Bold indicates less complex rules.

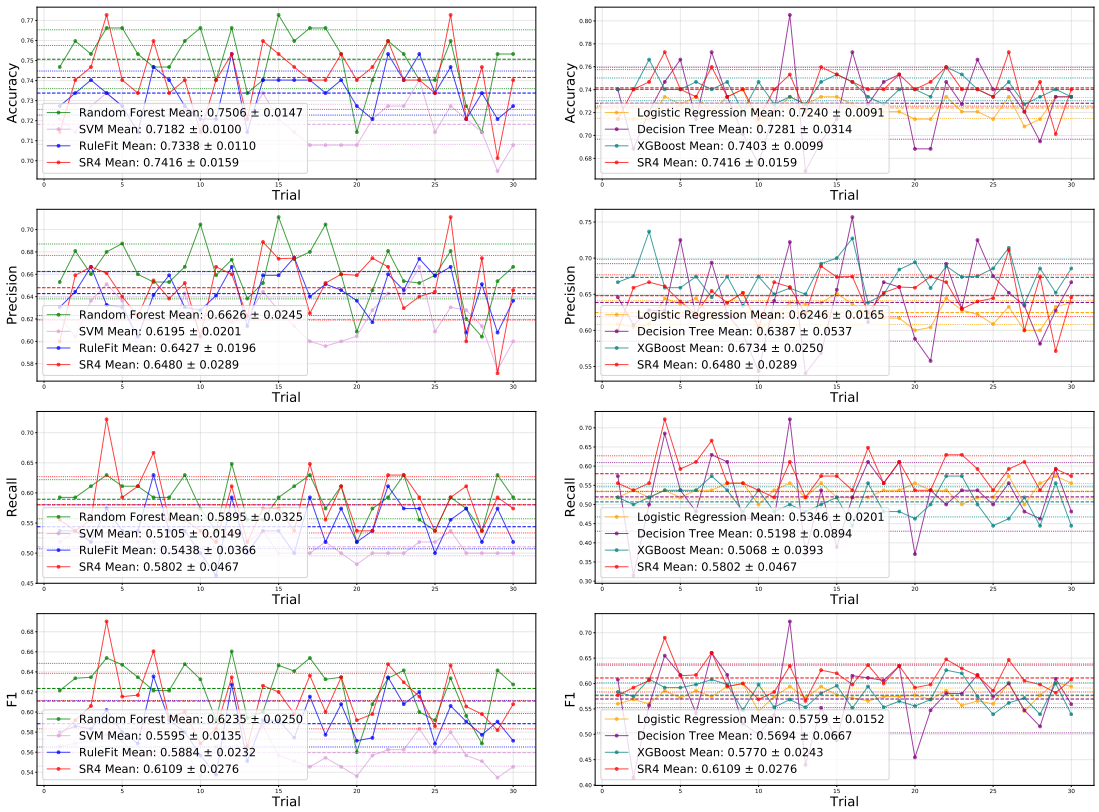



Figure 4.43: Line plot comparison of model performance metrics for Pima Indians data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit, whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| **Dataset** | **Random Forest** | **RuleFit** | **SR4-Fit** | **Decision Tree** |
|---|---|---|---|---|
| Breast Cancer | 0.43±0.14 | 0.47±0.18 | **0.49±0.18** | 0.44±0.20 |
| E. coli | 0.44±0.14 | 0.60±0.11 | **0.61±0.11** | 0.37±0.14 |
| Page Blocks | 0.53±0.12 | 0.48±0.13 | 0.52±0.14 | **0.54±0.13** |
| Pima Indians | **0.46±0.13** | 0.35±0.32 | 0.37±0.31 | 0.36±0.18 |
| Vehicle | 0.49±0.16 | 0.41±0.13 | 0.41±0.12 | **0.53±0.15** |
| Yeast | 0.49±0.14 | 0.63±0.12 | **0.64±0.12** | 0.37±0.15 |

Table 4.16: Average interpretability score (IPS) ± standard deviation across 30 trials for public classification datasets. Bold indicates the highest average IPS per dataset.

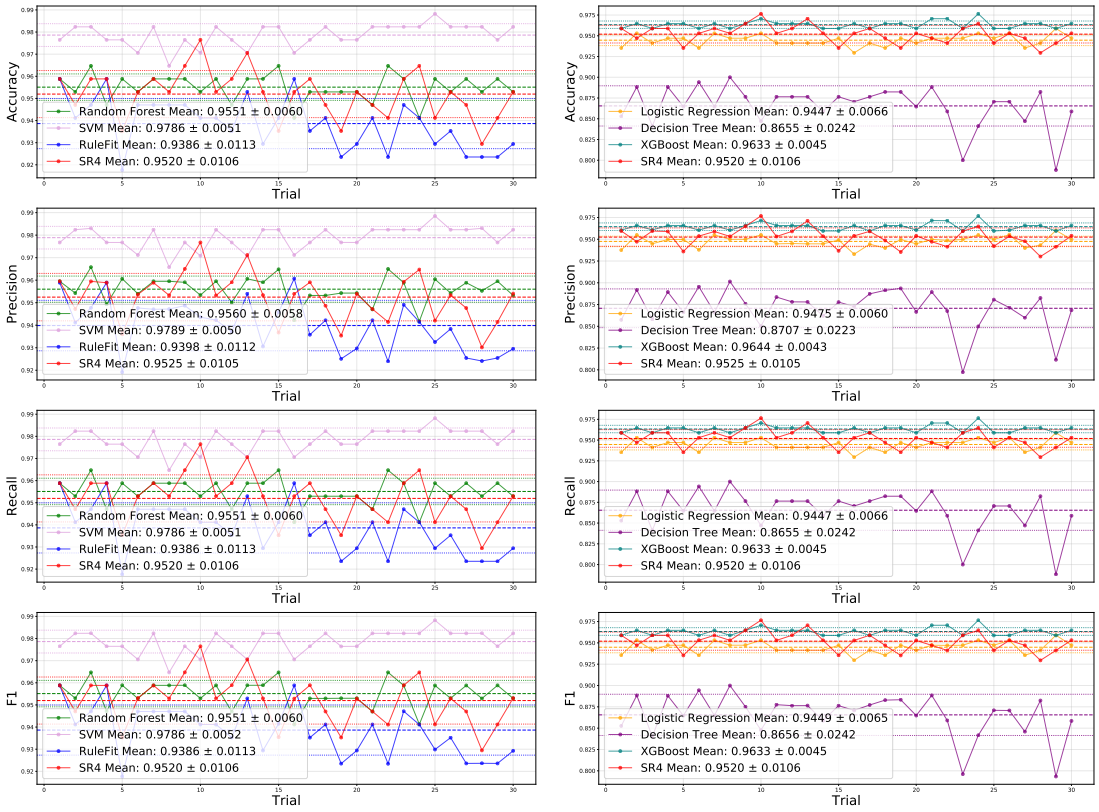



Figure 4.44: Line plot comparison of model performance metrics for vehicle data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit, whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| **Dataset** | **Random Forest** | **RuleFit** | **SR4-Fit** | **Decision Tree** |
|---|---|---|---|---|
| Abalone | 0.01±0.00 | 0.50±0.00 | **0.51±0.01** | 0.25±0.16 |
| Bone | 0.01±0.03 | 0.27±0.00 | **0.37±0.03** | 0.02±0.02 |
| Diabetes | 0.01±0.01 | 0.55±0.00 | **0.58±0.01** | 0.14±0.09 |
| Housing | 0.05±0.02 | **0.61±0.00** | 0.58±0.01 | 0.04±0.06 |
| Machine | 0.01±0.01 | 0.41±0.01 | **0.61±0.01** | 0.01±0.02 |
| MPG | 0.05±0.01 | 0.50±0.00 | 0.37±0.01 | **1.00±0.00** |
| Ozone | 0.01±0.02 | 0.42±0.00 | **0.46±0.01** | 0.05±0.05 |
| Prostate | 0.01±0.01 | 0.50±0.01 | **0.55±0.04** | 0.01±0.01 |

Table 4.17: Average Dice–Sorensen Index ± standard deviation across 30 trials for public regression datasets. Bold indicates the best-performing method per dataset.

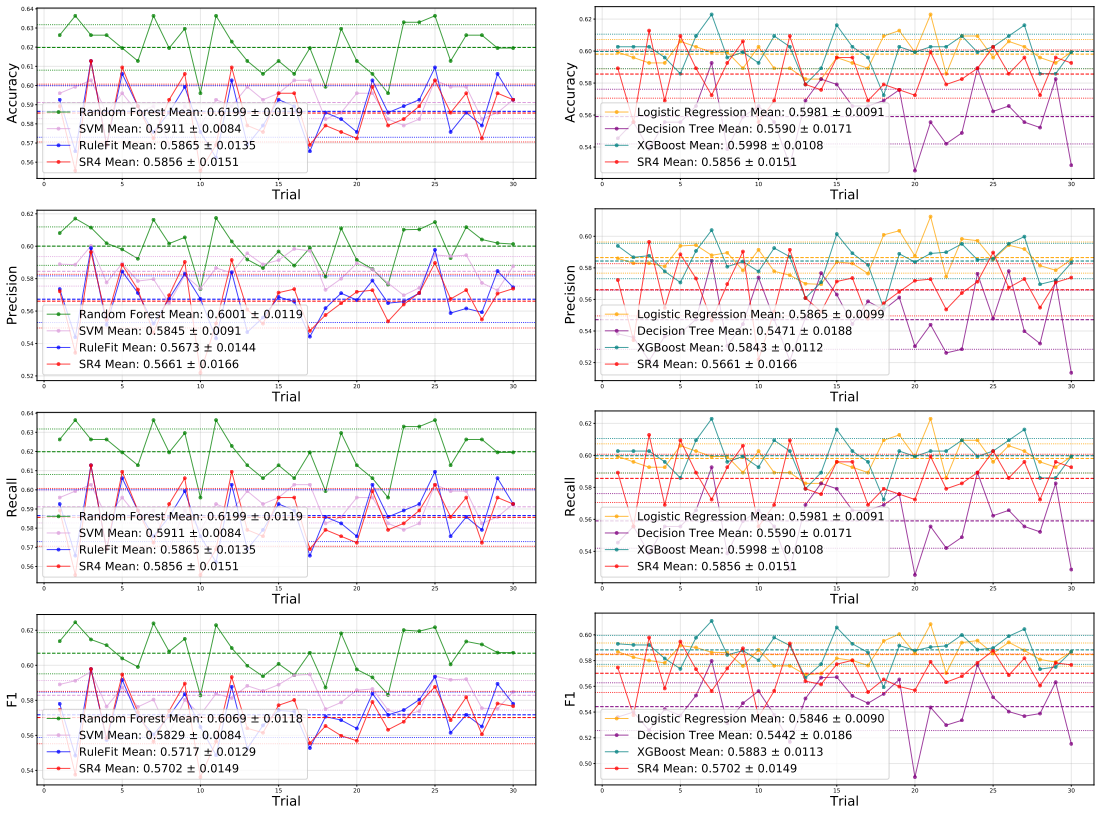



Figure 4.45: Line plot comparison of model performance metrics for yeast data across 30 trials. The left panel reports results obtained for models—random forest, SVM, rulefit, and SR4-fit, whereas the right panel shows results with models logistic regression, decision tree, and XGBoost.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Abalone | 8666.53±101.33 | 16.00±0.00 | 15.96±0.18 | **1.00±0.00** |
| Bone | 8151.33±298.36 | 11.00±0.00 | 2.73±0.44 | **1.03±0.18** |
| Diabetes | 7823.36±209.19 | 18.00±0.00 | 17.00±0.69 | **1.00±0.00** |
| Housing | 7244.06±142.93 | 21.00±0.00 | 21.90±0.95 | **3.06±1.08** |
| Machine | 4166.13±207.00 | 21.80±1.5177 | 14.66±0.84 | **2.66±0.80** |
| MPG | 9090.80±205.65 | 16.00±0.00 | 21.40±1.13 | **1.00±0.00** |
| Ozone | 7968.26±144.58 | 28.00±0.00 | 25.70±1.23 | **2.20±0.48** |
| Prostate | 3220.96±75.75 | 15.90±0.30 | 12.33±1.24 | **3.33±0.88** |

Table 4.18: Average number of rules ± standard deviation across 30 trials for public regression datasets. Lower values indicate more compact models.

