

UMoncton CCNB-Innov Project *Technical Report*

M-A. Blais, M. Akhloufi
Perception, Robotics and Intelligent Machines (PRIME),
Université de Moncton,
18 Antonine Maillet Ave.
Moncton, NB E1A 3E9

2024-09-16

Contents

1	Introduction	5
2	Proposed Approach	8
3	Data	10
3.1	Data Origin	10
3.2	Data Modification	12
3.3	Datasets	14
4	Classification	17
4.1	All Features	17
4.1.1	Training Results	17
4.1.2	Feature Reduction	19
4.2	Specific Features	24
4.2.1	Training Results	24
4.2.2	Feature Reduction	25
4.3	Dataset Comparison	29
5	Regression	30
5.1	All Features	31
5.1.1	Training Results	31
5.1.2	Feature Reduction	34
5.2	Specific Features	39
5.2.1	Training Results	39
5.2.2	Feature Reduction	43
5.3	Dataset Comparison	47
6	Detailed Analysis of Ecosystemic Function (EF)s using Machine Learning (ML)	49
6.1	PR	50
6.2	NR	51
6.3	SR	52
6.4	WS	52
6.5	SFST	53
7	CausalML	54
7.1	Results	55
7.2	Discussion	57
8	Tetrad	57
9	Future Directions	58
9.1	Data	58
9.2	Machine Learning	58

9.3 Causal Inference	59
10 Conclusion	60
10.1 Classification Analysis	60
10.2 Regression Analysis	60
10.3 Phosphorus Retention (Phosphorus Retention (PR))	61
10.4 Nitrate Removal & Retention (Nitratre Remove & Retention (NR))	61
10.5 Sediment Retention & Stabilization (Sediment Retention & Stabilization (SR))	62
10.6 Water Storage & Delay (Water Storage & Delay (WS))	62
10.7 Stream Flow Support (Stream Flow Support (SFST))	62
10.8 Causal Inference with CausalML	63
10.9 Challenges with Tetrad	63
10.10Overall Conclusion	63
11 References	64
12 Annexes	66
12.1 Datasets	66
12.2 Features	66
12.3 Machine Learning Algorithms	68
12.4 Classification	71
12.4.1 All Features	71
12.4.2 Specific Features	73
12.5 Regression Models	74
12.5.1 All Features	74
12.5.2 Specific Features	81
12.6 CausalML	87

Executive Summary

This report presents an in-depth analysis of various ecosystem functions (EFs) using advanced machine learning (ML) techniques and Causal Inference (CI) methods. The primary objective was to evaluate the effectiveness of different feature sets in predicting EF ratings, assess the potential of specific features versus all available features and to explore the causal relationships between features and their effects on the various EFss. The EFs analyzed include Phosphorus Retention (PR), Nitrate Removal & Retention (NR), Sediment Retention & Stabilization (SR), Water Storage & Delay (WS) and Stream Flow Support (SFST).

The study employed classification and regression models to predict EF scores and ratings. It compared models trained on all features against those trained on a reduced set of specific features. The analysis revealed that while classification models generally performed better with all features, certain EFss, such as PR and WS, demonstrated robust performance with specific features as well. This indicates that key features carry significant predictive power, allowing for efficient predictions with fewer data points. However, for complex EFs

such as SR and SFST, maintaining a comprehensive set of features was important for the classification accuracy.

Regression models, which provided continuous predictions, generally outperformed classification models, particularly for EFss such as SR, NR and SFST. The results underscored the importance of using all available features for certain EFs.

For PR, the analysis identified features such as F43 (Channel Connection and Outflow Duration), F44 (Outflow Confinement) and F45 (Through Flow Resistance) as critical predictors. These features are directly related to the wetland's hydrological connectivity and water retention capacity, which significantly influence phosphorus retention. Similarly, for NR, features related to water retention and connectivity were consistently important across models, highlighting their role in nitrate removal through denitrification.

The study also explored the benefits of these EFs to human populations. For instance, PR Benefit, which evaluates the human benefits of phosphorus retention, relied on features like OF19 (Water Quality Downstream) and OF22 (Wetland Size Relative to Catchment). These results emphasize the importance of considering both environmental and human-related factors in environmental modeling.

In addition to traditional ML methods, we proposed CausalML, a Python library integrating CI with ML. This library can be used to identify causal relationships between features and EF outcomes. The causal analysis reinforced earlier findings, with features like F43 emerging as crucial across multiple EFss. This approach provided valuable insights into the mechanics of each EF, allowing for more targeted interventions and resource allocation.

However, attempts to further explore causal relationships using Tetrad, a CI modeling software, did not yield significant results, leading to a focus on the insights provided by CausalML. The challenges encountered with Tetrad suggest that further research and collaboration with CI specialists may be necessary to fully leverage its capabilities.

In conclusion, this report highlights the critical role of feature selection in ML models for environmental functions. It demonstrates that while feature reduction can streamline models and reduce data requirements, it is not without risks. The integration of CI techniques like CausalML offers a promising avenue for better understanding the dynamics at play within these ecosystems. It could provide more information for environmental decision-making. The findings of this study have significant implications for environmental management, conservation efforts and policy-making, offering a pathway toward more efficient and effective environmental assessments.

Keywords: Ecosystem Functions, Machine Learning, Causal Inference, Phosphorus Retention, Nitrate Removal, Sediment Retention, Water Storage, Stream Flow Support, Feature Selection, Environmental Modeling.

List of Acronyms

ML Machine Learning

WESP-AC Wetland Ecosystem Services Protocol for Atlantic Canada

CI Causal Inference

EF Ecosystemic Function

NB New-Brunswick

NRNB Natural ressources NB

WS Water Storage & Delay

PR Phosphorus Retention

SR Sediment Retention & Stabilization

NR Nitratre Remove & Retention

SFST Stream Flow Support

VT Variance Threshold

SKB SelectKBest

FS Function Score

BS Benefit Score

OF Office Forms

F Field Forms

S Stressors

NNorm Non-Normalized

Norm Normalized Non-Normalized

PNorm Pre-Normalized

1 Introduction

Wetlands are semi-aquatic ecosystems characterized by their frequent flooding, whether permanent or seasonal. These unique environments are essential for supporting a rich and complex biodiversity, contributing significantly to the health and sustainability of our planet's ecosystems [1]. Wetlands serve as critical habitats for a diverse array of plants [2], animals [3], birds [4] and fish [5], each benefiting uniquely from the wetland's resources. The abundance of nutrients in wetlands, such as nitrate and phosphorus, supports a diverse range of plant species, including aquatic plants, grasses, sedges and trees. This diversity is crucial

for the ecological functioning and sustainability of our planet [6]. They play a vital role in the ecosystem of our planet by efficiently converting CO₂ into oxygen, which is essential for maintaining the balance of our environment. Wetlands also provide essential breeding and feeding grounds for many animal species. Birds, in particular, utilize wetlands as critical stopover points during migration, while many fish species rely on the shallow waters of wetlands to protect their young from predators. Each wetland has its own fragile and complex ecosystem, where all components work in balance. There are several types of wetlands, such as marshes, bogs, fens and swamps, each with its unique biodiversity. Climate change represents one of the most significant threats to wetlands, particularly coastal wetlands, which face rising tides and increased erosion [7]. The delicate balance of wetland ecosystems is further disrupted as climate change alters the composition and distribution of plant and animal species. The addition or removal of species in wetlands can have a substantial impact on its functioning; for example, the removal of phragmites can reduce denitrification [8]. Logging, even when conducted near wetlands, can have long-lasting negative impacts by degrading vegetation and destabilizing the ecosystem [9]. The construction of logging roads can disrupt wetlands, leading to increased erosion, the spread of fires and other long-lasting ecological damage, persisting even after the roads have been abandoned [10]. Additionally, urban expansion and pollution can lead to the irreversible loss of wetlands [11, 12]. Climate change and other human activities cause irreversible damage to wetlands and their Ecosystem Functions (EFs).

EFs refer to the natural processes and interactions within an ecosystem that support its health, stability and sustainability. These functions include water filtration, carbon sequestration, nutrient cycling, flood control and habitat provision for a wide range of species. In wetlands, these functions are critical for maintaining water quality, regulating local climates and providing breeding grounds for many aquatic and terrestrial species. Each type of wetland contributes differently to these functions, depending on its specific characteristics and location. Water conservation and the retention of nutrients like nitrates, phosphorus and sediments are pivotal to the health of wetlands. The ability of wetlands to store water and regulate nutrient flows is essential for maintaining ecological balance and water quality in surrounding areas. Unchecked, excess nitrates and phosphorus from agricultural and urban runoff can lead to eutrophication, while suspended sediments can degrade aquatic habitats and biodiversity. Evaluating these EFs, such as nitrate retention, is crucial for assessing wetland health and provides conservation experts with valuable insights into the ecosystem's performance and benefits to surrounding communities. Accurate assessment of wetland health and its ecological importance, is a complex task that traditionally requires extensive data collection and long-term monitoring. To streamline this process, advanced tools like the Wetland Ecosystem Services Protocol for Atlantic Canada (WESP-AC) have been developed. This Excel-based tool analyzes data from structured questionnaires, comparing the functionality of wetlands against calibration sites, allowing researchers to assess the relative health of a wetland. The WESP-AC necessitates an extensive evaluation of each site, often taking several hours or even days to complete. The questionnaire is designed with intentional redundancy, where many questions overlap to ensure reliability.

To enhance the efficiency and rapidity of wetland evaluations, we propose approaches involving Machine Learning (ML) and Causal Inference (CI). By leveraging ML algorithms,

it becomes possible to identify the most influential variables across various EFs within wetlands, significantly reducing the burden of data collection. Our study incorporates multiple ML techniques, including classification, regression and CI, to streamline the evaluation process. Our goal is to significantly reduce the number of input features required to achieve results comparable to those obtained through the WESP-AC. Furthermore, we explore path analysis using Tetrad, a method rooted in CI, to uncover causal relationships within the ecosystem. Our findings demonstrate that by applying ML techniques, we were able to achieve an accuracy rate of 90% or higher for most EFs. EFs such as PR, NR, SR, WS and SFST achieved these results using as few as two to five key features. This significant advancement would enable researchers to assess wetland conditions in a matter of minutes rather than hours. This would dramatically accelerate the process of data collection and analysis and thus, enabling quicker and more effective conservation efforts. However, it is important to note that while these results are promising, they are exploratory in nature. Further research is needed to validate these findings, particularly by incorporating additional data from various provinces and wetland types to ensure the robustness and generalizability of the models.

In conclusion, the integration of advanced technologies such as ML and CI with traditional tools like WESP-AC presents promising opportunities for the sustainable management of wetlands. By optimizing the assessment of ecosystem functions such as water storage, nutrient retention and sediment stabilization, we can enhance our ability to protect these critical ecosystems. This, in turn, ensures their preservation and the continued provision of essential services, including high-quality water supplies, for future generations.

Our report is divided as follows:

1. **Data Collection:** We begin by discussing the comprehensive data collection process using the WESP-AC, which involves an extensive evaluation of various wetland sites. We also present our data modifications and the input features used to train our models.
2. **Proposed Method:** We present our proposed method which aims to reduce the number of input features required to achieve reliable results by leveraging ML techniques. We explore how advanced algorithms can streamline the assessment process.
3. **Classification:** We present various ML classification models to predict wetland functions and benefits ratings with high accuracy. The goal is to identify key features that significantly impact the performance of these models.
4. **Regression:** We also present regression models which are utilized to predict specific scores for various EFs. This section focuses on the effectiveness of these models in estimating wetland functions with reduced feature sets. We also introduce class grouping which is used to classify the sites based on the regression predictions.
5. **CausalML:** This section presents the integration of ML with CI techniques, specifically using the CausalML library. This approach aims to identify the causal relationships between features and their impact on wetland functions.
6. **Tetrad Path Analysis:** Finally, we investigate the potential of path analysis using Tetrad to uncover deeper causal relationships within the ecosystem. This method

is assessed for its ability to complement and enhance our understanding of wetland dynamics.

Our github can be found here: [Github](#)

2 Proposed Approach

We propose multiple approaches for this project, primarily based on Machine Learning (ML) with the integration of Causal Inference (CI) techniques. The two main objectives are feature reduction and feature comprehension.

For feature reduction, we focus on utilizing ML methods, specifically classification and regression, to accurately determine the ratings of Ecosystem Functions (EFs). The primary goal of feature reduction is to significantly reduce the number of input features required to assess a wetland's condition. By minimizing the necessary features, researchers can streamline site visits and potentially evaluate a greater number of wetlands more efficiently. Initially, our approach involves directly classifying each site into its respective EF ratings. These ratings are categorized into three levels: *lower*, *moderate* and *higher*, reflecting the wetland's performance relative to calibration sites. Throughout the project, we generated and tested multiple datasets derived from various sources and configurations. In this report, we compare the effectiveness of using a targeted set of specific features versus using all features and extra features. The datasets representing these approaches, named **D2** (specific features) and **D4** (all + extra features), showed the most promise. Our objective in this comparison is to evaluate how the inclusion of a greater or lesser number of features affects the accuracy and efficiency of the ML models. By analyzing the performance differences between **D2** and **D4**, we aim to determine the optimal balance between feature quantity and model effectiveness. This balance is crucial for enhancing the speed and precision of wetland assessments, enabling more rapid and reliable conservation efforts. We also present three additional datasets (D1, D3 and D5) in a supplementary document. To optimize the performance of our models, we trained various classification algorithms using a grid search, allowing us to identify the best combinations of algorithms, hyperparameters and data filling methods. To achieve our primary goal of reducing the workload, we implemented feature reduction techniques that identify the most critical features that are able to achieve similar results. Additionally, we employed ensemble learning to further enhance the performance of our models. We compared the impact of these methods on the final classification outcomes, evaluating how different feature reduction techniques, ensemble learning and other approaches influenced the accuracy and efficiency of the models.

Similarly, we propose using regression to predict the raw scores of each EF rather than directly predicting the classes. The objective of this approach is to develop a method that can predict the precise scores, which can then be utilized to determine the ratings of wetlands. As with our classification models, we compared the performance of regression models using the D2 and D4 datasets. Additionally, we evaluated the impact of training with both non-normalized and pre-normalized scores. Our rationale for employing non-normalized scores is to allow our models to function independently of specific calibration sites, thereby increasing

their versatility. However, we also explored the effects of training our algorithms with pre-normalized scores to understand any potential benefits. As with the classification approach, we applied feature reduction techniques and ensemble learning to enhance the performance of our regression models. To assess how well our regression models perform in comparison to classification models, we conducted a class grouping analysis using the predicted regression scores. This process involved using the predicted raw scores to categorize the wetlands into their respective classes. To ensure a fair comparison between non-normalized and pre-normalized scores, we also present the normalized predictions derived from the non-normalized scores. Finally, we identified the top-performing algorithms that achieved the best accuracy with the least number of features, across both classification and regression tasks. This comprehensive approach ensures that our models are not only accurate but also efficient in terms of feature utilization.

Three feature reduction techniques were used for the feature reduction axis: two variations of SelectKBest (SelectKBest (SKB)) and a variance threshold (Variance Threshold (VT)).

1. **Select K Best (F Classify)**: This technique utilizes the ‘SelectKBest’ class from ‘sklearn.feature_selection’ with the ‘f_classif’ score function. It selects the top k features with the highest ANOVA F-value.
2. **Select K Best (Mutual Info Classify)**: This method uses the ‘SelectKBest’ class with the ‘mutual_info_classif’ score function. It selects features based on the mutual information between each feature and the target variable, which measures the dependency between variables.
3. **VarianceThreshold**: This technique uses the ‘VarianceThreshold’ class from ‘sklearn.feature_selection’. It removes all features whose variance does not meet the specified threshold. A threshold of 0.1 was used in this study, but it can be adjusted based on the dataset.

For our feature comprehension axis, the goal is to provide insights into the features and their relationships with one another. By understanding the impact of features and their interactions, researchers can develop better models. We propose using the results of our ML algorithms and CI by implementing the CausalML library in Python. Through CausalML, we aim to identify which features have the most influence on various EFs. Determining the most important features can help researchers prioritize them during analysis. We also experimented with CI using Tetrad, which yielded less promising results. Given the limited initial success of these approaches, we focused our efforts primarily on feature reduction. However, we did interpret some feature comprehension results with our feature reduction findings.

3 Data

3.1 Data Origin

The data used in this project was derived from a local adaptation of the Wetland Ecosystem Services Protocol (WESP), specifically tailored for Atlantic Canada and known as WESP-AC. Originally developed by Dr. Paul Adamus, WESP is a rapid assessment technique designed to evaluate the critical natural functions of non-tidal wetlands. Non-tidal wetlands are freshwater ecosystems situated inland, unaffected by tidal forces, distinguishing them from their tidal counterparts.

The WESP-AC protocol evaluates various wetland ecosystemic functions using both field-based and office-based assessments. These functions include, but are not limited to, Water Storage & Delay (WS), Stream Flow & Temperature Support (SFST), Water Cooling, Sediment Retention & Stabilization (SR), Phosphorus Retention (PR), Nitrate Removal & Retention (NR), Carbon Sequestration and various habitat assessments. Each attribute is assigned a Function Score (FS) and a Benefit Score (BS), each calculated from their own set of features.

The WESP-AC features consist of a series of multiple-choice questions regarding the wetland, such as vegetation type. They are categorized into Office Forms (OF), Field Forms (F) and Stressors (S). A total of 111 features are evaluated, divided into 38 office-based, 68 field-based and 5 stressor-based features. These features are utilized to generate both the FS and BS of each EFs, which range from 0 to 10 before normalization. After normalization, these scores reflect the effectiveness of wetlands relative to previously assessed calibration sites.

The FS represent the effectiveness of specific wetland attributes, while the BS indicate how the EFs benefit humans. For instance, a highly effective but isolated wetland might have a high FS but a low BS due to its limited accessibility. The scores are normalized using the minimum and maximum scores observed in calibration sites, which can occasionally result in scores falling outside the 0-10 range. Additionally, ratings are assigned to both FS and BS, categorized as lower, moderate, or higher. The classes are based on statistical bounds generated using the Jenks natural breaks optimization method using the calibration sites. Table 1 present the minimum, maximum and the two boundaries used to generate the ratings.

Feature	Min	Max	Lower	Higher
WS	1.58	8.61	3.07	6.17
PR	2.07	10.00	3.66	6.11
NR	4.10	10.00	2.06	4.42
SR	2.29	10.00	3.02	6.67
SFST	0.00	7.71	1.05	6.51
WS Benefit	0.08	10.00	2.65	6.50
PR Benefit	0.49	10.00	3.29	6.68
NR Benefit	0.71	10.00	4.10	7.76
SR Benefit	0.49	8.79	2.94	6.19
SFST Benefit	0.00	7.19	1.86	5.30

Table 1: Jenks Configuration Values for Each Feature

These ratings provide valuable insights for local, regional and national authorities regarding wetland conservation. However, it is important to note that ratings provided by the WESP-AC are directly relative to the calibration sites.

For this project, we focused on WESP-AC data from wetlands located in New-Brunswick (NB), normalized using 98 wetlands from the same region. The EF selected for analysis were PR, SR, NR, WS and SFST. The EFs, as defined in the WESP-AC manual, encompass essential hydrologic functions crucial for wetland sustainability: They are defined as:

1. Water Storage & Delay (WS)

- **Definition:** The capacity to store runoff or delay the downslope movement of surface water for varying durations.
- **Potential Benefits:** Flood control and the maintenance of ecological systems.

2. Stream Flow Support (SFST)

- **Definition:** The effectiveness in contributing water to streams, particularly during the driest part of the growing season.
- **Potential Benefits:** Supporting fish and other aquatic life.

3. Sediment Retention & Stabilization (SR)

- **Definition:** The ability to intercept and filter suspended inorganic sediments, allowing deposition and reducing the energy of waves and currents, thereby stabilizing underlying sediments or soil.
- **Potential Benefits:** Maintaining water quality and protecting shoreline structures from erosion.

4. Phosphorus Retention (PR)

- **Definition:** The capacity to retain phosphorus over extended periods (at least 1 growing season).

- **Potential Benefits:** Maintaining the quality of receiving waters.

5. Nitrate Removal & Retention (NR)

- **Definition:** The ability to retain particulate nitrate and convert soluble nitrate and ammonium into nitrogen gas while minimizing the production of nitrous oxide (a potent greenhouse gas).

- **Potential Benefits:** Maintaining the quality of receiving waters.

An overview of the classes for each EF can be seen in Table 2.

	Lower	Moderate	Higher
PR	12	102	96
SR	95	63	52
NR	65	67	78
WS	85	70	55
SFST	74	48	88
PR Benefit	109	9	92
SR Benefit	118	75	17
NR Benefit	76	89	45
WS Benefit	144	41	25
SFST Benefit	80	48	82

Table 2: Classification statistics for PR, SR, NR, WS, SFST and their respective benefits

This table shows that certain EFs are unbalanced, such as PR which has 12, 102 and 96 sites for lower, moderate and higher respectively. Others, such as NR are balanced with 65, 67 and 78 sites respectively. This difference in classes can cause issue when training the models since it may learn to only predict the most popular classes. During our initial classification training, we implemented weight balancing to handle this issue.

While this study focused on these attributes, the same methodologies and techniques can be applied to other wetland functions. A total of 212 wetlands were utilized for training and validating our approaches. During initial training we used a split of 90:10 for training and validation while a split of 80:20 was used in later tests. For more comprehensive information, please refer to the *Manual for Wetland Ecosystem Services Protocol for Atlantic Canada (WESP-AC): Non-tidal Wetlands*.

3.2 Data Modification

In response to minor errors identified in WESP-AC version 3.3, a revised version, named 3.4, was created for this project. One significant change was the correction of the normalized benefit score formula. The normalization formula had an error which made the BS score normalized based on Nova-Scotia calibration data. Additionally, several trivial errors related to formatting and clarity were addressed to enhance the protocol's usability.

The data used for this project was provided by Natural resources NB (NRNB), referred to as "NRNB dataset". It combined the questionnaire responses for 212 non-tidal wetlands in NB, organized into an Excel file with a sheet for each feature type in the questionnaire (OF/F/S). Each row in these sheets corresponds to a specific feature, while columns represent individual wetland sites, each identified by a unique ID. However, the NRNB dataset did not include pre-calculated scores and ratings. This necessitated the development of a solution to generate WESP-AC 3.4 files for each site based on the NRNB data. A code was developed to create these files, which can be adapted for past and future versions of WESP-AC. A code was then generated to extract the scores and ratings of each of the WESP-AC 3.4 files.

To train the algorithms and techniques used in this project, the generated WESP-AC files were combined into a single dataset that included features and their scores or ratings. Two datasets were created: one for scores and one for ratings. The datasets were generated using specific code that processed the various WESP-AC features according to their format. A list of these processing techniques is provided below:

1. Binary Encoding:

- Converts a sequence of binary values (0s and 1s) to an integer based on the position of the '1'.

2. Float Extraction:

- Directly extracts floating-point values from specified cells.

3. F3 Split:

- Custom method to split data from specific rows and columns into separate values. This method created multiple feature columns for the F3 feature.

4. Binary to Integer:

- Converts a sequence of binary values (0s and 1s) to an integer.

5. Score Extraction:

- This method is used for specific cells in the 'Scores' sheet to accurately extract and round values. It uses the Xlwings library to generate and save the scores.

6. Ratings Extraction:

- The classification technique is used to convert textual categorical data into numeric values for easier processing and analysis. This is particularly useful when the data contains qualitative categories like "lower," "moderate," and "higher," which need to be transformed into numeric representations for further computations.

The ratings dataset is used for classification while regression techniques are applied to the scores dataset. Once generated, each dataset contained the features and either the scores or ratings for each site. The datasets were manually parsed to remove any unnecessary features. Features that contained only a single value (OF1, OF12, F60, F61), mostly the same values (OF32, F12, F27), or were mostly empty (OF35, OF36, F66A, S3, S6) were excluded from

this analysis. In future iterations, adding more sites could introduce greater feature variety, potentially making these features usable.

Some features had a few missing values due to the format of the WESP-AC questionnaire. To address this, nine different data filling methods were employed to impute missing data, resulting in nine sub-datasets. These methods are detailed below:

1. KNN:

- Imputes missing values based on the values of the nearest neighbors.