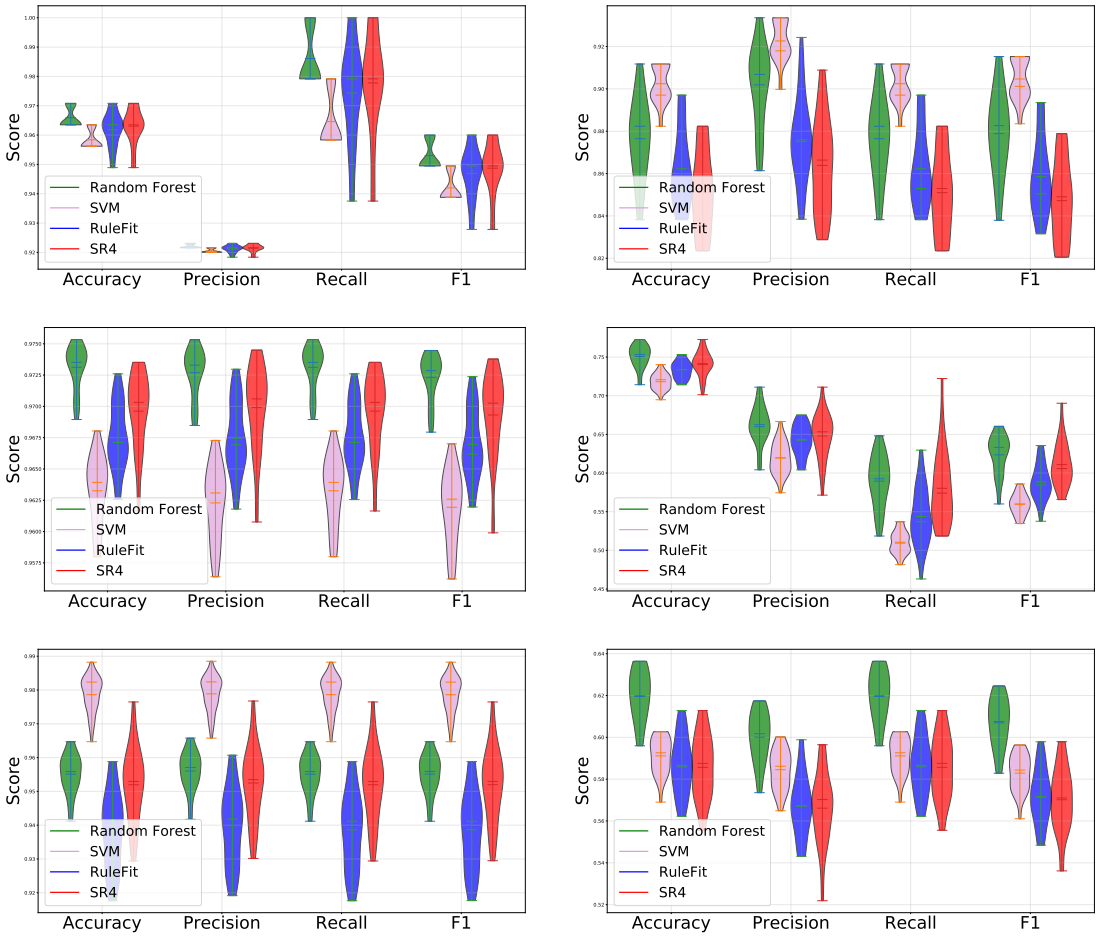



Figure 4.46: Violin plot comparison of model (random forest, SVM, RuleFit, and SR4-fit) performance across multiple standard benchmark classification datasets for different prediction metrics. Top left: Breast cancer; top right: E. coli; middle left: Page blocks; middle right: Pima Indians; bottom left: Vehicle; bottom right: Yeast.

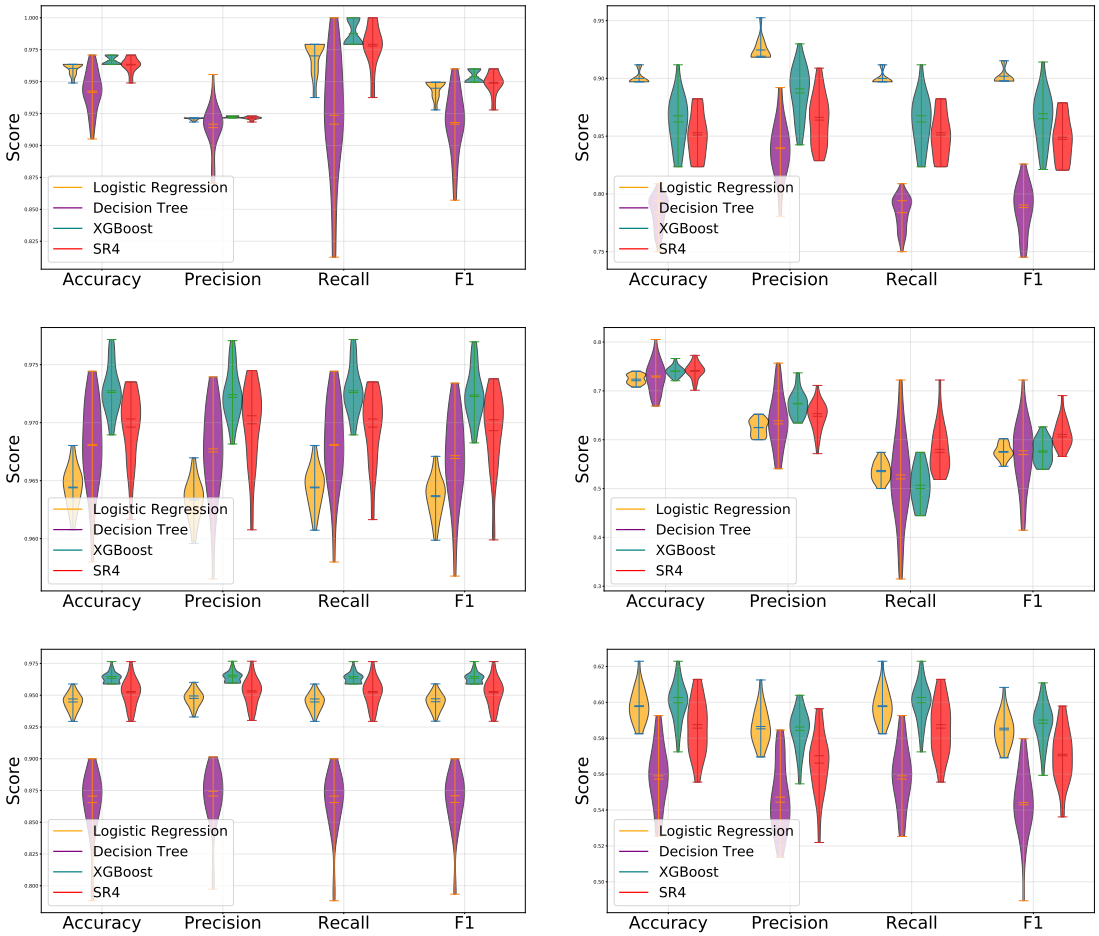



Figure 4.47: Violin plot comparison of model (logistic regression, decision tree, and XGBoost) performance across multiple standard benchmark classification datasets for different prediction metrics. Top left: Breast cancer; top right: E. coli; middle left: Page blocks; middle right: Pima Indians; bottom left: Vehicle; bottom right: Yeast

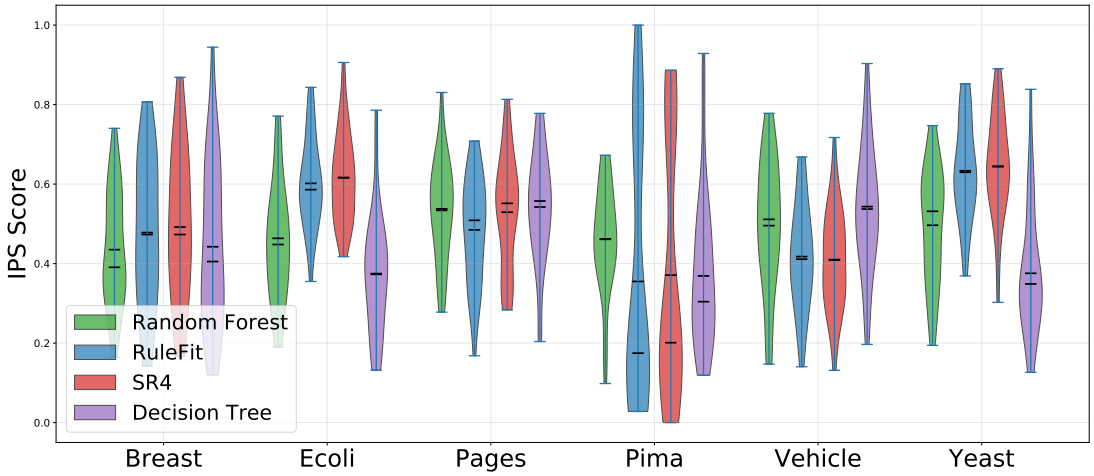



Figure 4.48: Violin plot comparison of interpretability scores for all rule-based models across standard public classification datasets.

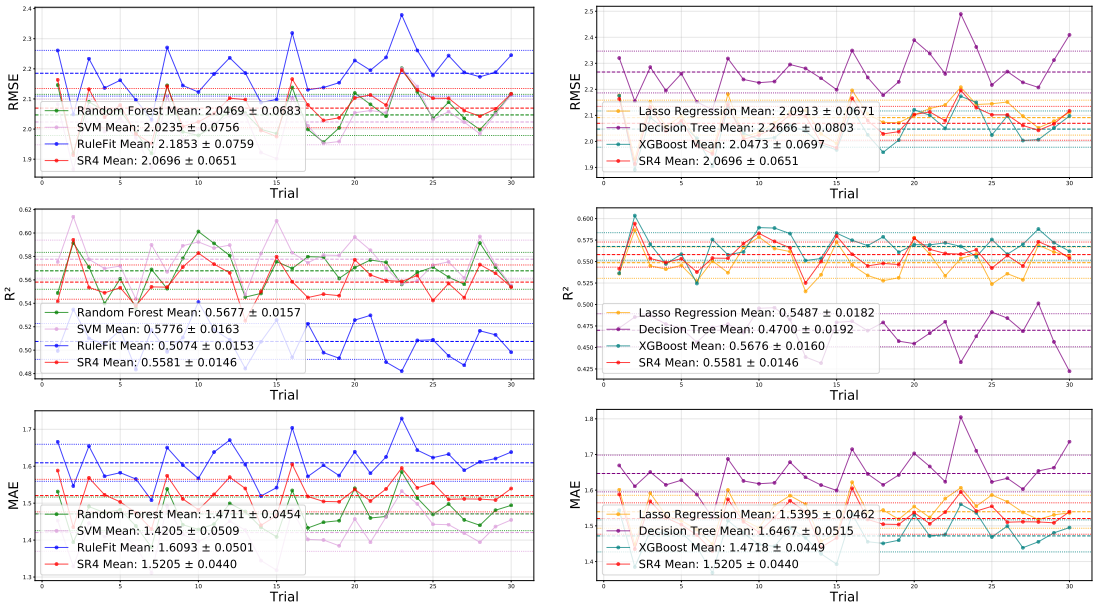



Figure 4.49: Line plot comparison of model performance metrics for abalone data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

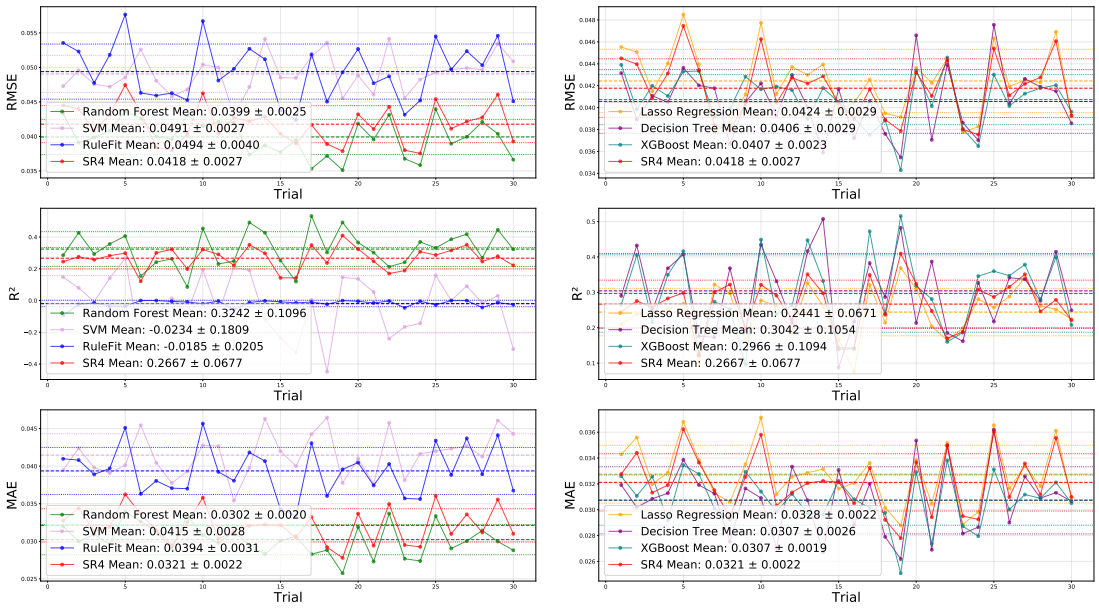



Figure 4.50: Line plot comparison of model performance metrics for bone data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit, whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

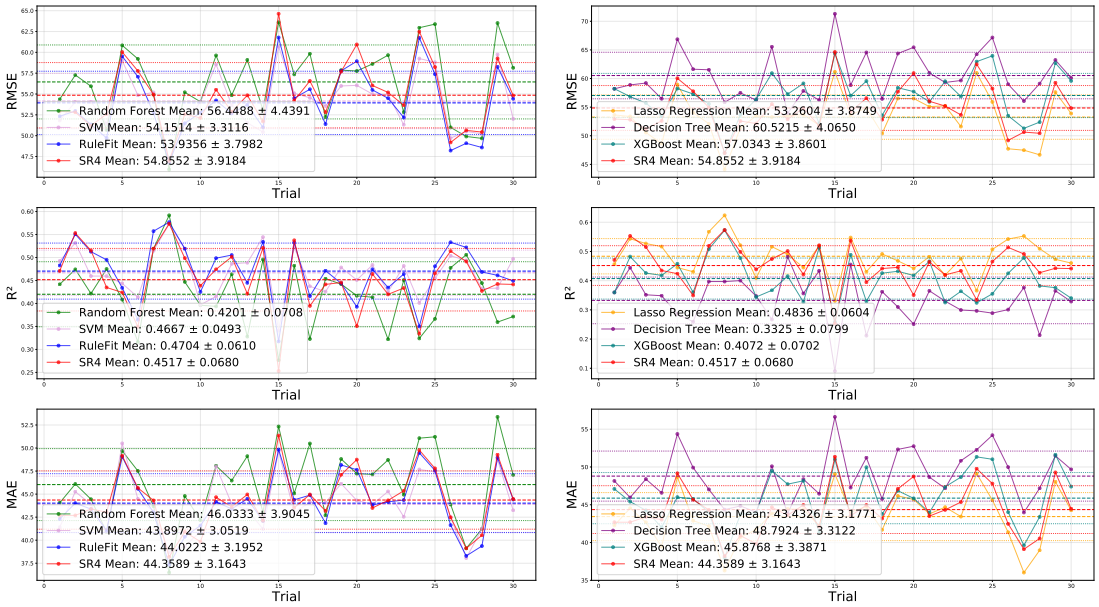



Figure 4.51: Line plot comparison of model performance metrics for diabetes data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

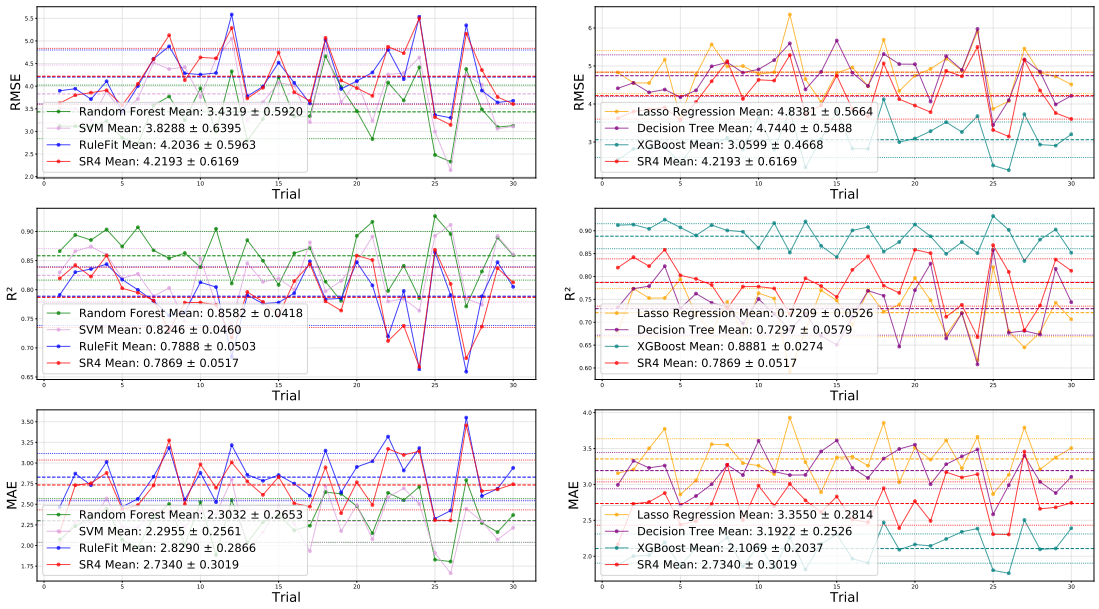



Figure 4.52: Line plot comparison of model performance metrics for housing data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

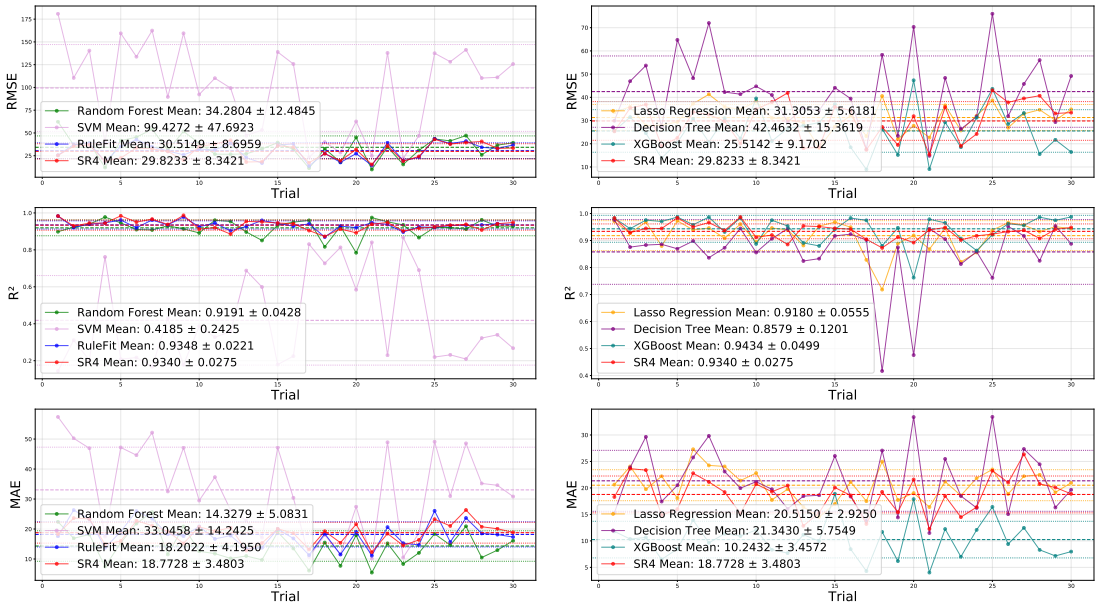



Figure 4.53: Line plot comparison of model performance metrics for machine data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

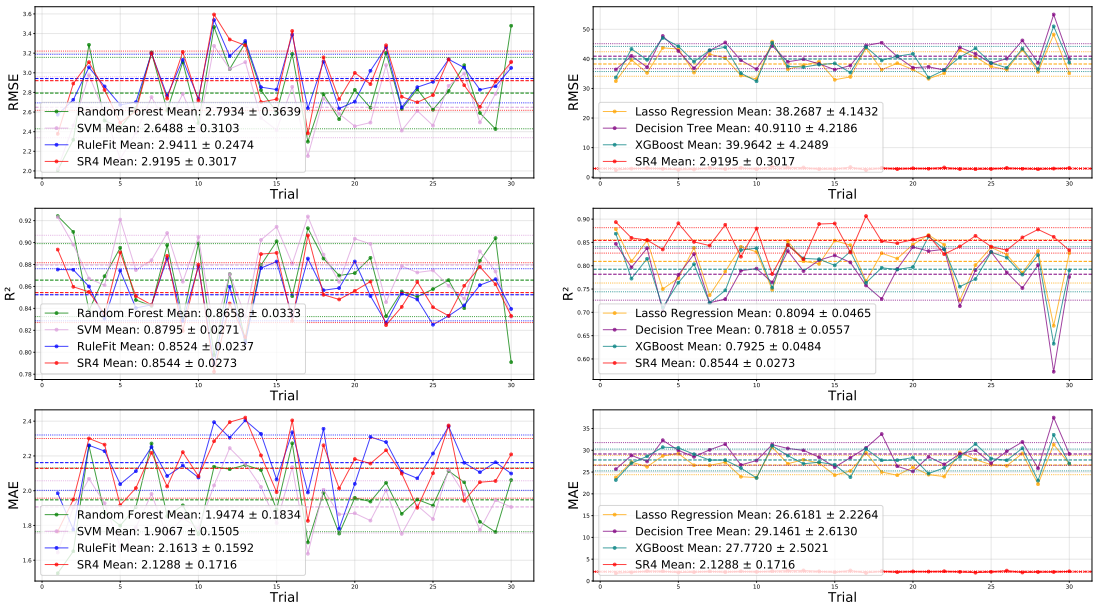



Figure 4.54: Line plot comparison of model performance metrics for MPG data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