2. Mean:

- Fills missing values with the mean of the respective column.

3. Median:

- Fills missing values with the median of the respective column.

4. Mode:

- Fills missing values with the mode (most frequent value) of the respective column.

5. Forward Fill (FFill):

- Fills missing values by propagating the last valid observation forward.

6. Backward Fill (BFill):

- Fills missing values by propagating the next valid observation backward.

7. Linear Interpolation:

- Fills missing values by interpolating between known values.

8. Iterative:

- Imputes missing values using an iterative process that models each feature with missing values as a function of other features.

9. Custom:

- Imputes missing values using a custom-defined rule or value (e.g., -1).

The final datasets for both scores and ratings contained an ID, 32 OF, 69 F and 4 S. They also included the scores or ratings for WS, NR, PR, SR and SFST, for both their functions and benefits. More information can be found in the associated GitHub repository and within the comments in the code. Our data transfer method, from older version to recent ones can be found here: WESP-AC-Transfer

3.3 Datasets

Three types of data were used to generate our datasets for training. We used all features, the specific features for an EF and the extra features.

All features consist of using all 32 OF, 69 F and 4 S features generated in Section 3.2. The specific features were those used to generate the scores and ratings for the specific EFs. These features are presented in Table 3.

EF	Feature Count	Specific Features
PR	22	OF22, OF26, OF27, F17, F20, F21, F23, F24, OF28, OF29, F31, F33, F34, F35, F36, F38, F43, F44, F45, F49, F63, F65
PR Benefit	11	OF18, OF19, OF20, OF21, OF22, OF23, OF24, F41, F48, F50, F52
SR	19	OF22, OF26, OF27, F9, F17, F20, F22, F28, OF29, F31, F33, F34, F35, F36, F43, F44, F45, F49, S5
SR Benefit	14	OF18, OF19, OF20, OF21, OF22, OF23, OF24, F24, OF28, F41, F50, F52, F55, S4
NR	33	OF16, OF18, OF22, OF25, OF26, OF27, F1, F3, F6, F17, F18, F20, F21, F22, F23, F24, OF28, F31, F33, F34, F36, F43, F44, F45, F48, F49, F54, F65, S5
NR Benefit	14	OF9, OF10, OF11, OF19, OF20, OF21, OF22, OF23, OF24, F13, F41, F50, F51, F52
WS	13	OF22, OF26, F3, F20, F21, F22, OF28, F31, F43, F44, F45, F48, F49
WS Benefit	6	OF8, OF17, OF18, OF23, OF24, F51
SFST	14	F1, F3, F14, F17, F21, F24, F25, OF29, F31, F33, F34, F43, F47, F48
SFST Benefit	6	OF18, OF22, OF25, OF27, OF28, F42, F50

Table 3: Selected Features for EFs

Our extra data consists of using data collected outside the WESP-AC. This data is a collection of various sources and aggregates all 212 sites. In the annex, Table 30 shows the data and provides a quick description.

We combined these features to create various datasets, which are presented in Table 4. After preliminary analysis, we decided to present D2 and D4 in our final report due to their pertinence. However, we also present the D1, D3 and D5 datasets in a supplementary document.

Dataset Name	WESP-AC Features	Specific Features	Extra Features
<i>D1</i>	✓		
D2		✓	
<i>D3</i>			✓
D4	✓		✓
<i>D5</i>		✓	✓

Table 4: Dataset and Subset Tables

4 Classification

In ML, classification involves predicting a categorical class label based on a set of input features. For this project, we utilized Python’s SkLearn library to train and evaluate various ML classification algorithms. To ensure that each model was tuned for optimal performance, we conducted a comprehensive grid search across a range of hyperparameters, selecting the best configurations for each algorithm. A detailed list of these algorithms and their corresponding hyperparameters is provided in table 31.

In our analysis, we explored two distinct approaches to feature selection. In Section 4.1, we evaluate the performance of classification models using all available features, including additional ones beyond the standard dataset. Conversely, in Section 4.2, we focus on models trained solely with specific features directly related to the EFs. This dual approach allowed us to assess the impact of feature quantity and relevance on classification accuracy, providing insights into how different feature sets influence model performance. Our goal is to identify the most efficient feature subset that maintains high predictive accuracy while reducing the overall complexity of the model.

4.1 All Features

Initially, we trained our models using various algorithms, hyperparameters and data filling methods to identify the optimal configurations for each EF. After selecting the best-performing model for each EF, we conducted a deeper analysis to evaluate its effectiveness. To streamline the models and reduce computational complexity, we applied feature reduction techniques aimed at achieving high performance with a minimal set of features.

4.1.1 Training Results

Table 5 presents an overview of the best-performing algorithms paired with specific data filling methods.

Function	Model	Data	Accuracy
PR	KNeighbors	All Data	95.24%
NR	LogisticReg.	All Data	80.95%
SR	GradientBoosting	BFill	90.48%
WS	GradientBoosting	FFill, Mode, Interpolated	95.24%
SFST	GradientBoosting	All data	95.24%
PR Ben.	GradientBoosting	Custom, FFill, KNN, Median, Mode	95.24%
NR Ben.	GradientBoosting	All Data	95.24%
SR Ben.	Ridge, GradientBoosting	All Data	100.0%
WS Ben.	AdaBoost	BFill, Custom, FFill, Iterative, KNN, Mode	95.24%
SFST Ben.	AdaBoost,KNeighbors	FFill	95.24%

Table 5: Best Model Accuracies

In this table, each EF is shown with the accuracy of the best-performing model and data filling method. This table shows the accuracy achieved using all features with the D2 dataset. Most EFs were classified with an accuracy of 95.24% or higher. An exception is NR, which appears to have difficulties being classified. Additionally, the data filling method does not seem to have a significant impact, as most methods achieved similar results. In terms of algorithms, GradientBoosting seems to be slightly more advantageous than others. However, other algorithms, such as Ridge Classifier, performed similarly. To enhance the performance of our algorithms, we propose using ensemble learning to combine multiple models. We combined the top five models based on their highest accuracies. Overall, ensemble learning did not increase the performance of our algorithms, as shown in Table 6.

Function	Accuracy	Ensemble Accuracy
PR	95.24%	95.24%
NR	80.95%	80.95%
SR	90.48%	85.71%
WS	95.24%	95.24%
SFST	95.24%	95.24%
PR Ben.	95.24%	90.48%
NR Ben.	95.24%	95.24%
SR Ben.	100.0%	100.0%
WS Ben.	95.24%	95.24%
SFST Ben.	95.24%	85.71%

Table 6: Ensemble Model Accuracies

In fact, some EFs saw their accuracies decrease, such as SFST Benefit, which dropped from 95.24% to 85.71%. To better understand why ensemble learning can reduce performance,

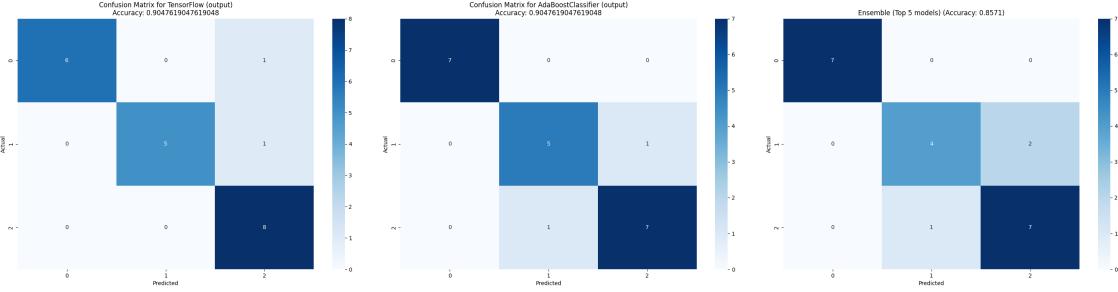


Figure 1: Effect of EL on SFST Benefit using Classification and All Features

we compared confusion matrices. Figure 1 shows the confusion matrices for the top two algorithms and ensemble learning for SFST Benefit.

This figure demonstrates why ensemble learning can reduce performance. In this figure, the leftmost image represents the best model, the middle image shows the second-best and the rightmost image illustrates the ensemble learning approach using the top five models. Since the top models have few errors that differ from model to model, when combined, these errors become more impactful. This example highlights the downside of ensemble learning when using small datasets. Overall, these results demonstrate that ML can be used to classify wetlands using the WESP-AC features. Our next section explores feature reduction techniques to reduce the input vector.

4.1.2 Feature Reduction

Feature reduction involves techniques to decrease the size of the input vector while maintaining or improving model performance. In this section, we explore the three feature reduction techniques outlined in Section 2. Using accuracy as the primary metric, we compared the performance of algorithms for each EF trained with reduced feature sets. The best-performing algorithm for each function was selected based on the accuracy metrics from Table 5. Each algorithm was then trained with the three reduction techniques, using a varying number of features, ranging from just two to the entire feature set. We analyzed and compared our results through both graphical and numerical representations. We used a training:validation split of 80:20, rather than 90:10, when training our reduced models.

Figures 12 to 21 in the annex illustrate the results for each EF with the three reduction techniques. In these figures, each EF is presented with its corresponding accuracy across different feature counts, offering deeper insights into the impact of feature reduction on model performance. Figures 2 and 3 provide a closer look at the impact of feature reduction techniques on SR Benefit and NR Benefit, respectively. In Figure 2, the relationship between feature count and accuracy is highlighted, showing that as the number of features increases, accuracy improves up to a point. It also illustrates the differences between the SKB and VT reduction techniques. SKB, using the F-classifier, achieved maximum accuracy with around 45 features, while VT required approximately 70 features to reach a similar level.

VT consistently performed the worst and only converged after using over 100 features with some EFs. This effect can be observed in figure 3.

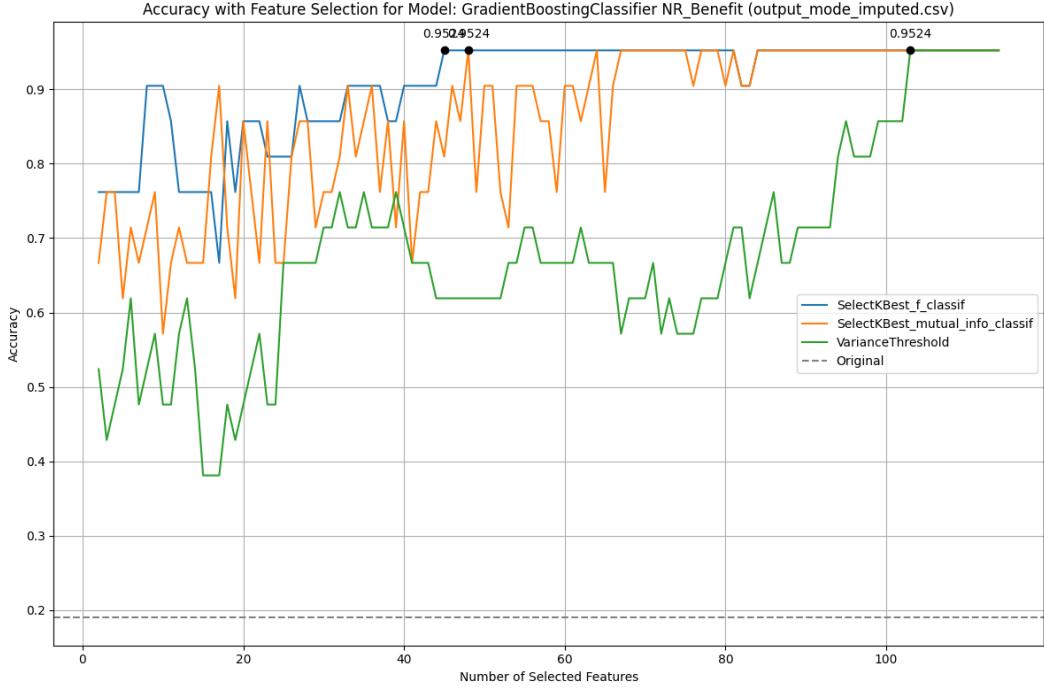


Figure 2: NR Benefit Feature Reduction D4

These results clearly demonstrate that SKB significantly outperforms VT in terms of both accuracy and efficiency. Furthermore, the SKB method enables a considerable reduction in the number of input features without compromising accuracy. Nonetheless, our feature reduction techniques have achieved interesting results. Our findings are further validated in Table 7, which presents the two best overall models with significantly reduced feature sets.