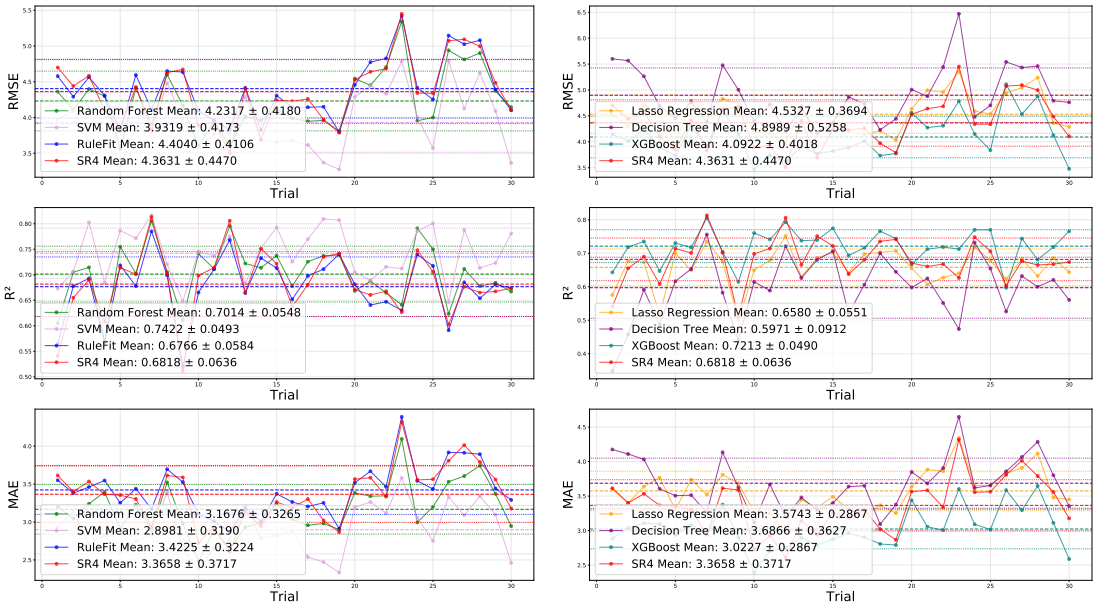



Figure 4.55: Line plot comparison of model performance metrics for ozone data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

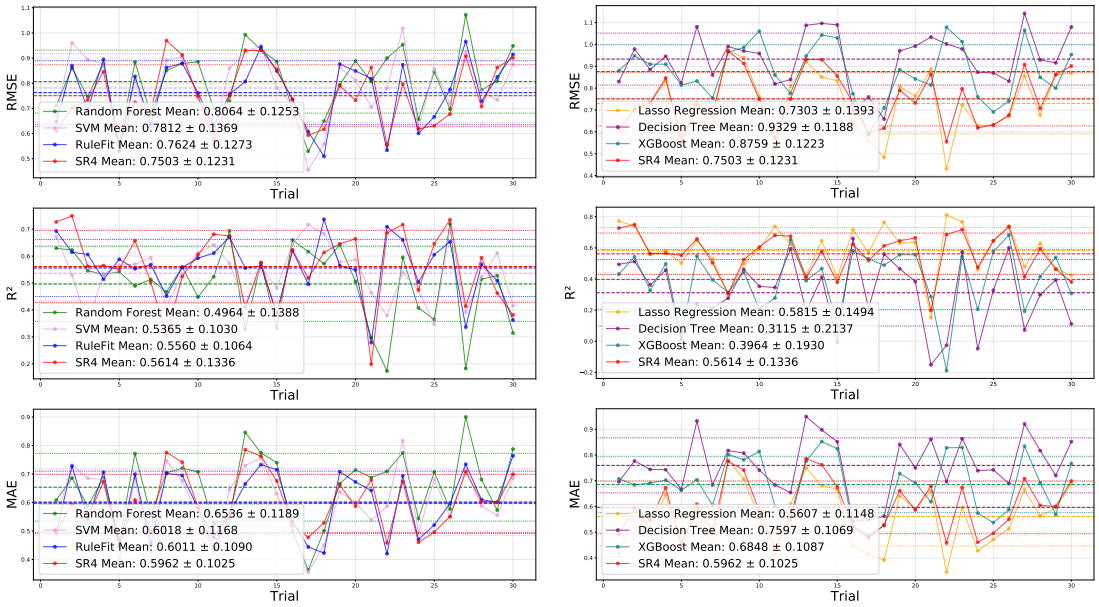



Figure 4.56: Line plot comparison of model performance metrics for prostate data across 30 trials. The left panel reports results obtained for models—random forest, SVM, RuleFit, and SR4-fit , whereas the right panel shows results with models LASSO regression, decision tree, and XGBoost.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Abalone | 6.54±0.02 | 2.00±0.00 | 1.99±0.01 | **1.00±0.00** |
| Bone | 5.95±0.03 | 2.45±0.00 | 2.24±0.15 | **1.73±0.98** |
| Diabetes | 5.88±0.02 | 1.88±0.00 | 1.82±0.05 | **1.00±0.00** |
| Housing | 5.89±0.02 | **1.76±0.00** | 2.22±0.07 | 3.00±0.00 |
| Machine | 5.62±0.05 | 2.61±0.14 | **1.75±0.07** | 2.72±0.58 |
| MPG | 5.98±0.02 | 2.00±0.00 | 2.87±0.06 | **1.00±0.00** |
| Ozone | 5.86±0.01 | 2.71±0.00 | **2.59±0.07** | 2.98±0.09 |
| Prostate | 4.75±0.05 | 1.98±0.04 | **1.79±0.14** | 2.39±0.31 |

Table 4.19: Average rule complexity ± standard deviation across 30 trials for public regression datasets. Bold indicates less complex rules

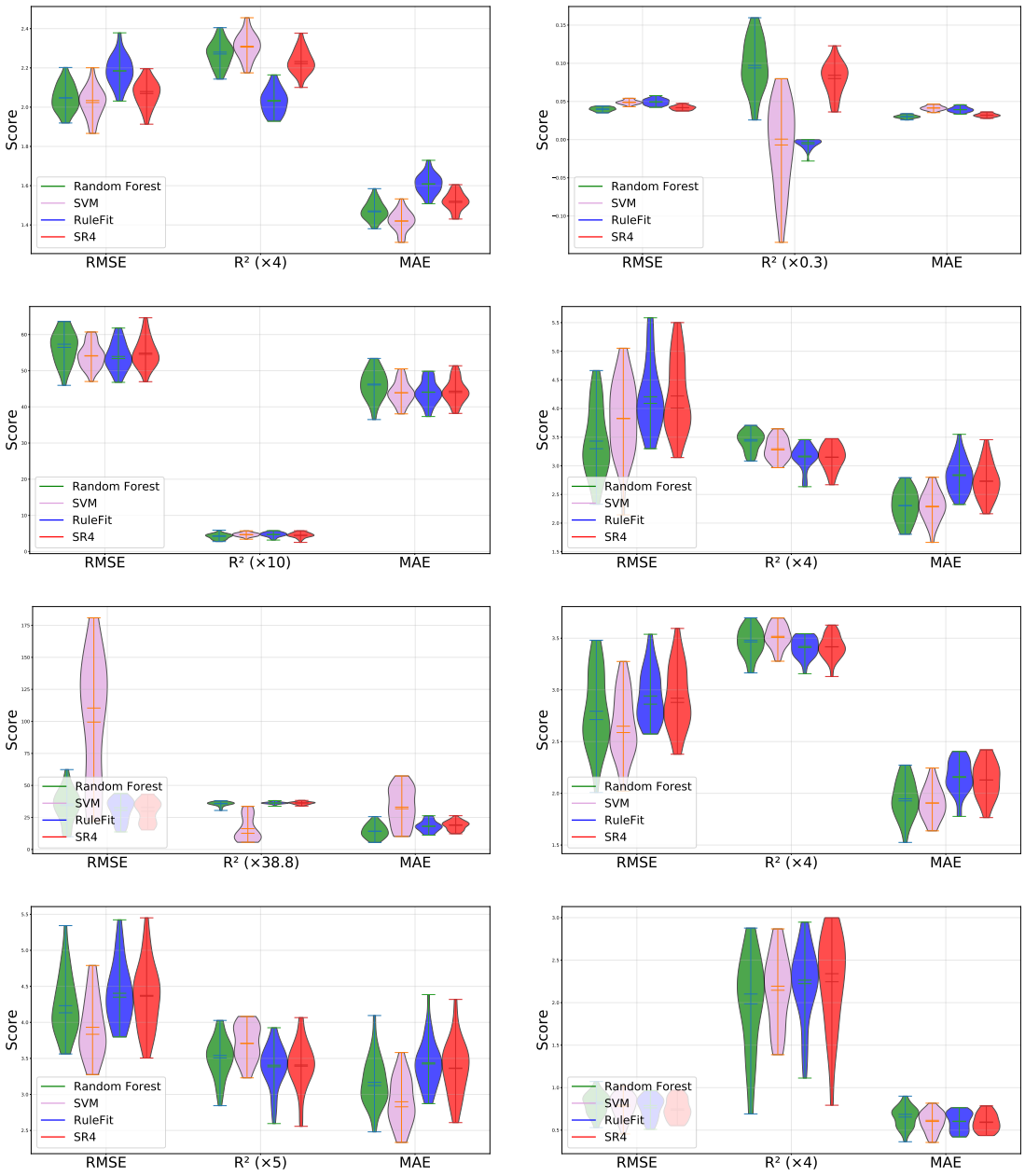



Figure 4.57: Violin plot comparison of models (random forest, SVM, RuleFit, and SR4-fit) performance across multiple public regression datasets for different prediction metrics. Top left: Abalone; top right: Bone; second row left: Diabetes; second row right: Housing; third row left: Machine; third row right: MPG; bottom left: Ozone; bottom right: Prostate.

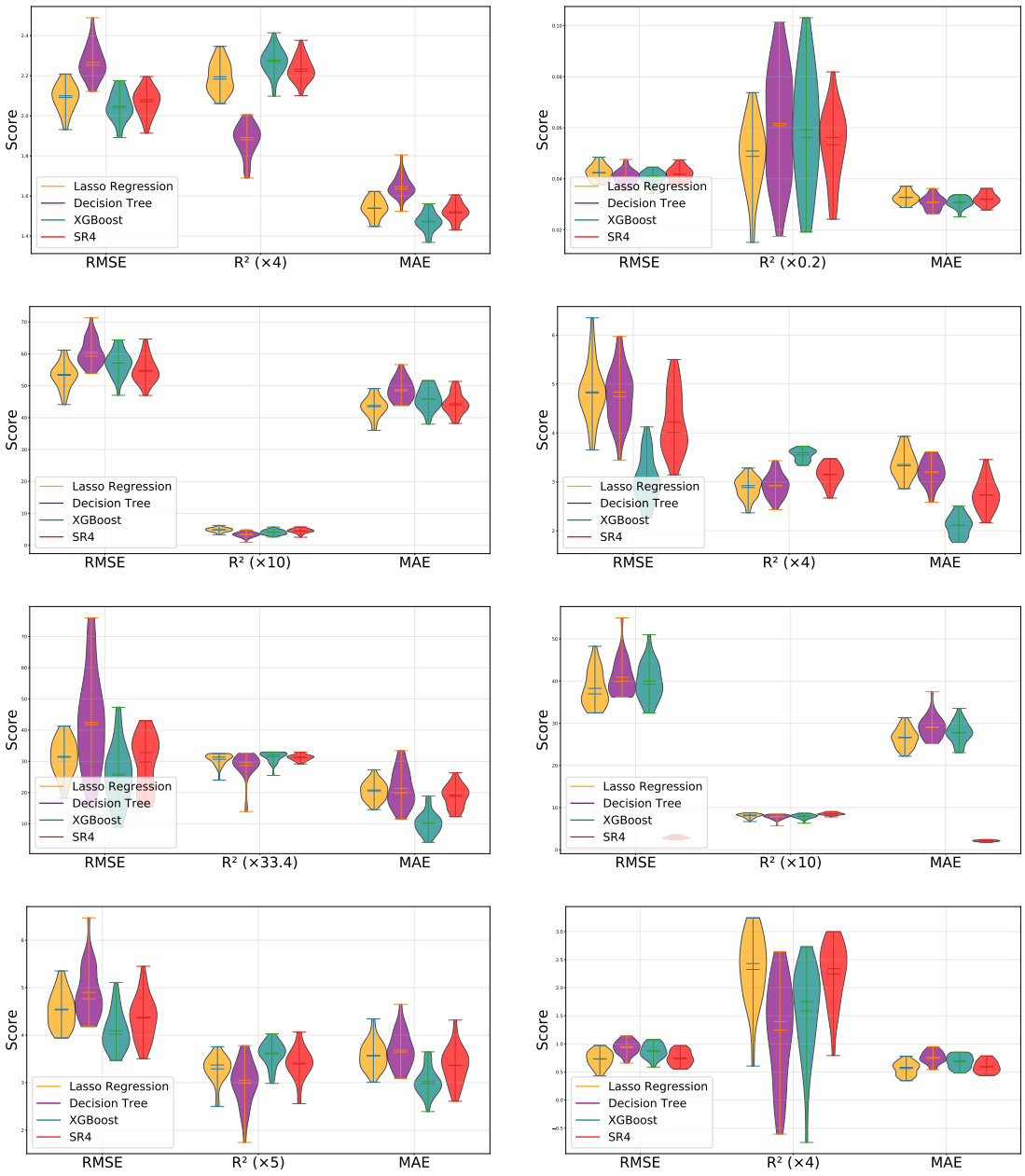



Figure 4.58: Violin plot comparison of models (LASSO regression, decision tree, and XGBoost) performance across multiple public regression datasets for different prediction metrics. Top left: Abalone; top right: Bone; second row left: Diabetes; second row right: Housing; third row left: Machine; third row right: MPG; bottom left: Ozone; bottom right: Prostate.

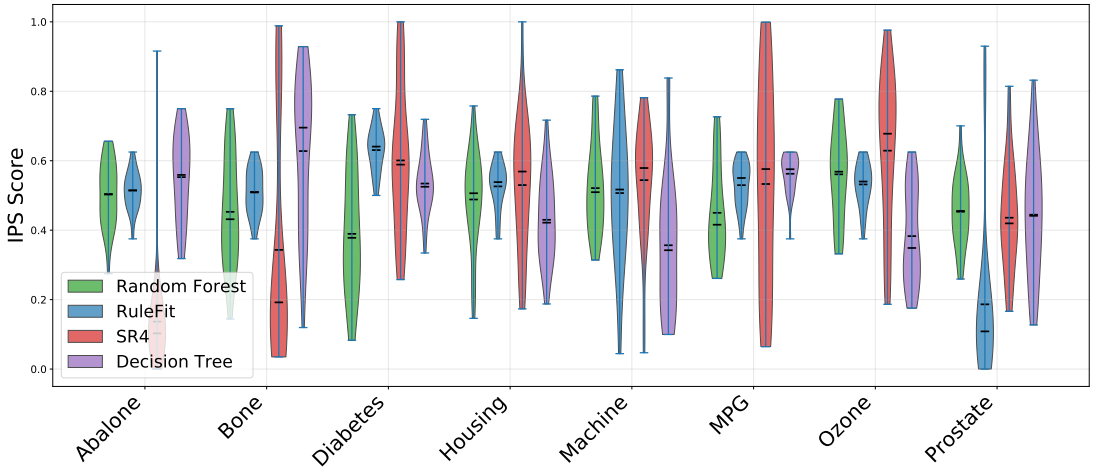



Figure 4.59: Violin plot comparison of interpretability scores for all rule-based models across standard public regression datasets.

| Dataset | Random Forest | RuleFit | SR4-Fit | Decision Tree |
|---|---|---|---|---|
| Abalone | 0.50±0.09 | 0.51±0.05 | 0.13±0.15 | **0.55±0.12** |
| Bone | 0.45±0.15 | 0.51±0.06 | 0.34±0.33 | **0.62±0.21** |
| Diabetes | 0.38±0.17 | **0.63±0.06** | 0.60±0.20 | 0.53±0.08 |
| Housing | 0.48±0.13 | 0.52±0.06 | **0.53±0.18** | 0.42±0.12 |
| Machine | 0.52±0.12 | 0.50±0.19 | **0.54±0.16** | 0.35±0.19 |
| MPG | 0.45±0.13 | 0.52±0.06 | 0.53±0.28 | **0.56±0.05** |
| Ozone | 0.56±0.13 | 0.53±0.06 | **0.62±0.21** | 0.38±0.13 |
| Prostate | **0.45±0.09** | 0.18±0.23 | 0.43±0.15 | 0.44±0.17 |

Table 4.20: Average interpretability score (IPS) ± standard deviation across 30 trials for public regression datasets. Bold indicates the highest average IPS per dataset.

# Chapter 5

# Discussion

The following analysis presents representative rules extracted from each dataset and interprets the associated demographic and structural characteristics of the regions these rules classify. In addition to classification-oriented rules, the analysis also examines rules derived from regression-based demographic modeling, where SR4-Fit identifies structured combinations of predictors associated with continuous outcomes. Furthermore, representative rules from publicly available datasets are analyzed for both classification tasks (binary and multi-class) and regression settings to demonstrate the general applicability of the proposed approach.

## 5.1 Including party percentages—Classification

In the minimum dataset, SR4-Fit builds a concise, interpretable model primarily driven by the Democratic percentage and Republican percentage. The most consistently important rule indicates that districts where the Democratic percentage is greater than about 48.7% while the Republican percentage is at or below about 45.5% are likely to elect Democrats. This pattern is not surprising,

since districts where votes for Democratic candidates previously outnumbered votes for Republican candidates are likely to elect Democratic candidates in the future.

Similarly, districts where the Democratic percentage is at or below about 48.7% and the Republican percentage is greater than about 45.5% typically signal Republican elections. At first glance this may seem surprising because it could include districts in which Democrats slightly outnumber Republicans. However, several explanations are possible: Republican turnout may be higher than Democratic turnout, independent voters may favor Republican candidates, or some Democrats may cross over to vote for Republican candidates in those districts and years.

Interestingly, this latter pattern occasionally received negative coefficients in several trials, meaning it predicted Democratic outcomes in some cases. In those instances, the districts had a median age of 39.24 years compared to the overall average of 37.49 years, a White non-Hispanic population share of 69.46% compared to 63.64%, and a bachelor's degree attainment rate of 19.06% compared to 17.91%. In other words, although districts without a clear Democratic majority often elect Republicans, districts that are older, whiter, and more highly educated sometimes elect Democrats. These demographic characteristics are widely known to influence voting patterns.

In the standard dataset, SR4-Fit builds an interpretable model shaped largely by age distribution, party support levels, and socioeconomic indicators. Age between 45 and 54 emerges as the most influential feature and consistently receives high positive coefficients, indicating Republican-leaning tendencies in regions where middle-aged populations are prominent. Similarly, the age groups 65–74 and 18–24 also show strong positive associations with Republican classification,

highlighting the predictive importance of both older and younger demographic groups.

A more complex rule involving closely balanced Democratic and Republican percentages combined with relatively low median income (around $23,717 or lower) has only a small positive influence, suggesting that lower income levels may slightly shift predictions toward Republican classification in closely contested regions. Another narrow-range rule involving modest Democratic support, moderate Republican support, and bachelor's degree attainment above about 21.7% also receives a small positive coefficient. This suggests that relatively high educational attainment may sometimes push moderately divided districts toward Republican classification.

In the expanded dataset, SR4-Fit captures more nuanced voting behavior through interpretable patterns primarily driven by Democratic and Republican support percentages along with demographic factors. One pattern involving moderately split Democratic and Republican support tends to produce Democratic classifications. Districts satisfying this pattern tend to have a slightly older population, with a median age of about 38.97 years compared to the overall average of 37.49 years. These districts also show higher Asian population shares (8.48% versus 4.86%) and higher multiracial population shares (3.95% versus 2.68%). In addition, they have somewhat higher White non-Hispanic population shares, higher bachelor's degree attainment rates (19.38% compared to 17.91%), higher median incomes (about $30,764 compared to $26,873), and lower poverty levels (11.22% compared to 14.63%). Together, these characteristics suggest that moderate diversity combined with higher socioeconomic status tends to correspond with Democratic outcomes in these districts.

In the previous dataset, SR4-Fit identifies patterns where both Democratic

and Republican support levels are relatively high but Republican support is slightly stronger, resulting in Republican classification. Districts following this pattern tend to have slightly older populations, with a median age of about 37.70 years compared to the overall average of 37.49 years. These districts are also less diverse, with higher White non-Hispanic population shares (71.58% versus 63.64%) and lower representation of Black, Asian, Hispanic, and multiracial populations. Educational attainment is also somewhat higher in these districts, with about 19.37% holding bachelor's degrees and 11.68% holding graduate or professional degrees compared to 17.91% and 10.65% overall.

## 5.2 Without party percentages—Classification

For the minimum dataset, several demographic patterns emerge that help explain the predicted electoral outcomes. Districts with White population between approximately 26.8% and 42.25% tend to predict Democratic outcomes, suggesting that districts with greater racial diversity are more likely to support Democratic candidates. Similarly, districts with moderate White population levels between approximately 41.55% and 54.65%, combined with bachelor's degree attainment above 12.95%, also tend to predict Democratic outcomes, indicating that moderate educational attainment, together with demographic diversity, may contribute to Democratic support. In contrast, districts characterized by very high White population shares exceeding approximately 77.15%, male population proportions above about 48.35%, and bachelor's degree attainment greater than approximately 17.35% are more likely to predict Republican outcomes. Age structure also appears to influence predictions: districts with lower White population shares ($\leq 42.7$) combined with older median ages above about 41.7 years, as well as dis-

tricts with lower White population shares ($\leq$ 41.55%) and younger median ages below approximately 40.05 years, tend to predict Republican outcomes. Additionally, temporal factors appear to play a role, as districts with White population shares greater than about 53.85%, male population proportions above roughly 48.55%, and median ages above approximately 35.55 years were associated with Democratic outcomes.

Across the extracted rules in the standard dataset, districts with lower White population shares (generally $\leq$42.7%) combined with moderate working-age populations aged 25–44 (approximately 23.65%–24.55%) and younger median ages ($\leq$41.7 years) frequently predict Republican outcomes, indicating that districts with younger working-age demographics may lean Republican in the rule set. Several rules also associate Republican predictions with districts where middle-aged populations (Age 45–54: $\geq$12.65%) or pre-retirement populations (Age 55–64: $\geq$10.45%) are present, while older populations such as Age 65–74 remain relatively small ($\leq$8–9%), suggesting that certain age structures influence voting behavior. In addition, Republican predictions appear in districts with low minority population shares, including Black population $\leq$1.95% and Asian population $\leq$0.85%, as well as moderate Hispanic population shares $\leq$38.2%. Some rules further associate Republican outcomes with family-structured populations containing both youth (Age 5–17 $\geq$13.45%) and elderly populations (Age 75+ $\geq$8.2%), suggesting the presence of multi-generational demographic structures. In contrast, Democratic predictions are observed in districts with lower White population shares ($\leq$42.25%) combined with higher median income levels ($\geq$ \$30,298.5), indicating that economically stronger and demographically diverse districts may favor Democratic candidates. Additional Democratic outcomes appear in districts with higher proportions of young adults (Age 18–24 $\geq$11%) and

moderate educational attainment levels, such as some college education exceeding 27.45% and high school graduation rates above 30.3%, suggesting that education and younger populations may contribute to Democratic voting patterns.

Across the rules extracted from the expanded dataset, many rules involve districts where the proportion of individuals who have never been married is $\leq$32.95% and the proportion of married individuals is $\leq$54.05–54.45%, suggesting that marital status composition is a common structural feature in the model. Democratic predictions frequently occur in districts with greater demographic diversity, including Black population shares between approximately 20.75% and 29.0% or above 4.85%, Hispanic population shares greater than about 1.85%, and Asian population shares above roughly 1.45%, combined with higher educational attainment such as graduate or professional degree levels exceeding about 4.85–10.3% and median income levels above roughly \$19,394. These districts may also contain younger adult populations (Age 18–24 above approximately 8.65–11.7%), indicating the influence of demographic diversity, education, and younger age groups in Democratic predictions. In contrast, Republican predictions appear in districts with moderate educational attainment, including bachelor's degree levels above about 10.9% and relatively low proportions of individuals without a high school diploma ($\leq$12.15%), along with lower Black population shares ($\leq$8.9%), moderate language diversity where households speaking languages other than English remain $\leq$34.15%, and moderate income groups in the \$65,000–\$74,999 range exceeding about 2.25%.