Function	Accuracy	Features	2nd Accuracy	2nd Features
PR	85.71%	F43, F45	90.47%	ProvClass, Fed-Class, F1, F14, F28, F29, F43, F44, F45
NR	80.95%	F43, F44	80.95%	F43, F44
SR	80.95%	Hydrogeo., F24, F29, F43, F44, F45	80.95%	F30, F31, F43, F44, F45
WS	90.45%	F43, F46	85.71%	F43, F44
SFST	95.24%	F43, F44	95.24%	F43, F44
PR Benefit	90.45%	F14, F24, F41, F43, F46, F47	85.71%	F41, F43

NR Benefit	90.45%	MossCover, OF9, OF10, OF19, F1, F14, F41, F43	76.19%	OF10, F43, F44
SR Benefit	100.00%	OF19, OF21, F41, F43	95.24%	OF19, F41
WS Benefit	95.24%	OF17, OF23	95.24%	OF7, OF17, OF23, F52
SFST Benefit	90.45%	ProvClass, MossCover, F1, F5, F23, F31, F43, F44, F45, F46	80.95%	F43, F44

Table 7: Maximum Proportionate Accuracy

In this table, when using a limited set of features, all EFs achieved an accuracy above 80%, with most achieving over 90%. Most impressively, WS achieved 90.45% while SFST achieved 95.24% using only two features. SR, NR and PR achieved respective accuracies of 80.95%, 80.95% and 85.71%. These slightly lower accuracies compared to WS and SFST suggest that these EFs may different training procedures to fully capture their complexity. Nevertheless, the results remain impressive given the reduction in input features. For instance, using only F43 and F44, we achieved accuracies of 80.95% and 85.71% for NR and PR, respectively. SR required slightly more features, utilizing F30, F31, F43, F44 and F45 to achieve 80.95%. WS and SFST performed exceptionally well, achieving accuracies of 90.45% and 95.24% with only two features. WS used F43 and F46, while SFST relied on F43 and F44. A clear pattern emerges, showing that features F43 to F44 are crucial in the classification of wetlands. F43, in particular, appears to be the most influential, as it is included in all EFs.

For the Benefit ratings, the achieved accuracies were generally higher than the function ratings. The accuracy ranged from 90.45% to 100.00% for the Benefit ratings, though they typically required slightly more features than the function scores. It's noteworthy that the function ratings (e.g., NR, PR) did not achieve higher scores with an increased number of features. The reduced feature accuracy was rarely higher than by using all features. In the rare cases where accuracy was higher, it required a significant amount of features. Regarding the classification of benefits, SR Benefit achieved the best performance, with 100% accuracy using four features (OF19, OF21, F21, F43) and 95.24% using only OF19 and F41. WS Benefit also performed well, achieving 95.24% accuracy with only OF17 and OF23. The results for WS Benefit are particularly interesting since our model relied solely on OF features. This approach could allow researchers to assess the WS Benefit of wetlands without needing to visit the site.

NR Benefit struggled to achieve high accuracy with fewer features. For example, it reached only 76.19% accuracy with three features (OF10, F43, F44) but achieved 90.45% with eight features. This indicates that some functions may require significantly more features due

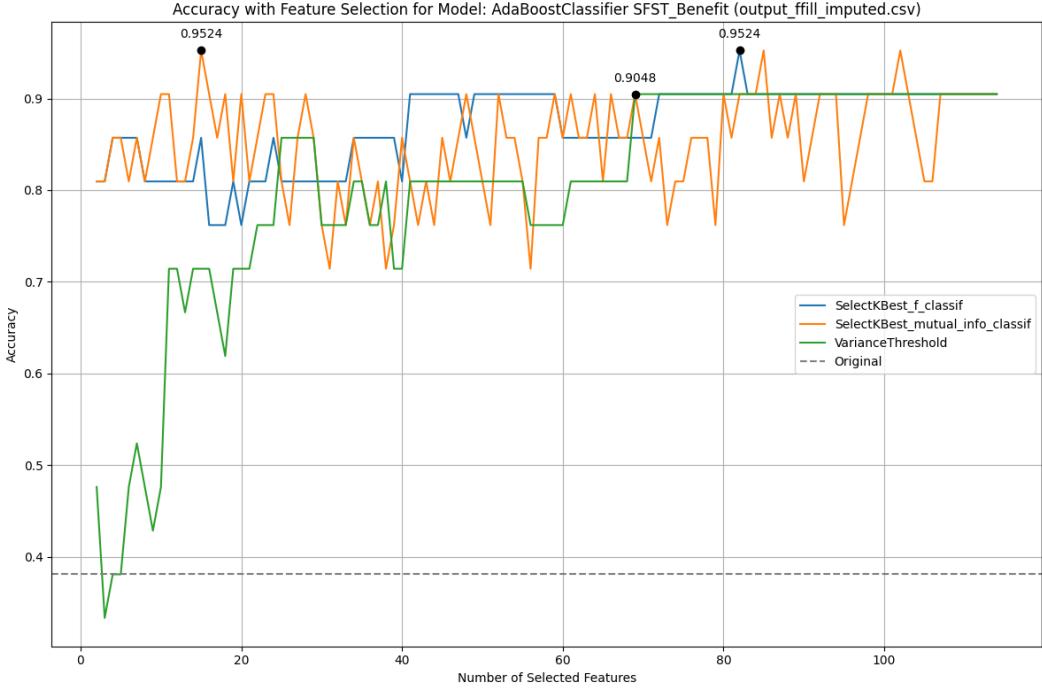


Figure 3: SFST Benefit Feature Reduction D4

to their inherent complexity. Similarly, PR Benefit achieved 90.45% with six features and 85.71% using only F41 and F43. Finally, SFST Benefit achieved 90.45% with ten features and 80.95% using F43 and F44. These results highlight the differences between function and Benefit ratings. Both types of ratings performed well with a limited number of features, but the Benefit ratings required slightly more features to achieve higher accuracies. This could be explained by the fact that the benefit of a wetland may be highly complex in nature. Overall, features F41 to F43 seem to be critical for both function and Benefit ratings. Additionally, the benefit ratings seem to be more influenced by OF features than the function ratings. This can be explained by the fact that an EF function is mostly related to on-site activities such as vegetation, soil and water. On the contrary, the benefit is more influenced by human activities, which can be gathered without visiting the site. Both Provincial and Federal classes appear to play a role in predicting certain EFs, likely because these features represent multiple features. Overall, the extra features did not seem to benefit our models with the exception of a few EF.

4.1.2.1 Ensemble Learning To further improve the performance of our feature reduction approach, we implemented ensemble learning. This technique is designed to average out errors and enhance performance without the need for retraining or complex algorithms. Our ensemble model for each EF was based on the top five models from the feature reduction process, using no more than ten features per model.

Table 8 shows the performance of each function before and after applying ensemble learning.

Function	Best Acc.	2nd Best Acc.	Ensemble Learning Acc.	Features Used
PR	90.48%	85.71%	85.71%	F1, F14, F28, F29, F43, F44, F45, FedClass, ProvClass
NR	80.95%	80.95%	76.19%	F43, F44, F45, F46
SR	80.95%	80.95%	80.95%	F1, F14, F24, F28, F29, F30, F31, F3e, F41, F43, F44, F45, F46, FedClass, OF22, ProvClass
WS	90.48%	90.48%	90.48%	F43, F44, F45, F46
SFST	95.24%	95.24%	95.24%	F43, F44, F45, F46
PR Benefit	90.48%	85.71%	85.71%	F14, F24, F41, F43, F44, F46, F47
NR Benefit	90.48%	90.48%	85.71%	F1, F14, F31, F3d, F41, F43, F44, F46, MossCover, OF10, OF19, OF9
SR Benefit	100.00%	100.00%	95.24%	F14, F28, F29, F41, F43, F44, OF19, OF21
WS Benefit	95.24%	95.24%	95.24%	F50, F52, F54, OF17, OF23, OF7
SFST Benefit	90.48%	85.71%	85.71%	F1, F14, F23, F31, F43, F44, F45, F46, F5, Moss Cover, ProvClass

Table 8: Ensemble Learning Accuracy

In this table we present the accuracy of the best and second model, the ensemble learning accuracy and the features used for ensemble learning.

The top two models in this table are not necessarily the same as those in Table 7. The five models chosen for ensemble learning were selected purely based on the highest accuracy with fewer than ten features, aiming to minimize bias. This table confirms our earlier observation that ensemble learning can sometimes reduce accuracy. Indeed, seven out of ten EFs experienced an approximate 5% drop in accuracy, while the others remained unchanged. Additionally, the number of features used was significantly increased with ensemble learning. However, ensemble learning could still be beneficial if more training data was available. Ensemble learning typically requires larger datasets than those we used. Given the smaller size of our dataset, our validation data was limited to approximately 40 sites, meaning a

single error could reduce the overall accuracy by 2.5%. This also indicates that the ensemble learning approach can be significantly impacted by a single misclassification. As a result, ensemble learning did not enhance our algorithms as expected. Nonetheless, even without ensemble learning, our feature reduction techniques yielded strong performance.

4.2 Specific Features

This section utilizes our D2 datasets, which consist solely of features specific to each EF. This significantly reduces the input vector from over 100 features to a range of six to 33. As in the previous section, we compared our results across various data filling methods; however, only the best results are presented.

First, we present the training results and the ensemble learning approach using all D2 features in Section 4.2.1. Our ensemble learning algorithm combines the top five best-performing algorithms, with predictions averaged to improve overall accuracy.

Section 4.2.2 details our feature reduction approach, leveraging the results from Section 4.2.1. In this section, we evaluate the performance of our approach for each EF based on prediction accuracy. Our goal, as in the previous section, is to achieve strong performance with the fewest possible features. We also investigate the impact of applying ensemble learning to the reduced feature models, aiming to enhance accuracy while maintaining a streamlined feature set.

4.2.1 Training Results

First, we performed a grid search of our algorithms, hyperparameters and data filling methods on our D2 dataset. Table 9 provides an overview of the best-performing algorithms for each EF.

Function	Model	Data	Accuracy
PR	Ridge	All Data	95.24%
NR	SGD	Mean	90.48%
SR	TensorFlow	Median	100.00%
WS	GradientBoosting	All Data	100.00%
SFST	GradientBoosting	All data	95.24%
PR Ben.	KNeighbors	All Data	95.24%
NR Ben.	SVC	All Data	100.00%
SR Ben.	Ridge, GradientBoosting	All Data	100.0%
WS Ben.	RandomForest	BFill, Custom, Interpolated, Iterative, Mean, Median, Mode	95.24%
SFST Ben.	RandomForest	Custom, Mean	71.42%

Table 9: Best Model Accuracies

In this table, each EF is shown with the accuracy of the best-performing model and data filling method. A slight increase in performance can be observed when comparing these results with the results from using D4, found in Table 5. In fact, the accuracy has increased for five EFs, remained unchanged for four and was reduced for one EF. Most impressively, using only specific features, we achieved an accuracy of 100% on five EFs. Notably, SR increased from 90.48% to 100.00% and NR from 80.95% to 90.48%. However, SFST Benefit was considerably reduced from 95.24% in the previous section to 71.42%. We thoroughly verified these results and determined that a feature present in D4 but not in D2 is crucial for this EF. Comparing the specific features for SFST Benefit and the results from feature reduction using all features in Table 7 highlights this gap. SFST Benefit appears to be mostly dependent on OF questions, while using all features demonstrated that F features are among the most important. Questions remain as to why our models using specific features did not perform well for SFST Benefit. The data may be highly uncorrelated when using specific features, which could hinder the model’s ability to learn. Nonetheless, we achieved very good performance on all other EFs using only specific features.

To further improve the performance of our algorithms, we proposed using ensemble learning to combine multiple models. Overall, ensemble learning did not consistently enhance the performance of our algorithms, as shown in Table 10.

Function	Accuracy	Ensemble Accuracy
PR	95.24%	95.24%
NR	90.46%	85.71%
SR	100.00%	100.00%
WS	100.00%	100.00%
SFST	95.24%	95.24%
PR Ben.	95.24%	100.00%
NR Ben.	100.00%	100.00%
SR Ben.	100.0%	100.0%
WS Ben.	95.24%	95.24%
SFST Ben.	71.42%	76.19%

Table 10: Ensemble Model Accuracies

SFST Benefit was slightly improved through ensemble learning, increasing from 71.42% to 76.19%. Similarly, PR Benefit increased from 95.24% to 100.00%. Figures 4 and 5 show PR Benefit and SFST Benefit using the top two models and ensemble learning, respectively.

In these figures, we see how ensemble learning can enhance results by averaging bad predictions into their correct classes.

4.2.2 Feature Reduction

As with D4, we applied feature reduction techniques to minimize the input feature vector of our D2 dataset. In this section, we explore the three feature reduction techniques outlined

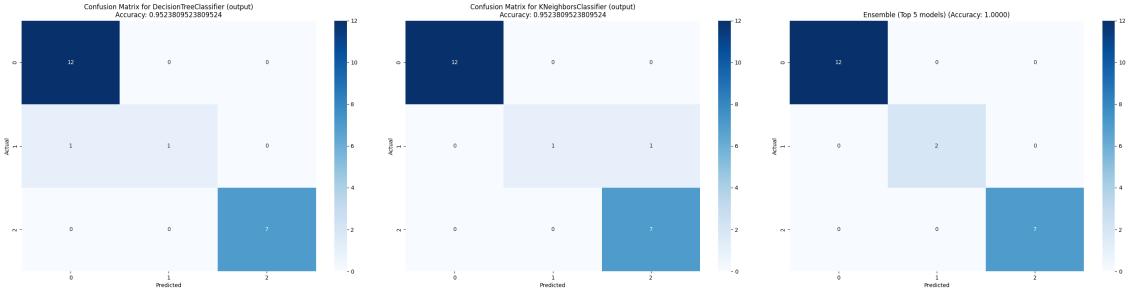


Figure 4: Effect of EL on PR Benefit using Classification and Specific Features

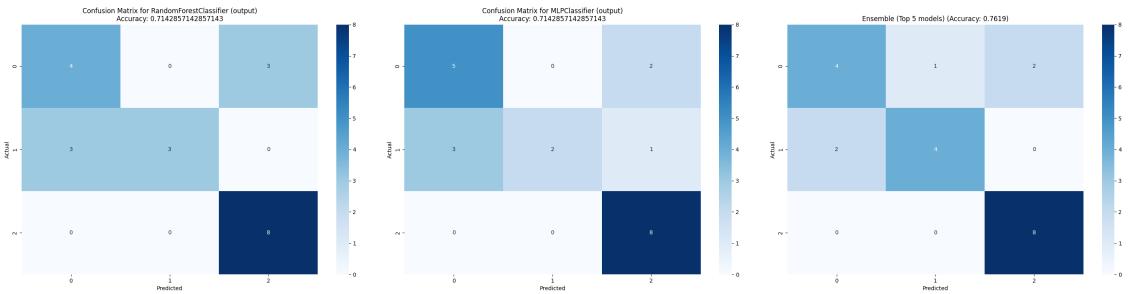


Figure 5: Effect of EL on SFST Benefit using Classification and Specific Features

earlier in Section 2. Using accuracy as the primary metric, we compared the algorithms for each EF trained with reduced features. The best-performing algorithm for each function was selected based on the training accuracy from Table 9. Each algorithm was trained with the three reduction techniques, utilizing features ranging from two to all features specific to the EF.

Table 11 presents the two best overall models for each EF with reduced features.

Function	Accuracy	Features	2nd Accuracy	2nd Features
PR	85.71%	F43, F45	90.45%	F21, F23, F24, F28, F29, F31, F43, F44, F45
NR	80.85%	F43, F44, F45	76.19%	F43, F44
SR	71.42%	OF22, F35, F43, F45, F49	61.90%	F43, F44
WS	85.71%	F43, F44	85.71%	F31, F43
SFST	95.24%	F1, F43	95.24%	F1, F24, F43

Function	Accuracy	Features	2nd Accuracy	2nd Features
PR Benefit	85.71%	OF22, F41	90.47%	OF22, F41, F48, F50
NR Benefit	90.48%	OF9, OF10, OF19, OF21, F41	80.95 %	OF10, OF20, F41
SR Benefit	95.24%	OF19, F41	100.0%	OF19, OF21, F41
WS Benefit	80.95%	OF17, OF23	95.24%	OF17, OF24, F51
SFST Benefit	57.14%	OF22, OF28	57.14%	OF18, OF22, OF28

Table 11: Maximum Proportionate Accuracy

At first glance, this dataset underperformed compared to D4, with accuracies ranging from 71.42% to 95.24%. When using D2, all EFs achieved lower accuracies than those achieved using the reduced feature set from the D4 dataset. SFST Benefit was the most significantly impacted, with its accuracy dropping from 80.95% using two features to 57.14%. Overall, it seems that using all available features and then applying feature reduction techniques yields better results than using only specific features from the outset. This discrepancy could be due to certain features, absent in D2, that capture complex interactions or provide unique insights not available in the specific feature set. Figures 22 to 31 in the annex demonstrate the effect of the various feature reduction techniques. Figure 6 highlights the impact of feature reduction on WS.

In this figure, the earlier-mentioned difference between SKB and VT is clearly visible once again. SKB significantly outperformed VT, converging more quickly and efficiently. In fact, SKB with F-classifier achieved 100.00% accuracy using nine features, while VT required 15. This figure underscores the importance of carefully analyzing results to better understand the behavior of different reduction techniques.

4.2.2.1 Ensemble Learning To further enhance the performance of our feature reduction approach, we proposed using ensemble learning. This technique is designed to average out errors and improve performance without retraining or the need for complex algorithms. Our ensemble model for each EF is based on the top five models derived from feature reduction, with no more than 10 features per model.

Table 12 shows the performance for each function before and after applying ensemble learning.

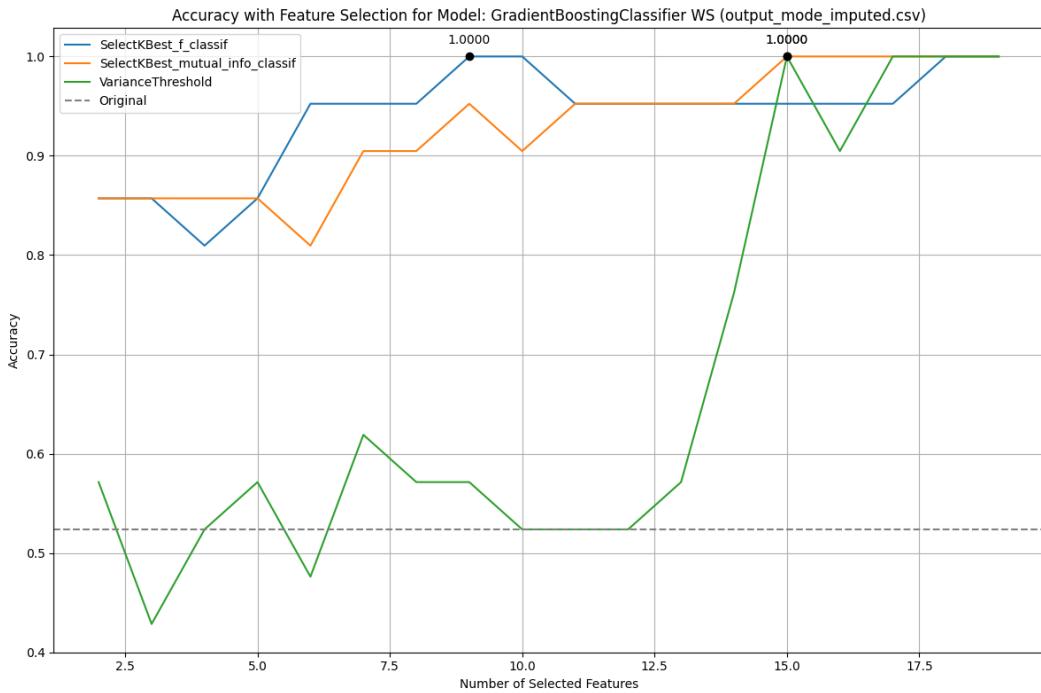


Figure 6: WS Feature Reduction D4

Function	Best Acc.	2nd Best Acc.	Ensemble Learning Acc.	Features Used
PR	90.48%	90.48%	85.71%	F21, F23, F24, F28, F29, F31, F43, F44, F45, F49
NR	80.95%	76.19%	71.43%	F1, F24, F28, F31, F43, F44, F45
SR	80.95%	76.19%	76.19%	F22, F28, F29, F31, F33, F34, F35, F36, F43, F44, F45, F49, OF22
WS	100.00%	100.00%	100.00%	F20, F21, F22, F28, F31, F3_a, F3_b, F3_c, F3_d, F3_e, F3_f, F3_g, F43, F44, F45, F48, F49, OF22, OF26
SFST	95.24%	95.24%	95.24%	F1, F14, F24, F31, F43
PR Benefit	95.24%	95.24%	95.24%	F41, F48, F50, F52, OF18, OF19, OF20, OF21, OF22, OF23, OF24
NR Benefit	90.48%	90.48%	90.48%	F13, F41, F50, F52, OF10, OF19, OF20, OF21, OF22, OF23, OF24, OF9
SR Benefit	100.00%	100.00%	100.00%	F24, F28, F41, OF19, OF20, OF21, OF22

WS Benefit	95.24%	95.24%	95.24%	F51, OF17, OF18, OF23, OF24, OF8
SFST Benefit	57.14%	57.14%	52.38%	F50, OF18, OF22, OF25, OF28

Table 12: Ensemble Learning Accuracy

In this table, the first column lists the EF, while the second and third columns display the best and second-best accuracy from Section 4.2.2 using a maximum of 10 features. The fourth column represents the accuracy of the ensemble learning model and the last column lists the features of all five models used for ensemble learning. As seen in previous sections, ensemble learning did not lead to an increase in the performance of our models.

4.3 Dataset Comparison

To facilitate a clear comparison of the results, we present the best-performing models for each EF using both all features and specific features. The models that yielded the most notable outcomes are summarized in Table 13.

Function	Acc. All	Features All	Acc. Specific	Features Specific
PR	85.71%	F43, F45	85.71%	F43, F45
NR	80.95%	F43, F44	80.85%	F43, F44, F45
SR	80.95%	Hydrogeo., F24, F29, F43, F44, F45	71.42%	OF22, F35, F43, F45, F49
WS	90.45%	F43, F46	85.71%	F43, F44
SFST	95.24%	F43, F44	95.24%	F1, F43
PR Benefit	90.45%	F14, F24, F41, F43, F46, F47	85.71%	OF22, F41
NR Benefit	90.45%	MossCover, OF9, OF10, OF19, F1, F14, F41, F43	90.48%	OF9, OF10, OF19, OF21, F41
SR Benefit	100.00%	OF19, OF21, F41, F43	95.24%	OF19, F41
WS Benefit	95.24%	OF17, OF23	80.95%	OF17, OF23
SFST Benefit	90.45%	ProvClass, MossCover, F1, F5, F23, F31, F43, F44, F45, F46	57.14%	OF22, OF28

Table 13: Comparison of Best Accuracy and Features from Two Tables

Table 13 illustrates that classification using all features generally outperforms using specific features. The accuracy for several EFs, such as SR and SFST Benefit, decreased significantly when specific features were employed. This reduction in accuracy suggests that some critical features may be absent in the D2 dataset. For example, SFST Benefit’s accuracy dropped from 90.45% to 57.14%, indicating the importance of certain features available only in the D4 dataset.

Conversely, some EFs, such as PR and WS, maintained consistent accuracy across both datasets, highlighting the robustness of the selected features in these cases. The results for WS Benefit are particularly noteworthy, as the model achieved a high accuracy of 95.24% using only OF features. This finding is significant because it suggests that WS Benefit can potentially be analyzed without requiring field data, which could significantly simplify the analysis process.

However, other EFs, such as NR Benefit, required a larger number of features to achieve satisfactory accuracy. This complexity could be due to the intricate nature of these EFs, which necessitate more comprehensive data for accurate classification. Overall, while feature reduction techniques can simplify the model and reduce the required data, they may also result in a loss of accuracy, particularly for more complex EFs.