Across the extracted rules in the previous dataset, Republican predictions frequently occur in districts with moderate educational attainment, such as populations with some college or associate degrees exceeding approximately 31.65%, combined with low Asian population shares ($\leq$1.15%) and moderate represen-

tation of other racial groups ($\geq$1.95%). Additional Republican outcomes appear in districts with large working-age populations (Age 25–44 above roughly 24.45–24.55%) and higher White population shares exceeding approximately 54.75%, particularly when family-age populations are present, such as youth populations (Age 5–17 $\geq$18.15%). Republican predictions also appear in districts with moderate racial diversity, including Asian population shares greater than about 2.05% and multiracial populations above approximately 0.95%, as well as districts where Black population shares remain below roughly 38.45%. In contrast, Democratic predictions tend to occur in districts with very high Hispanic population shares ($\geq$67.2%), as well as districts where lower educational attainment levels are present, such as less than high school completion exceeding approximately 19.15%, combined with greater representation of other racial groups ($\geq$2.30%).

## 5.3 Including REP percentage and DEM percentage as label—Regression

In the minimum dataset, the rules primarily highlight the strong influence of Republican vote share (REP percentage) on predicting Democratic vote percentage. When Republican support is extremely low (REP percentage$\leq -2.0186$), the model increases the predicted Democratic percentage by 23.72, indicating that districts with very weak Republican presence strongly favor Democratic outcomes. When Republican support remains low but not extremely low ($-2.0186<$REP Percentage$\leq -0.7020$) and the share of the white non-Hispanic population is relatively smaller ($\leq -0.4248$), the prediction increases by 18.12, suggesting that demographic composition further reinforces Democratic vote share in these districts.

Additionally, in districts with very low Republican support (REP percentage$\leq -2.0338$) and moderate levels of bachelor's degree attainment ($\leq 0.2639$) increase the predicted Democratic percentage by 7.56.

In the standard dataset, the rules again emphasize the strong role of Republican vote share (REP percentage) in predicting Democratic vote percentage. When Republican support is extremely low (REP percentage$\leq -2.0308$), the model increases the predicted Democratic vote share by 16.96, indicating that districts with very weak Republican presence strongly favor Democratic outcomes. Additionally, when Republican support is low (REP percentage$\leq -0.7970$) and a relatively high Black population share (Black$>1.6788$), the predicted Democratic percentage increases by 15.74.

In the expanded dataset, when Republican support is relatively low (REP percentage$\leq -0.8104$) and the share of the Black or African American population is high ($>1.5405$), the model increases the predicted Democratic vote percentage by 17.12. This suggests that districts with weaker Republican presence and a higher proportion of Black residents tend to exhibit substantially stronger Democratic electoral support.

In the previous dataset, the rules highlight the influence of Republican vote share, demographic characteristics, and historical voting patterns in predicting Democratic vote percentage. When Republican support falls within a moderately low range ($-0.7020<$REP percentage$\leq -0.3607$) and the share of the population aged 25–44 years is relatively higher ($>-0.4926$), the predicted Democratic percentage increases by 9.19, suggesting that districts with larger working-age populations tend to favor Democratic outcomes when Republican support is moderate. When Republican vote share remains moderately low ($-0.8065<$REP percentage$\leq -0.1539$) but the 25–44 age population is smaller, combined with

higher proportions of residents with less than a high school education ($>-0.4735$), the prediction increases by 6.84. Finally, when Republican support is very low (REP percentage$\leq -0.8128$), the model increases the predicted Democratic vote percentage by 3.91, even when the Black population share remains below the specified threshold.

## 5.4 Without REP percentage and DEM percentage as label—Regression

In the minimum dataset without including Republican vote share as a feature, the rules highlight the influence of racial composition, education, and demographic characteristics in predicting Democratic vote percentage. When the share of the White non-Hispanic population is relatively low ($\leq -1.0747$), the predicted Democratic percentage increases by 17.82. A similar pattern appears when the White non-Hispanic population share remains low ($\leq -0.7819$) combined with specific ranges of male population share, increasing the prediction by 18.73, indicating strong Democratic support in districts with a lower White population share. Additionally, districts with higher educational attainment ($0.4004<$Bachelor's degree$\leq 1.3966$) and higher White population share ($>0.5837$) still increase the predicted Democratic percentage by 14.04, suggesting that education plays an important role in shaping voting behavior.

In the standard dataset without including Republican vote share as a feature, the rules emphasize the importance of racial composition, educational attainment, and age distribution in predicting Democratic vote percentage. In districts with low White population share (White$\leq -1.0973$) and very high Black popu-

lation share (Black>2.9733) produce the strongest increase in Democratic vote share, contributing 19.53 to the prediction. Similarly, districts with higher Asian population share (>0.0735) and lower proportions of younger residents increase the predicted Democratic percentage by 18.08. Other rules highlight the role of age distribution, particularly larger shares of the 25–44 age group (>0.6508), which increases predictions by 15.06 to 15.21 when combined with lower White population share and specific education levels. Additionally, districts with higher educational attainment (graduate degree>0.0250) or higher proportions of high school graduates (>0.8662) also contribute positively to Democratic vote predictions.

In the expanded dataset without including Republican vote share as a feature, the rules highlight the influence of marital status, racial composition, education, income, and age distribution in predicting Democratic vote percentage. Districts with lower shares of the White non-Hispanic population and higher shares of Black residents show strong increases in Democratic vote share, contributing 17.14 to the prediction. Similarly, districts characterized by high proportions of high school graduates combined with lower median income produce the strongest effect, increasing predicted Democratic vote share by 27.95. Other rules highlight the importance of marital status, particularly districts with higher shares of individuals who have never married, which increase Democratic vote predictions by 13.17 to 15.14 when combined with specific age distributions and demographic characteristics. Additionally, racial diversity and language composition further refine predictions, contributing 12.27 to 13.49 to the predicted Democratic percentage.

In the previous dataset without including Republican vote share as a feature, when the previous party indicator reflects Democratic alignment and the share of

the White non-Hispanic population is relatively higher ($>-0.7791$), the predicted Democratic vote percentage increases by 17.10. Similarly, when previous electoral alignment remains Democratic and the White non-Hispanic population share is relatively lower ($\leq-0.7882$), the model increases the predicted Democratic percentage by 14.84.

## 5.5 Including DEM percentage and REP percentage as label—Regression

In the minimum dataset where Republican vote percentage is the target variable and Democratic vote percentage is included as a feature, the rule highlights the inverse relationship between the two party vote shares. Specifically, when Democratic vote share is relatively low ($-2.1595<$DEM percentage$\leq-0.5436$), the model increases the predicted Republican vote percentage by 6.47. This indicates that districts with weaker Democratic support tend to exhibit higher Republican vote percentages, reflecting the expected competitive relationship between the two major parties in electoral outcomes.

In the standard dataset where Republican vote percentage is the label and Democratic vote percentage is included as a feature, When Democratic vote share is extremely low (DEM percentage$\leq-2.1753$), the predicted Republican vote percentage increases substantially by up to 15.11 units, representing the strongest contribution among the rules. When Democratic support remains relatively low ($-2.1629<$DEM percentage$\leq-0.5604$), the predicted Republican vote share increases by 8.00 to 10.16, particularly in districts with higher White population share or lower educational attainment. Additional rules show that moderately low

Democratic vote share ($-0.5604<$DEM percentage$\leq -0.1930$) contributes smaller increases ranging from 5.43 to 5.73, depending on demographic and socioeconomic characteristics such as lower Hispanic population share and lower Black population share and higher median income.

In the expanded dataset where Republican vote percentage is the label and Democratic vote percentage is included as a feature, when Democratic vote share is extremely low (DEM percentage$\leq -2.1629$), the predicted Republican vote percentage increases substantially by 16.18, particularly in districts where the income group earning \$35,000–\$49,999 remains below the specified threshold. When Democratic support remains relatively low ($-2.1629<$DEM percentage$\leq -0.5604$) and the share of married individuals is relatively higher, the model increases the predicted Republican vote share by 9.95. Finally, when Democratic vote share falls within a moderate low range ($-0.5604<$DEM percentage$\leq -0.2231$) and the share of individuals who have never married remains below the specified threshold, the predicted Republican percentage increases by 6.24.

In the previous dataset where Republican vote percentage is the label and Democratic vote percentage is included as a feature, when Democratic vote share is relatively low ($-2.1323<$DEM percentage$\leq -0.5254$), the predicted Republican vote percentage increases significantly by up to 12.92, especially in districts with lower Black population share. Similarly, districts with low Democratic support ($-2.1851<$DEM percentage$\leq -0.4622$) and specific educational characteristics contribute 11.97 to Republican vote share. When Democratic vote share falls within moderate low ranges (approximately $-0.57$ to $-0.21$), the predicted Republican percentage increases more modestly by 6.05 to 6.85, depending on demographic factors such as higher White population share, lower Hispanic population share, and racial composition.

## 5.6 Without DEM percentage and REP percentage as label—Regression

In the minimum dataset where Republican vote percentage is the target variable and Democratic vote share is not included as a feature, the rules highlight the importance of demographic composition, education, age structure, and geographic context in explaining Republican vote share. Districts with higher shares of White non-Hispanic population ($>-0.9646$) and a higher male population share tends to increase predicted Republican vote percentages. The strongest contribution occurs when the proportion of individuals with bachelor's degrees is relatively low ($\leq-0.3805$) and the median population age is lower ($\leq-0.1370$), increasing the predicted Republican vote share by 20.05. Other rules show that districts with higher educational attainment (>0.2185 bachelor's degree share) contribute 12.90, while similar demographic conditions contribute 7.72.

In the standard dataset where Republican vote percentage is the label and Democratic vote share is not included as a feature, the rules highlight several demographic factors associated with higher Republican vote share. Districts with higher White population share ($>-0.9312$) and lower proportions of graduate-level educational attainment ($\leq$0.3989) increase the predicted Republican vote percentage by 9.21. Additionally, individual demographic features such as White population share (+10.20) and population aged 65–74 (+11.57) contribute strongly to Republican vote predictions, suggesting that districts with larger White and older populations tend to exhibit higher Republican vote percentages. Educational attainment and ethnic composition also contribute smaller but meaningful increases, with bachelor's degree share adding 6.47 and Hispanic

or Latino population share contributing 5.41.

In the expanded dataset where Republican vote percentage is the label and Democratic vote share is not included as a feature, the rules highlight the influence of marital status, age distribution, education, income, and economic status on Republican vote share. Districts with lower proportions of individuals who have never married ($\leq -0.2858$) and higher shares of individuals with less than a high school education ($> -0.3833$) increase the predicted Republican vote percentage by 9.69, particularly during later election periods. Additional rules show that moderate levels of never-married individuals combined with higher median age groups such as 45–54 contribute 7.64 to Republican vote predictions. Demographic features such as population aged 65–74 (+4.24) and bachelor's degree attainment (+4.45) also contribute positively, while Hispanic or Latino population share adds 9.04 to the prediction. Socioeconomic characteristics including median income levels and poverty thresholds further influence predictions, contributing 4.74 to 6.80 depending on the combination of conditions.

In the previous political dataset where Republican vote percentage is the label and Democratic vote share is not included as a feature, the rules highlight the importance of historical electoral alignment, demographic composition, age distribution, and educational attainment in explaining Republican vote share. Districts where the previous party indicator suggests prior Republican alignment and the share of individuals with graduate or professional degrees exceeds $-0.2243$ increase the predicted Republican vote percentage by 14.06, representing the strongest contribution among the rules. Additional demographic factors also contribute positively, including population aged 65–74 (+6.40) and White non-Hispanic population share (+5.74), suggesting that districts with larger older and White populations tend to favor Republican electoral outcomes. Educational

attainment further contributes to the prediction, with the bachelor's degree share adding 3.84.