The application of ensemble learning produced mixed results. While some EFs benefited from the approach, others experienced reduced accuracy. Future research should focus on identifying the optimal balance between feature reduction and model accuracy, potentially by exploring additional feature selection techniques or expanding the dataset to include more diverse wetland types. Moreover, the exploration of more advanced ensemble learning methods or hybrid models could further improve classification accuracy, especially for the more complex EFs.

5 Regression

Regression is a common type of algorithm used in ML with a key difference from classification. Classification predicts a class, such as lower, moderate or higher, while regression predicts a continuous value. In our case, the continuous value would be the score for the function or benefit of an EF. The classification results from 4 have shown great promise and is interesting for the project. However, the algorithms trained for classification used the normalized scores, for both the function and benefit. In turn, the algorithms are specific to regions, such as for NB and depend on the calibration wetlands. If different calibration wetlands are used or the region modified, the algorithms could underperform significantly. For this reason, we propose using regression to predict the non-normalized and normalized scores of the various EF. We propose using the non-normalize score such that it does our approach does not depend on calibration sites or region. However, we also compare using pre-normalized scores to compare the results. The data was collected, modified and generated using simple pre-processing techniques as seen from section 3. We present our results using both D4 and D2 datasets, respectively using all features and specific features. We use the Mean Squared

Error (MSE) as metric, which averages the squares of the errors and is defined in equation 1.

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

5.1 All Features

Similar to our classification technique, we perform a grid search of various regression algorithms, hyperparameters and data filling methods. We trained our algorithms and techniques for PR, NR, SR, WS and SFST, for both the Function and Benefit. More information about the training procedure can be found within the GITHUB, under regression. The algorithms used for regression are shown in table 31. Similar to classification, we first propose comparing the regression algorithms using all features for the EF. We compare our training results, feature reduction techniques and class grouping. Class grouping consists of classifying the regression predictions such that we can compare our regression and classification. We perform the same tests using our D2 datasets which consists of specific features only.

5.1.1 Training Results

First, we show the training results using the top accuracy over any algorithm and data filling method in table 14. This table shows the MSE for the Non-Normalized (NNorm), Normalized Non-Normalized (Norm) and Pre-Normalized (PNorm) of the best models. Using this figure, we can see a significantly increase in MSE when comparing NNorm and Norm. In fact, the MSE is often doubled when normalizing the prediction from using NNorm. This is expected since normalizing predictions scales the errors into a bigger scale. When comparing NNorm and PNorm, NNorm achieved better results for most EFs. When looking at EF Benefits, NNorm performed better on PR and WS while PNorm performed better on SR, NR and WS. NNorm achieved an MSE of 1.19 for WS Benefit while PNorm achieved a lower MSE of 0.63. On the contrary, PNorm achieved an MSE of 0.49 for PR Benefit while NNorm achieved 0.22. This difference in results demonstrates the purpose of comparing both NNorm and PNorm data.

Feature	NNorm MSE	Norm MSE	PNorm MSE
NR	0.17	0.48	0.31
PR	0.12	0.19	0.21
SR	0.37	0.62	0.46
SFST	0.20	0.33	0.26
WS	0.25	0.49	0.60
NR Benefit	0.39	0.45	0.28
PR Benefit	0.22	0.23	0.49
SR Benefit	0.17	0.22	0.07
SFST Benefit	0.51	0.98	0.71
WS Benefit	1.19	1.21	0.63

Table 14: Summary of Best MSE for Non-Normalized, Normalized and Pre-Normalized Data

In the annex, tables 32 and 33 respectively show the full results for the NNorm, Norm and PNorm data. In terms of algorithm, the AdaBoost regressor has achieved an overall better performance. The MLP and SGD regressor were also found to be effective at predicting the scores. In terms of data filling method, each EF had their best performing method but overall each performed similarly. The results presented in this section demonstrated interesting results. It showed that both NNorm and PNorm data can be used for regression while Norm should not be prioritized.

We also implemented ensemble learning in the goal to increase our performance. Table 15 presents the results with and without ensemble learning for all three normalization. In this table, we can see that our results were not significantly impacted by ensemble learning. The MSE for NR Benefit using NNorm was slightly reduced from 0.39 to 0.27 while WS Benefit was reduced to 0.86 from 1.19. This small increase in performance could be beneficial for our class grouping.

Feature	Training			Ensemble Learning		
	Non-Norm	Norm	Pre-Norm	Non-Norm	Norm	Pre-Norm
PR	0.12	0.19	0.21	0.12	0.19	0.22
NR	0.17	0.48	0.31	0.17	0.48	0.28
SR	0.37	0.62	0.46	0.39	0.65	0.50
WS	0.25	0.49	0.60	0.25	0.49	0.61
SFST	0.20	0.33	0.26	0.21	0.36	0.27
NR Benefit	0.39	0.45	0.28	0.27	0.64	0.36
PR Benefit	0.22	0.23	0.49	0.31	0.32	0.31
SR Benefit	0.17	0.22	0.07	0.13	0.17	0.06
WS Benefit	1.19	1.21	0.63	0.86	0.87	0.68
SFST Benefit	0.51	0.98	0.71	0.38	0.72	0.60

Table 15: Ensemble Learning Effect on All Features Training

In the annex, figures 32 to 41 show graphs with the algorithms and ensemble learning. In

these figures, we present all three normalization techniques with each their top three models and their ensemble approach. When looking at the overall clusters of normalization for each EF, we can observe the difference between Norm or PNorm and NNorm data. Normalized data, either post or pre-normalized, were more spread out while non-normalized data was often in small clusters. This demonstrates the benefit of using normalized data since it provides a more usable range. However, it also showcases the issue with using pre-normalized data. When using pre-normalized data, our model is bounded by the normalization unless the model is retrained. Using our models to predict non-normalize scores while normalizing the predictions offers the benefits without the issues. However, normalizing non-normalized data increases the MSE significantly due to the scaling.

When looking at the WS Benefit figure 40, it demonstrates the powerfullness of ensemble learning. More specifically, looking at the NNorm results we can observe that one model always over estimates while two underestimate. However, with the ensemble learning these results are averaged out and a better prediction is achieved. These figures, such as figure 36, demonstrates how well our models performed. In fact, in this figure most sites were predicted correctly with only one missclassification. However, this figure demonstrates how well our models were able to group clusters.

Table 16 show the class grouping while tables 34, 35 and 36 show the detailed results in the annex.

Feature	Non-Norm		Norm		Pre-Norm	
	Overall	Ensemble	Overall	Ensemble	Overall	Ensemble
PR	100.00%	95.24%	90.48%	90.48%	100.00%	95.24%
NR	90.48%	90.48%	85.71%	85.71%	85.71%	85.71%
SR	100.00%	100.00%	100.00%	95.24%	100.00%	100.00%
WS	100.00%	100.00%	100.00%	95.24%	100.00%	100.00%
SFST	85.71%	85.71%	100.00%	95.24%	100.00%	95.24%
PR Benefit	90.48%	90.48%	95.24%	90.48%	90.48%	90.48%
NR Benefit	95.24%	95.24%	100.00%	95.24%	100.00%	100.00%
SR Benefit	95.24%	95.24%	95.24%	95.24%	95.24%	95.24%
WS Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST Benefit	90.48%	90.48%	95.24%	90.48%	95.24%	95.24%

Table 16: Summary of Overall and Ensemble Accuracy for Non-Normalized, Normalized and Pre-Normalized Data

In table 16, we present the class grouping accuracies for all three normalization. For non-normalized, values were not normalized but the same boundaries for the classes were used as the normalized data. All three methods achieved similar results with Norm and PNorm achieving slightly better results on most EFs. However, NNorm performed better on NR than either normalized data and achieved 90.48% compared to 85.71% for normalized. Over all three normalization techniques, we achieved six 100%, three 95.24% and one 90.48%. These results are highly promising and demonstrated that regression can be used for predicting

the scores. Not only can they accurately predict the scores, they are able to use those scores to classify the sites. When compared against our classification models, our regression models significantly outperforms them. In fact, using all features we achieved an average accuracy of 93.811% for classification and 97.62% for regression. This difference is important especially since regression offers the possibility of directly predict the scores. Similar to classification, NR had difficulties converging to a good model compared to the other EFs. Further exploration into this would be highly beneficial in next iterations of this project.

5.1.2 Feature Reduction

Similar to classification, our primary goal is to achieve similar results with considerably less features. Table 17 show our most promising results.

Function	NNorm	Norm	Features	PNorm	Features
PR	0.3721	0.5917	F43, F44, F45	0.5237	F43, F44, F45, Fed-Class
SR	0.6319	1.0629	F24, F28, F29, F31, F43	0.8303	F24, F28, F29, F44
NR	0.2786	0.5506	F24, F43, F44, F45	0.7361	F24, F43, F44, F45
WS	0.2627	0.5250	F22, F24, F31, F43	0.4642	F22, F31, F43
SFST	0.2953	0.4967	F30, F43, F44, F45, F46	0.5688	F43, F44
PR Benefit	1.2611	1.3809	F14, F41, F43, OF19	1.3627	F14, F41, F43, OF19
SR Benefit	0.6983	0.6315	F41, OF19, OF20, OF21	2.0620	F41, OF19, OF21
NR Benefit	3.0280	3.5086	F14, F41, OF10, OF19, OF9	3.4702	F14, OF10, OF19, OF9
WS Benefit	0.7507	0.7629	F51, F65, OF17, OF23, OF24	0.9022	Hydrogeo., OF17, OF23, OF24
SFST Benefit	1.1648	2.2533	F1, F43, F44, F46	2.3641	F43, F44

Table 17: Best MSE with Corresponding Features

In this table, we present each function with the best NNorm and PNorm models. We choose the models for each based on the lowest MSE with a maximum of five features. We present both their MSE and features used with the addition of the MSE for the normalized values of NNorm, named Norm. Overall, Norm achieved a higher MSE than NNorm with certain EF more than quadrupling their MSE. Certain EFs achieved a lower MSE when normalizing the NNorm data. SR Benefit had its MSE reduced from 0.6983 to 0.6315 using the same NNorm model with normalized predictions. It even achieved a lower MSE than by using the PNorm dataset. NNorm obtained the overall best MSE which could be caused by the range of values being lower. By using a smaller range, the models can easily learn to predict the values into a smaller cluster, thus reducing the MSE. Using NNorm data could easily provide