## 5.7 Interpretation of SR4-Fit rules for public classification datasets

Across the extracted rules for the breast cancer dataset, SR4-Fit identifies several cellular morphology patterns that distinguish malignant from benign tumors. Malignant classifications frequently occur when features indicating cellular irregularity and abnormal nuclei are elevated. Malignant outcomes appear when clump thickness exceeds approximately 0.7301 and bare nuclei values are greater than about $-0.4242$, even when uniformity of cell size remains at or below 0.1140. Malignancy is also associated with high variability in cell size (uniformity cell size $>$ 0.4405) combined with marginal adhesion values greater than about $-0.4647$, indicating irregular cell growth and stronger cell clustering. Additional malignant rules involve nuclear structure features such as bland chromatin values above approximately $-0.3861$ and bare nuclei values greater than about $-0.0123$, as well as rules where uniformity of cell shape remains below about 0.0954 with normal nucleoli values below approximately 0.2066 and mitoses below roughly 0.5179, highlighting the importance of chromatin structure and nuclear activity in identifying malignant cells. In contrast, benign classifications typically occur when cellular structures remain more regular. For instance, benign outcomes appear when uniformity of cell shape is relatively low ($\leq -0.2395$), bare nuclei values remain within moderate ranges (between about $-0.0123$ and 0.5370), and uniformity of cell size remains above approximately $-0.2125$. Additional benign

patterns occur when clump thickness remains small ($\leq 0.3753$) and single epithelial cell size remains above about $-0.3305$, indicating thinner and more stable cell clusters. Other benign rules involve lower marginal adhesion values ($\leq -0.4647$), even when uniformity of cell size exceeds 0.4405, suggesting that larger cell sizes without strong adhesion may still correspond to non-invasive tissue behavior.

Across the extracted rules for the E. coli dataset, most rules predicting the cp (cytoplasmic) protein class involve moderate or low values of alm1 ($\leq 0.6955$) combined with strongly negative mcg values ($\leq -1.6469$ or $\leq -1.7241$) and moderate ranges of alm2 and aac, suggesting weaker signal characteristics associated with cytoplasmic proteins. In contrast, rules predicting the im (inner membrane) protein class consistently involve higher feature values, including alm1 greater than approximately 0.3473 to 0.6258, mcg greater than about 0.5914, and alm2 exceeding approximately 1.5317, indicating stronger signal patterns associated with inner membrane protein localization.

Across the extracted rules for the page blocks dataset, SR4-Fit identifies patterns primarily driven by geometric properties of document blocks, including height, eccentricity, block area, length, pixel composition, and white-to-black transitions. Most rules predict class 1, which reflects the dominant category in the dataset and typically corresponds to standard document regions such as text blocks or simple rectangular areas. These rules commonly involve moderate or small block heights ($\leq 0.8981$), low eccentricity values ($\leq -0.4422$), and moderate pixel composition measures, indicating regular block shapes and consistent document textures. Some rules rely solely on geometric attributes such as area or length, suggesting that the physical dimensions of blocks are strong predictors of common document structures. In contrast, class 2 predictions occur when blocks become larger or taller (height $> 0.8453$) while maintaining specific eccentricity

values, indicating structural elements that differ from typical text regions.

Across the extracted rules for the Pima Indians Diabetes dataset, diabetic predictions generally occur when glucose levels exceed approximately 0.96 to 1.15, often combined with moderate or high BMI values ($> -1.1286$) or additional factors such as pregnancy count, insulin levels, or blood pressure. These conditions reflect well-known clinical indicators of diabetes risk, particularly elevated blood glucose and body mass index. In contrast, non-diabetic classifications occur when glucose levels remain relatively low ($\leq 0.3945$ or $\leq -0.2001$), often accompanied by lower BMI values and younger age groups, suggesting lower metabolic risk. Some rules also indicate that genetic risk factors, represented by the Diabetes Pedigree Function, may play a secondary role when glucose levels remain low.

Across the extracted rules for the vehicle dataset, SR4-Fit identifies interpretable patterns based on vehicle shape and geometric descriptors, including aspect ratios, elongatedness, circularity, rectangularity, variance, and scatter measures derived from vehicle silhouettes. Bus classifications occur when vehicle shapes exhibit larger elongated structures or compact rectangular silhouettes, often associated with moderate or low aspect ratios and specific variance ranges. Car classifications tend to appear when vehicle shapes are more compact, balanced, and circular, often characterized by moderate circularity and reduced scatter or hollow ratios. In contrast, van classifications are typically associated with box-like and rectangular silhouettes, reflected by moderate variance values, lower circularity, and specific rectangularity thresholds.

Across the extracted rules for the yeast dataset, the rules that predict the CYT (cytosolic) class, indicates that proteins belonging to this class exhibit distinctive combinations of signal intensities across these attributes. Several rules involve very low vac values ($\leq -2.3346$) combined with moderate or low mcg and gvh

values, suggesting that weak localization signals for certain cellular compartments correspond to cytosolic proteins. Other rules predict CYT when mcg and gvh values are relatively high ( mcg $> 1.7113$ or gvh $> 2.1397$), indicating alternative signal patterns associated with cytosolic localization. Additional rules rely on very low alm values ($\leq -1.3277$) or low mitochondrial indicators (mit $\leq 0.4656$) combined with moderate nuclear signal values, suggesting that the absence of strong compartment-specific signals can also indicate cytosolic proteins.

## 5.8 Interpretation of SR4-Fit rules for public regression datasets

In the Abalone dataset, the regression rules primarily emphasize the role of shell weight in determining the predicted outcome. When the shell weight is relatively high (shell weight$>0.8448$) and the shucked weight remains moderate (shucked weight$\leq 0.8860$), the model adds 0.56 to the prediction, indicating the strongest positive contribution. Abalones with moderate shell weight ($0.1150<$shell weight$\leq 0.8448$) also increase the prediction by 0.48, suggesting that mid-range shell weight still has a strong positive relationship with the target variable. In contrast, when shell weight is lower ($-1.3939<$shell weight$\leq -0.6147$) and the encoded sex value is less than 0.6568, the model contributes only 0.12, indicating a comparatively weaker influence.

In the Bone dataset, the regression rules primarily highlight the influence of age on the predicted outcome. When age is relatively low (age$\leq -0.4755$) and the encoded gender value is greater than $-0.0568$, the model contributes 0.01 to the prediction, indicating a small positive effect. Similarly, when age falls within the

range $-1.0279<\text{age}\leq-0.5504$, the prediction increases by 0.01, suggesting that individuals in this younger age interval also contribute positively to the outcome.

In the Diabetes dataset, when BMI is very high (BMI>1.5267) and S2 is relatively low ($\leq$0.5068), the prediction increases substantially by 63.8, while a similar condition with S5>$-$0.0210, BMI>1.5237, and S2$\leq$0.5107 results in an even stronger contribution of 68.7. When BMI remains high but S2 becomes larger, the contribution drops to 16.76 or 21.97, suggesting that S2 moderates the impact of high BMI. For moderate BMI (0.0745< BMI$\leq$1.5267), elevated S6 (>0.6737) increases the prediction by 31.04, while lower S6 reduces the effect to 4.56. Additional rules show that high S5 combined with elevated blood pressure (BP>1.3623) contributes 55.17, indicating strong metabolic interactions. Even when BMI is relatively low, combinations involving S3$\leq-0.7763$, S4>1.9298, or S1$\leq$0.9309 still produce contributions around 24, demonstrating that biomarker interactions beyond BMI can also significantly influence predicted diabetes progression.

In the Housing dataset, the rules highlight the strong influence of the average number of rooms (RM) on predicted housing prices. When room counts are very high (RM>1.6312 or RM>1.7026) and favorable neighborhood characteristics such as low pupil–teacher ratios (PTRATIO$\leq$0.6006 or $\leq-0.0796$), moderate tax rates (TAX$\leq$0.7703), or reasonable environmental conditions (NOX$\leq$0.8502) are present, the predicted housing value increases substantially by 12.12 to 15.15. For moderately high room counts (0.9091<RM$\leq$1.6312) combined with low lower-status population percentages (LSTAT$\leq$0.4702) and reasonable distances to employment centers (DIS>$-$0.9158), the prediction increases more modestly by 3.46. Even when room counts are smaller, combinations involving controlled crime rates, housing age, and socio-economic factors still contribute small positive ef-

fects of 0.58 to 2.04.

In the Machine dataset, the rules emphasize the importance of hardware capacity and baseline performance metrics in predicting machine performance. When PRP values are extremely high (PRP>3.3872 or PRP>3.9534) and large memory capacity (MMAX), the model increases the predicted performance by 20.19 to 27.50, while combinations of high PRP and specific model categories can contribute as much as 47.75. The strongest contribution occurs when both memory capacity (MMAX>2.9077) and maximum channel capacity (CHMAX>1.2501) are high, resulting in an increase of 49.42. Additional rules show that model and vendor identifiers also influence predictions, indicating that certain machine configurations or manufacturers correspond to higher predicted performance.

In the MPG dataset, the rules highlight the importance of engine size, horsepower, vehicle weight, and model year in determining fuel efficiency. Vehicles with very small engine displacement ($\leq -0.9550$), low horsepower ($\leq -0.7266$), and newer model years (model year>0.3881) increase the predicted MPG by 5.32, while low-cylinder engines with extremely low horsepower ($\leq -1.2563$) in newer vehicles produces an even stronger effect of 7.45. Lightweight vehicles also contribute positively, as seen when weight$\leq -0.8996$ combined with small displacement and newer model years increases MPG by 4.05. Additional rules show that moderate horsepower combined with low displacement and light vehicle weight adds 3.39 to the prediction. In contrast, vehicles with larger engines and higher horsepower, particularly from older model years, produce smaller effects such as 1.16, indicating weaker contributions to fuel efficiency.

In the Ozone dataset, the rules emphasize the importance of temperature, atmospheric pressure conditions, humidity, and visibility in determining ozone levels. When temperatures at Sandburg and El Monte are very high (Temp

Sandburg>1.2484 or Temp El Monte>1.3626) and IBT values are elevated, the predicted ozone concentration increases substantially by 6.21 to 7.07. The strongest increase occurs when Sandburg temperature exceeds 0.7464, IBT>0.9268, and visibility is extremely low ($\leq -0.9002$), resulting in an increase of 8.23, indicating that hot and stagnant atmospheric conditions strongly promote ozone formation. Additional rules show that low visibility combined with high temperatures and low wind speeds contributes between 5.04 and 5.71, reflecting the accumulation of pollutants when atmospheric dispersion is limited. Even when IBT values are lower, combinations of moderate temperature, reduced visibility, and higher humidity still increase ozone predictions by 2.36.

In the prostate dataset, the rules emphasize the role of tumor volume (lcavol), seminal vesicle invasion (svi), prostate weight (lweight), and other clinical markers in predicting PSA levels. When tumor volume is very high (lcavol>1.2430) and benign hyperplasia measurements are elevated (lbph$>-0.4522$), the predicted outcome increases by 0.48, indicating a strong association between tumor size and PSA levels. Similarly, the presence of seminal vesicle invasion (svi>0.6291) combined with high tumor volume (lcavol>1.2786) increases the prediction by 0.43, highlighting the influence of tumor spread. Additional rules show that high tumor volume combined with capsular penetration (lcp>2.1708) or moderate tumor volume with controlled Gleason scores (pgg45$\leq$2.1094) contributes smaller increases ranging from 0.17 to 0.23. Even when tumor volume is lower, combinations involving larger prostate weight and older age still increase predictions by 0.19.

# Chapter 6

# Conclusions and Future Work

This chapter is organized into two sections. The first section presents the conclusions drawn from the research conducted in this dissertation. The second section discusses potential directions for future work that extend the contributions of this dissertation.

## 6.1 Conclusions

This work presented SR4-Fit, an interpretable machine learning framework designed to generate compact, stable, and rule-based predictive models while maintaining competitive predictive performance. The primary application examined in this study is the U.S. House of Representatives election dataset, where the objective is to explain variations in Democratic and Republican vote percentages using demographic and socioeconomic characteristics derived from the American Community Survey (ACS). By extracting interpretable decision rules, SR4-Fit enables transparent analysis of how demographic patterns across congressional districts relate to electoral outcomes.