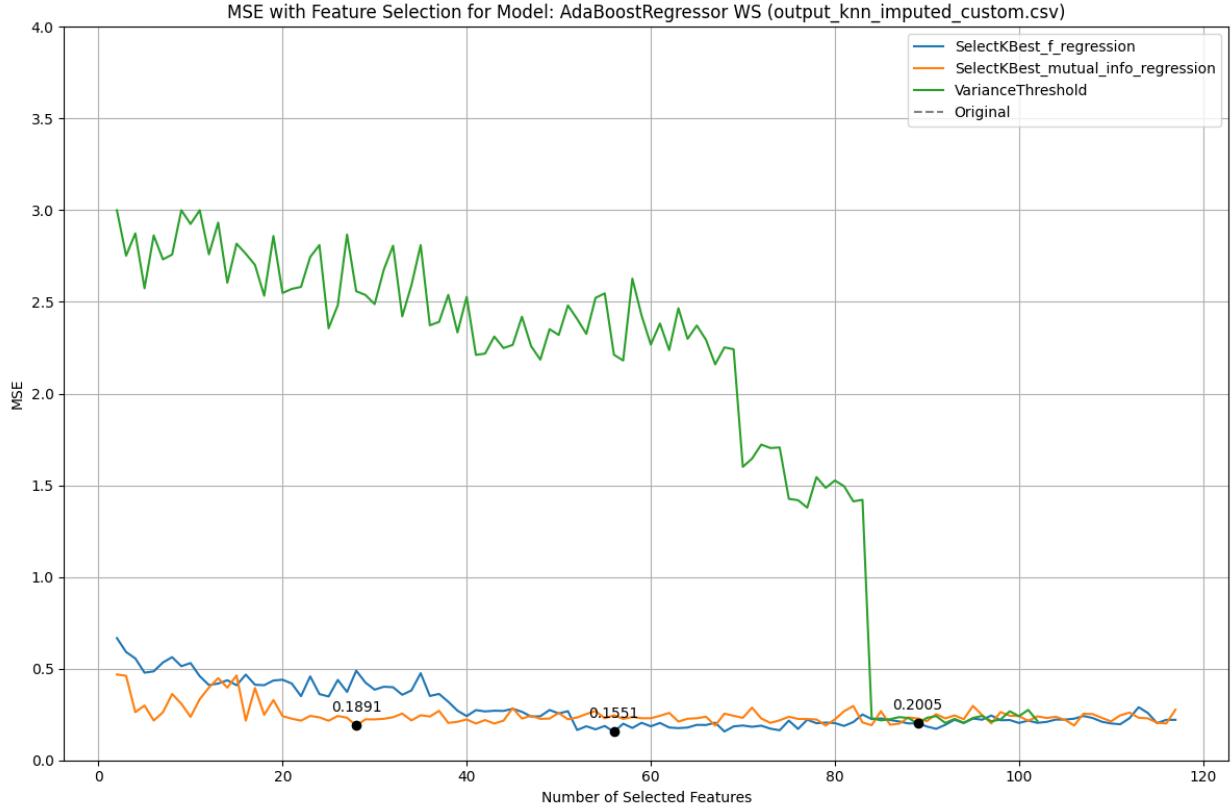


Figure 7: WS NNorm Feature Reduction

a solution to pinpoint the exact score. However, since our goal is to classify these sites into our three classes, we must normalize them accordingly. In terms of normalization, PNorm has a clear advantage over the Norm data. It achieved a better MSE on most EF and, more importantly, was consistent across all EFs. In this table, we can also observe a recurring issue with NR Benefit not converging with either NNorm, Norm or PNorm. In fact, compared to other EF functions using NNorm, it achieved 3.0280 compared to the second highest at 0.2611. For PNorm, NR achieved an MSE of 3.4702 compared to the second highest at 2.3641. Further exploration into this issue would benefit our project in future iterations. Other than NR Benefit, other EFs achieved good scores using either NNorm and PNorm. Similar to classification, the EF Functions converged more efficiently than the Benefits. In terms of input features, the WESP-AC features are mostly used with the exception of the hydrogeomorphic class in PR Benefit for PNorm. These results demonstrate that using the extra data may not be viable solution. More information can be found in the supplementary data which has a section dedicated on using only extra features. This experiment further confirmed our suspicion that the extra features were not beneficial currently. Adding more wetland sites or different extra features could possibly yield interesting results.

In terms of reduction technique, both SKB significantly outperformed VT for all EFs. Figure 7 show the effect of the various feature reduction techniques on WS.

In this figure, we can observe that both SKB quickly found the most important features

Feature	NNorm		Norm		PNorm	
	MSE	Features	MSE	Features	MSE	Features
PR	0.3807	F43, F44, F45	0.6054	F43, F44, F45	0.4887	F43, F44, F45, Fed-Class
SR	0.7661	F24, F28, F29, F31, F43, F44	1.2888	F24, F28, F29, F31, F43, F44	0.8795	F24, F28, F29, F44
NR	0.2617	F24, F43, F44, F45, ProvClass	0.5116	F24, F43, F44, F45, ProvClass	0.7226	F23, F24, F43, F44, F45, Prov-Class
WS	0.2692	F22, F24, F28, F31, F43	0.5371	F22, F24, F28, F31, F43	0.4695	F22, F24, F31, F43
SFST	0.2988	F30, F43, F44, F45, F46	0.5026	F30, F43, F44, F45, F46	0.4920	F30, F43, F44, F45, F46
PR Benefit	1.2746	F14, F41, F43, OF19	1.3957	F14, F41, F43, OF19	1.3258	F14, F41, F43, F46, OF19
SR Benefit	0.4994	F41, OF19, OF20, OF21	0.4863	F41, OF19, OF20, OF21	2.1352	F14, F41, OF19, OF20, OF21
NR Benefit	3.0567	F14, F41, OF10, OF19, OF9	3.5417	F14, F41, OF10, OF19, OF9	3.7400	F14, OF10, OF19, OF9
WS Benefit	0.7557	F51, F65, OF17, OF23, OF24	0.7679	F51, F65, OF17, OF23, OF24	0.8649	F51, Hydrogeo., OF17, OF23, OF24
SFST Benefit	1.1563	F1, F23, F43, F44, F45, F46	2.2367	F1, F23, F43, F44, F45, F46	2.3455	F1, F23, F43, F44, F46

Table 18: Ensemble Learning MSE and Features Used for Non-Normalized, Normalized and Pre-Normalized Data

while VT did not. This pattern is repeated for all **EFs!** (**EFs!**) using any NNorm, Norm or PNorm.

Using the top five models, Table 18 show our models with ensemble learning.

In this table, we present each NNorm, Norm and PNorm with the MSE achieved and features used for ensemble learning. Overall, ensemble learning did not increase our performance significantly. Using NNorm, SR Benefit was reduced to 0.4994 using ensemble learning compared to 0.6983 using the top model. Similarly, PNorm did not have a significant advantage

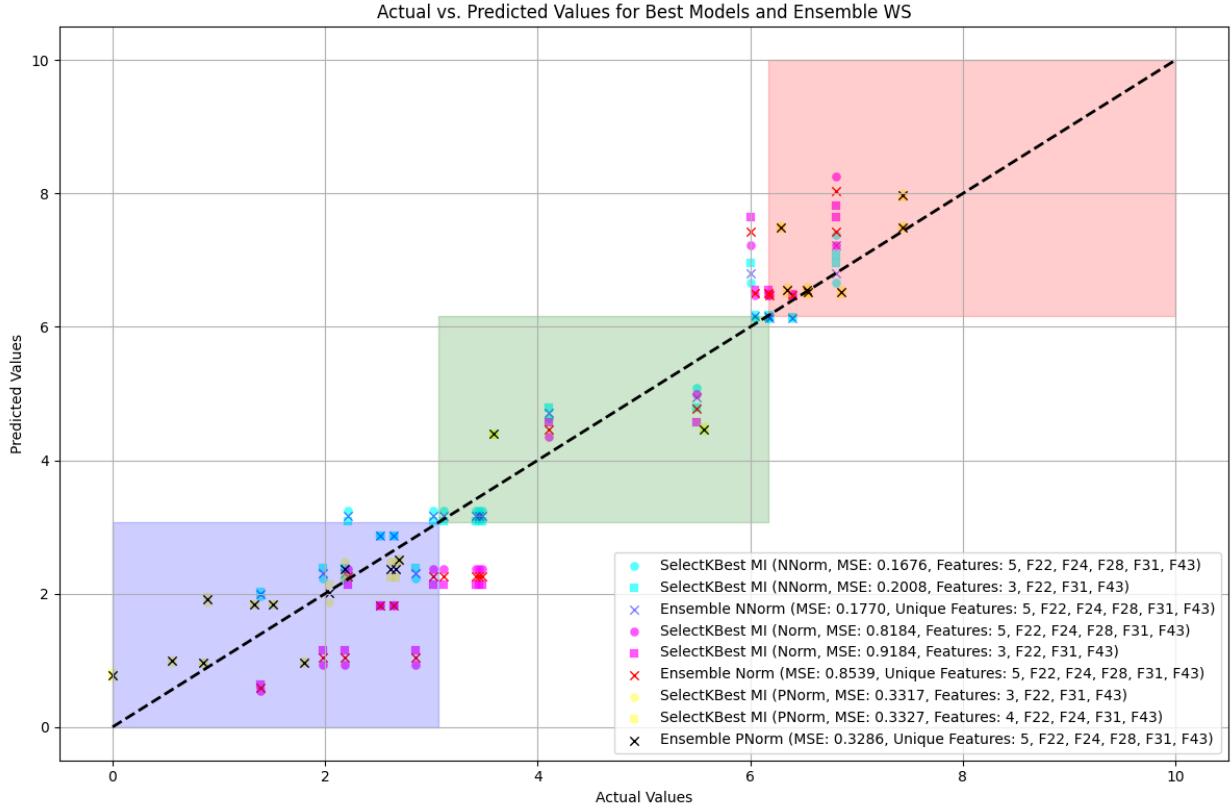


Figure 8: SR NNorm Feature Reduction

using the ensemble learning approach. Furthermore, ensemble learning increased the number of features used to achieve similar or worse results. For these reason, we assume ensemble learning to not be effective for regression for this project at this time. Further data in our training sequence could make ensemble learning beneficial.

Figure 8 show a graph the predictions of one of our best results, SR.

In this figure, the top two models and our ensemble learning for our three datasets (NNorm, Norm, PNorm) are presented. In this figure, the classes are presented with blue, green and red squares for lower, moderate and higher ratings. We can observe how the normalization has an important effect on the results when comparing NNorm against Norm and PNorm. NNorm seems to mostly be concentrated in the moderate class while Norm and PNorm are more evenly spread out. It also demonstrates our initial hypothesis on the reason why NNorm achieves a lower MSE than Norm and PNorm. NNorm can cluster the prediction closely together since the range is smaller. Nonetheless, both Norm and PNorm have achieved good results with most sites being in the correct class. Figures 42 to 51 in the annex show the graphs for all EFs. These figure also demonstrate the difference in results with some EFs performing very well, such as WS while others, such as SFST Benefit, did not. However, most of our EFs achieved impressive results and were able to converge.

To confirm our results, we propose class grouping to compare the results with classification.

We use the lower and higher boundaries presented in section 3 to classify the results. We present an overview of all three data normalization (NNorm, Norm, PNorm) in table 19. We compared the class grouping using the same boundaries for all three data normalization. Meaning the non-normalized results use the same classes as the normalized results. This was done in order to compare the results against each other.

Function	NNorm		Norm		PNorm	
	Overall	Ensemble	Overall	Ensemble	Overall	Ensemble
PR	88.10%	90.48%	76.19%	76.19%	78.57%	76.19%
SR	80.95%	88.10%	85.71%	85.71%	85.71%	83.33%
NR	85.71%	85.71%	85.71%	71.43%	88.10%	76.19%
WS	85.71%	73.81%	90.48%	90.48%	90.48%	90.48%
SFST	90.48%	90.48%	88.10%	88.10%	88.10%	88.10%
PR Benefit	80.95%	80.95%	85.71%	90.48%	90.48%	90.48%
SR Benefit	95.24%	92.86%	92.86%	92.86%	95.24%	90.48%
NR Benefit	90.48%	92.86%	78.57%	83.33%	92.86%	92.86%
WS Benefit	78.57%	78.57%	80.95%	80.95%	80.95%	83.33%
SFST Benefit	83.33%	83.33%	80.95%	80.95%	83.33%	83.33%

Table 19: Summary of Overall and Ensemble Accuracy for Non-Normalized, Normalized and Pre-Normalized Data for Class Grouping

When looking at the summary in table 19, PNorm seems to be the most promising normalization method. In fact, NNorm achieved an accuracy of 87.38%, Norm achieved 84.99% and PNorm achieved 87.62%. NNorm achieving similar classification results as PNorm is unexpected. Initially we assumed NNorm would perform the worst since it was not normalized with the classification bounds. Nonetheless, it achieved very impressive results with a classification range from 85.71% to 97.62%. PNorm achieved a larger range from 78.57% to a maximum of 95.24%. More impressive is the possibility of NNorm to classify NR with an accuracy of 90.48% compared to 78.57% using PNorm. When looking at the global accuracy, meaning the highest for each EF, we achieved an overall accuracy of 89.53%.

Table 20 shows the best model for each EF with the MSE, overall accuracy and features used.