The rule sets generated for the U.S. House election data reveal interpretable relationships between demographic and socioeconomic variables, such as race, age distribution, education levels, income, marital status, and historical voting behavior. These rule-based explanations provide insight into how combinations of district-level characteristics influence Democratic and Republican vote shares, illustrating the usefulness of interpretable models for political and policy analysis, where transparency is critical.

To support and validate the proposed approach beyond the primary electoral application, SR4-Fit was also evaluated on several public regression and classification benchmark datasets. These additional experiments demonstrate that the method performs consistently across diverse domains while producing interpretable rule structures.

Interpretability in this study is evaluated using a quantitative interpretability score (IPS) that combines several key aspects of interpretable modeling: rule stability, number of extracted rules, predictive accuracy (measured using RMSE for regression or accuracy for classification), and rule complexity. Stability is assessed using the Dice–Sorensen similarity index across repeated trials, reflecting the consistency of rule structures. The number of rules measures model compactness, while rule complexity captures the number of conditions required to represent each rule. Incorporating predictive accuracy ensures that interpretability is evaluated alongside model performance rather than in isolation.

The experimental results show that SR4-Fit achieves a strong balance between predictive performance and interpretability. Compared with baseline approaches, the framework produces more stable rule sets with fewer rules and lower rule complexity while maintaining competitive predictive accuracy. These characteristics make SR4-Fit well-suited for applications where transparent and

reliable model explanations are essential. In particular, the stability of the extracted rules across repeated trials demonstrates the robustness of the framework in identifying consistent patterns within the data. At the same time, the reduced number of rules and lower rule complexity improve the cognitive interpretability of the model, making the resulting explanations easier for domain experts to understand and analyze. The use of the interpretability score further enables a quantitative assessment of the trade-offs between predictive accuracy, rule stability, model compactness, and rule complexity, allowing a systematic comparison across different models and datasets. Together, these results highlight the effectiveness of SR4-Fit in generating models that are not only predictive but also transparent, stable, and practically interpretable, which is particularly important for real-world decision-making scenarios where understanding the reasoning behind predictions is as critical as the predictions themselves.

Overall, the results demonstrate that SR4-Fit provides a practical and reliable approach for interpretable machine learning in complex real-world settings. By generating stable, compact, and human-readable rule sets, the framework enables transparent understanding of predictive relationships without sacrificing predictive performance. The analysis of the U.S. House election dataset illustrates how interpretable rules can reveal meaningful connections between demographic characteristics and electoral outcomes, highlighting the broader potential of interpretable models for social and policy-related research. At the same time, consistent performance across multiple public benchmark datasets confirms the general applicability and robustness of the proposed method. These findings suggest that SR4-Fit can serve as an effective tool for domains where explainability, reliability, and trust in model outputs are essential, contributing to the growing need for interpretable machine learning approaches in data-driven decision-making.

## 6.2 Future Work

One important direction for future work is the application of SR4-Fit to child neglect and child welfare analysis. Child neglect is a complex and multifaceted issue influenced by interacting factors such as socioeconomic conditions, caregiver characteristics, and access to support services. Existing analytical approaches often rely on regression-based methods, which are effective at identifying individual risk factors but struggle to capture interactions between multiple variables in a transparent and interpretable manner. In datasets such as the National Child Abuse and Neglect Data System (NCANDS), there is a need for models that can uncover structured combinations of risk factors while remaining interpretable and policy-relevant. SR4-Fit can provide a promising approach in this context, as it can generate human-readable rule sets that describe how combinations of demographic and service-related factors contribute to neglect recurrence and severity. Future work will also focus on applying SR4-Fit to NCANDS data to identify patterns of chronic neglect, evaluate the effectiveness of intervention services, and support data-driven decision-making in child protective services.

A second direction for future work is the development of an automated system for rule interpretation and summarization. In this dissertation, the interpretation of rule sets generated by SR4-Fit has been performed manually, which can be time-consuming and subject to human bias, particularly when dealing with large numbers of rules across multiple datasets. An automated system could be designed to identify the most important rules based on measures such as support, robustness, and contribution to predictive performance and then generate concise, human-readable summaries of these rules. Such a system could leverage natural language processing techniques to translate rule-based outputs into struc-

tured explanations, enabling users to quickly understand key patterns without manually analyzing each rule. This would significantly enhance the usability and scalability of SR4-Fit, especially in large-scale applications where interpretability must be maintained alongside efficiency.

A third direction for future work is to extend the SR4-Fit framework to support multiple regression outputs, enabling the model to handle multi-target or multi-output regression problems. In many real-world applications, multiple related continuous outcomes must be predicted simultaneously. For example, in electoral analysis, both Democratic and Republican vote shares can be modeled jointly, in healthcare applications, multiple health indicators may be predicted together, and in economic modeling, several interrelated financial variables may need to be estimated simultaneously. Extending SR4-Fit to accommodate multiple regression targets would allow the model to capture shared structures and dependencies across outputs while maintaining interpretability through rule-based representations.

A fourth direction for future work is to incorporate the idea of causal reasoning, which could be impactful on high-stakes application-based decision systems. While SR4-Fit generates interpretable rule sets that reveal meaningful associations between features and outcomes, such as demographic characteristics and electoral vote shares, these associations are inherently correlational. Extending the framework to support causal reasoning would allow practitioners to move beyond identifying which combinations of variables predict an outcome to understanding why those variables exert influence.

# Bibliography


Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer, and Cynthia Rudin. Learning certifiably optimal rule lists for categorical data. *Journal of Machine Learning Research*, 18(234):1–78, 2018.

Theodore S Arrington. Affirmative districting and four decades of redistricting: The seats/votes relationship 1972-2008. *Politics & Policy*, 38(2):223–253, 2010.

Arthur Asuncion and David J. Newman. Uci machine learning repository. `https://archive.ics.uci.edu/ml`, 2003. Contains Boston Housing dataset.

Clément Bénard, Gérard Biau, Sébastien Da Veiga, and Erwan Scornet. Sirus: making random forests interpretable. *ArXiv*, abs/1908.06852, 2019. URL `https://api.semanticscholar.org/CorpusID:201070843`.

Clément Bénard, Gérard Biau, Sébastien Da Veiga, and Erwan Scornet. Interpretable random forests via rule extraction. In *International Conference on Artificial Intelligence and Statistics*, pages 937–945. PMLR, 2021.

Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

Leo Breiman, Jerome Friedman, Charles J. Stone, and Richard A. Olshen. *Classification and Regression Trees*. CRC Press, 1984.

Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730, 2015.

Anne Chao, Robin L Chazdon, Robert K Colwell, and Tsung-Jen Shen. Abundance-based similarity indices and their estimation when there are unseen species in samples. *Biometrics*, 62(2):361–371, 2006.

William W Cohen et al. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning*, pages 115–123, 1995.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–499, 2004. doi: 10.1214/009053604000000067.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, NY, 2001a. ISBN 978-0387952840.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Bone mineral density (bmd) data set. `https://web.stanford.edu/~hastie/ElemStatLearn/`, 2001b. From The Elements of Statistical Learning.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Prostate cancer data set. `https://web.stanford.edu/~hastie/ElemStatLearn/`, 2001c. From The Elements of Statistical Learning.

Jerome H Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001.

Jerome H Friedman and Bogdan E Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

John N Friedman and Richard T Holden. The rising incumbent reelection rate: What's gerrymandering got to do with it? *The Journal of Politics*, 71(2): 593–611, 2009.

Jiangang Hao and Tin Kam Ho. Machine learning made easy: a review of scikit-learn package in python programming language. *Journal of educational and behavioral statistics*, 44(3):348–361, 2019.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and Their Applications*, 13(4):18–28, 1998.

Timothy O Hodson. Root mean square error (rmse) or mean absolute error (mae): When to use them or not. *Geoscientific Model Development Discussions*, 2022: 1–10, 2022.

Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3):1350 – 1371, 2015. doi: 10.1214/15-AOAS848. URL `https://doi.org/10.1214/15-AOAS848`.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017.

Donato Malerba. Page Blocks Classification. UCI Machine Learning Repository, 1994. DOI: https://doi.org/10.24432/C5J590.

Vincent Margot and George Luta. A new method to compare the interpretability of rule-based algorithms. *AI*, 2(4):621–635, 2021.

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

MIT Election Data and Science Lab. U.S. House 1976–2018. `https://doi.org/10.7910/DVN/IG0UN2`, 2017.

Christoph Molnar. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Lulu Press, Morrisville, NC, 2020. Available at `https://christophm.github.io/interpretable-ml-book/`.

Pete Mowforth and Barry Shepherd. Statlog (Vehicle Silhouettes). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5HG6N.

Shyam Sundar Murali Krishnan and Dean Frederick Hougen. SR4-Fit: An interpretable and informative classification algorithm applied to prediction of u.s. house of representatives elections. In *2025 International Conference on Machine Learning and Applications (ICMLA)*, pages 382–389, 2025. doi: 10.1109/ICMLA66185.2025.00058.

W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

Kenta Nakai. Yeast. UCI Machine Learning Repository, 1991. DOI: https://doi.org/10.24432/C5KG68.

Kenta Nakai. Ecoli. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5388M.

Warwick J. Nash, Tony L. Sellers, Steve R. Talbot, Anthony J. Cawthorn, and Wes B. Ford. Abalone data set. `https://archive.ics.uci.edu/ml/datasets/Abalone`, 1994. UCI Machine Learning Repository.

Josue Obregon and Jae-Yoon Jung. RuleCOSI+: Rule extraction for interpreting classification tree ensembles. *Information Fusion*, 89:355–381, 2023.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

J. Ross Quinlan. Auto mpg data set. `https://archive.ics.uci.edu/ml/datasets/Auto+MPG`, 1993a. UCI Machine Learning Repository.

J. Ross Quinlan. Computer hardware data set. `https://archive.ics.uci.edu/ml/datasets/Computer+Hardware`, 1993b. UCI Machine Learning Repository.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016.

Britt Richardson and Dean F Hougen. Districts by demographics: Predicting us house of representative elections using machine learning and demographic data. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 833–838. IEEE, 2020.

Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5(Jan):101–141, 2004.

C. J. Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, USA, 2nd edition, 1979. ISBN 0408709294.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

Smith, J. W. and Everhart, J. E. and Dickson, W. C. and Knowler, W. C. and Johannes, R. S. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. `https://archive.ics.uci.edu/ml/datasets/pima+indians+diabetes`, 1988.

Ashley Suh, Gabriel Appleby, Erik W Anderson, Luca Finelli, Remco Chang, and Dylan Cashman. Are metrics enough? guidelines for communicating and visualizing predictive models to subject matter experts. *IEEE Transactions on Visualization and Computer Graphics*, 2023.

John A Swets. Effectiveness of information retrieval methods. *American Documentation*, 20(1):72–89, 1969.

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

John W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, Reading, MA, 1977. ISBN 978-0201076165.

U.S. Census Bureau. 2006–2016 American Community Survey 1-Year Estimates of Selected Characteristics of the Total and Native Populations in the United States, Table S0601. `https://data.census.gov/cedsci/table?tid=ACSST5Y2015.S0601`, 2020.

WIlliam Wolberg. Breast Cancer Wisconsin (Original). UCI Machine Learning Repository, 1990. DOI: https://doi.org/10.24432/C5HP4Z.

Hongyu Yang, Cynthia Rudin, and Margo Seltzer. Scalable Bayesian rule lists. In *International Conference on Machine Learning*, pages 3921–3930. PMLR, 2017.

Peng Zheng, Travis Askham, Steven L Brunton, J Nathan Kutz, and Aleksandr Y Aravkin. A unified framework for sparse relaxed regularized regression: SR3. *IEEE Access*, 7:1404–1423, 2018.