Function	Model/Data	MSE	Accuracy	Features	Figures
PR	NNorm, Ensemble	0.3807	90.48%	F43, F44, F45	42
SR	PNorm, AdaBoost, KNN	0.8303	88.10%	F24, F28, F29, F44	44
NR	PNorm, AdaBoost, KNN	0.7361	85.71%	F24, F43, F44, F45	43
WS	PNorm, AdaBoost, Custom	0.4642	90.48%	F22, F31, F43	45
SFST	PNorm, AdaBoost, Custom	0.5688	88.10%	F43, F44	46
PR Benefit	PNorm, MLP, FFill	1.3627	90.48%	F14, F41, F43, OF19	47
SR Benefit	PNorm, AdaBoost, FFill	2.0620	95.24%	F41, OF19, OF21	49
NR Benefit	PNorm, MLP, KNN	3.4702	92.86%	F14, OF9, OF10, OF19	48
WS Benefit	PNorm, AdaBoost, Mean	0.9022	80.95%	Hydrogeo., OF17, OF23, OF24	50
SFST Benefit	PNorm, AdaBoost, Iterative	2.3641	83.33%	F43, F44	51

Table 20: Best Model Results for All Features

In this table, we can see that PNorm was the most efficient data normalization technique while AdaBoost was the best algorithm.

5.2 Specific Features

5.2.1 Training Results

Table 21 shows a summary of our training results. In order, this table shows NNorm, Norm and PNorm for all EFs functions and benefits. With the exception of SFST Benefit, all EFs performed well and achieved good performance. In fact, we achieved MSE ranging from 0.01 to 0.16 when excluding SFST Benefit, which achieved 1.66. When comparing the data normalization, NNorm achieved the overall best performance, closely followed by PNorm. Norm achieved a higher MSE than NNorm and PNorm for most EFs, with a few exceptions. Certain EFs achieved a better MSE using NNorm, while others performed better using PNorm. PR Benefit excelled using any normalization and achieved an MSE of 0.01 across all normalization techniques. Overall, the Benefit scores for NR, SR and PR achieved a better MSE than their function counterparts. When compared against Table 14, which uses all features, we achieved significantly better results.

When comparing all EFs, we achieved an average MSE of 0.359, 0.520 and 0.402 for all features, while specific features achieved 0.295, 0.614 and 0.960 for NNorm, Norm and PNorm, respectively. However, due to SFST Benefit achieving an abnormally high MSE, we verified the results using the median. In terms of median, we achieved 0.22, 0.45 and 0.31 using all features, while we achieved 0.10, 0.19 and 0.24 using specific features. When removing SFST Benefit, we achieved averages of 0.312, 0.469 and 0.367 using all features, while specific features achieved 0.142, 0.211 and 0.137. These results demonstrate two things: first, they show how a single outlier can skew the perception of our results. At first glance, using the averages, all features performed better than specific features. However, when removing the outlier, the specific features significantly outperformed all features. These results also show that SFST Benefit is missing important features in the specific dataset to infer its score. Our specific features achieved lower results for both classification and regression for SFST Benefit. This highlights our reasoning for using all features rather than only specific features. The WESP-AC may only require specific features to calculate the score, while ML may require different ones. Nonetheless, our results using only specific features are highly promising, with most EFs achieving low MSE.

Feature	Non-Normalized MSE	Normalized MSE	Pre-Normalized MSE
NR	0.15	0.43	0.24
PR	0.11	0.18	0.24
SR	0.08	0.13	0.27
SFST	0.16	0.26	0.20
WS	0.10	0.19	0.08
NR Benefit	0.08	0.09	0.07
PR Benefit	0.01	0.01	0.01
SR Benefit	0.07	0.09	0.01
SFST Benefit	1.66	3.22	4.37
WS Benefit	0.53	0.54	0.11

Table 21: Summary of Best MSE for Non-Normalized, Normalized and Pre-Normalized Data

In the annex, figure 39 shows the full results for NNorm and Norm, while figure 40 shows PNorm. In these tables, we show which model and data filling method the results were achieved with. The AdaBoost regressor achieved the overall best results and was used for most EFs. Most filling methods performed similarly, with no clear benefit of using a specific one. However, our custom filling was the most popular, with seven appearances, while mean, iterative, or KNN appeared three times each. We assume our custom method, which used -1, enabled our models to know the value was missing. Using methods that infer missing values may confuse the model by using wrong values, while using -1 may enable the model to infer it by itself. In future iterations of our work, it could be interesting to use different custom methods. It would also be interesting to use different filling methods for different features. Features that are only missing sporadically could use methods such as iterative, while mostly missing features could use our custom approach.

We also implemented ensemble learning using the specific features dataset. Table 22 shows

the results of ensemble learning. In this table, each normalization method is presented with and without ensemble learning. Similar to previous sections, ensemble learning did not seem to benefit the models. In fact, in most instances, the MSE was increased when using ensemble learning. Norm benefited from ensemble learning the most, with some EFs achieving lower MSE with ensemble learning. However, the reduction in MSE is not significant since it is only by a few units.

Feature	Specific			Ensemble		
	Non-Norm	Norm	Pre-Norm	Non-Norm	Norm	Pre-Norm
PR	0.11	0.18	0.24	0.10	0.16	0.24
NR	0.15	0.43	0.24	0.14	0.40	0.22
SR	0.08	0.13	0.27	0.07	0.11	0.18
WS	0.10	0.19	0.08	0.06	0.11	0.13
SFST	0.16	0.26	0.20	0.17	0.29	0.20
PR Benefit	0.01	0.01	0.01	0.23	0.01	0.0
NR Benefit	0.08	0.09	0.07	0.11	0.12	0.04
SR Benefit	0.07	0.09	0.01	0.12	0.09	0.02
WS Benefit	0.53	0.54	0.11	0.54	0.55	0.09
SFST Benefit	1.66	3.22	4.37	1.90	3.61	4.40

Table 22: Ensemble Learning for Specific Features

In the annex, figures 52 to 61 show the predictions with the classes. When looking at the overall behavior of these figures, we can see that all three normalizations performed well. PNorm achieved better results than Norm as expected, with PNorm being closer to the actual values. NNorm achieved better results than PNorm for functions, while it achieved slightly lower results for benefits. Figure 9 below shows the ensemble learning graph for WS. In this figure, we can see how well our algorithms performed using any normalization.

Most sites, using Norm or PNorm, were within an MSE score of one or less. This figure also demonstrates how good our regression predictions can be for class grouping. Table 23 shows the class grouping, while tables 41, 42 and 43 show the detailed results.

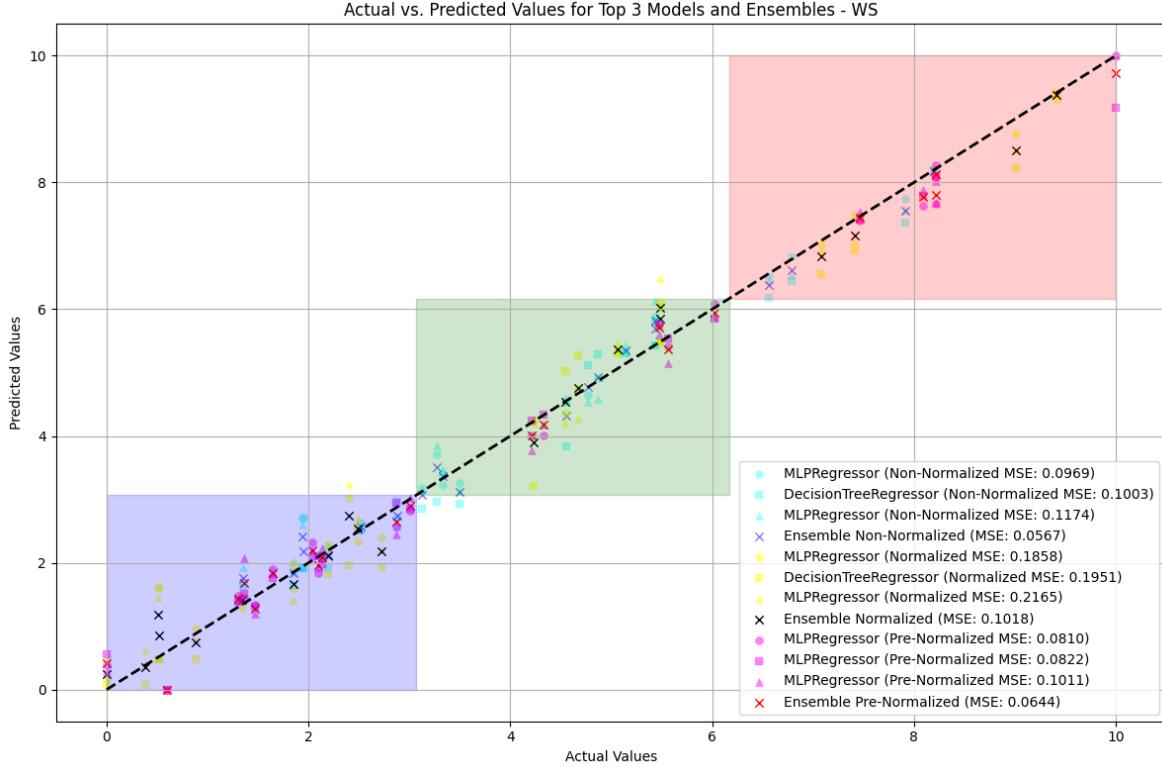


Figure 9: WS Graphs Specific Features

Function	NNorm		Norm		PNorm	
	Overall	Ensemble	Overall	Ensemble	Overall	Ensemble
PR	90.48%	92.86%	78.57%	76.19%	90.48%	90.48%
SR	85.71%	85.71%	80.95%	76.19%	92.86%	92.86%
NR	88.10%	85.71%	80.95%	71.43%	100.00%	95.24%
WS	90.48%	90.48%	97.62%	97.62%	95.24%	90.48%
SFST	90.48%	90.48%	88.10%	88.10%	97.62%	97.62%
PR Benefit	92.86%	90.48%	92.86%	92.86%	88.10%	88.10%
SR Benefit	92.86%	85.71%	90.48%	83.33%	100.00%	100.00%
NR Benefit	92.86%	92.86%	92.86%	92.86%	88.10%	88.10%
WS Benefit	78.57%	78.57%	80.95%	85.71%	92.86%	90.48%
SFST Benefit	71.43%	66.67%	52.38%	52.38%	50.00%	52.38%

Table 23: Summary of Overall and Ensemble Accuracy for Non-Normalized (NNorm), Normalized (Norm) and Pre-Normalized (PNorm) Data for Class Grouping

The overall results in Table 23 show promising results. NNorm performed very well when considering it does not normalize the data and the boundaries were generated using the normalization sites. In fact, it achieved better results on most EFs than the Norm data. PNorm achieved the best results for seven of the ten EFs, with the remaining three being close. In fact, NNorm, Norm and PNorm respectively achieved an average overall accuracy

of 87.083%, 83.472% and 89.426%. Impressively, NNorm achieved a classification accuracy of 71.43% on SFST Benefit compared to the second best of 52.38%. When excluding SFST Benefit, our data normalization techniques achieved 89.155%, 87.038% and 93.925% for NNorm, Norm and PNorm. These results are similar to using classification directly on the specific features. However, with regression, we are able to directly predict the score, which is highly beneficial.

5.2.2 Feature Reduction

Using the best performing model for each normalization technique, we performed feature reduction. We used the same three techniques as in the previous sections. Table 24 shows the best models with a maximum of five features.

EF	NNorm	Norm	Features	PNorm	Features
PR	0.2463	0.3916	F24, F28, F43, F44, F45	1.0651	F28, F43, F44, F45
SR	0.5499	0.9251	F28, F29, F31, F43, F44	1.3284	F28, F31, F44
NR	0.2879	0.7066	F1, F24, F43, F44, F45	0.0850	F43, F44
WS	0.0838	0.1665	F22, F28, F31, F43	0.8815	F22, F28, F31, F43, F44
SFST	0.2208	0.3715	F1, F14, F24, F43	0.6496	F1, F43
PR Benefit	0.7844	0.8554	F41, OF19, OF20, OF21, OF24	0.3265	F41, OF19, OF20, OF21, OF22
NR Benefit	2.6496	3.0701	F41, OF10, OF19, OF9	0.8918	F41, OF10, OF19, OF21, OF9
SR Benefit	2.5425	2.3806	F41, OF19	0.1807	F41, OF19, OF21
WS Benefit	0.6022	0.6119	F51, OF17, OF18, OF23, OF24	2.1603	F51, OF17, OF23, OF24, OF8
SFST Benefit	2.2402	4.3334	OF22, OF25, OF28	9.8386	F50, OF22, OF25, OF28

Table 24: Best MSE with Corresponding Features

In this table, we show each EF with both the NNorm and Norm MSE with their respective

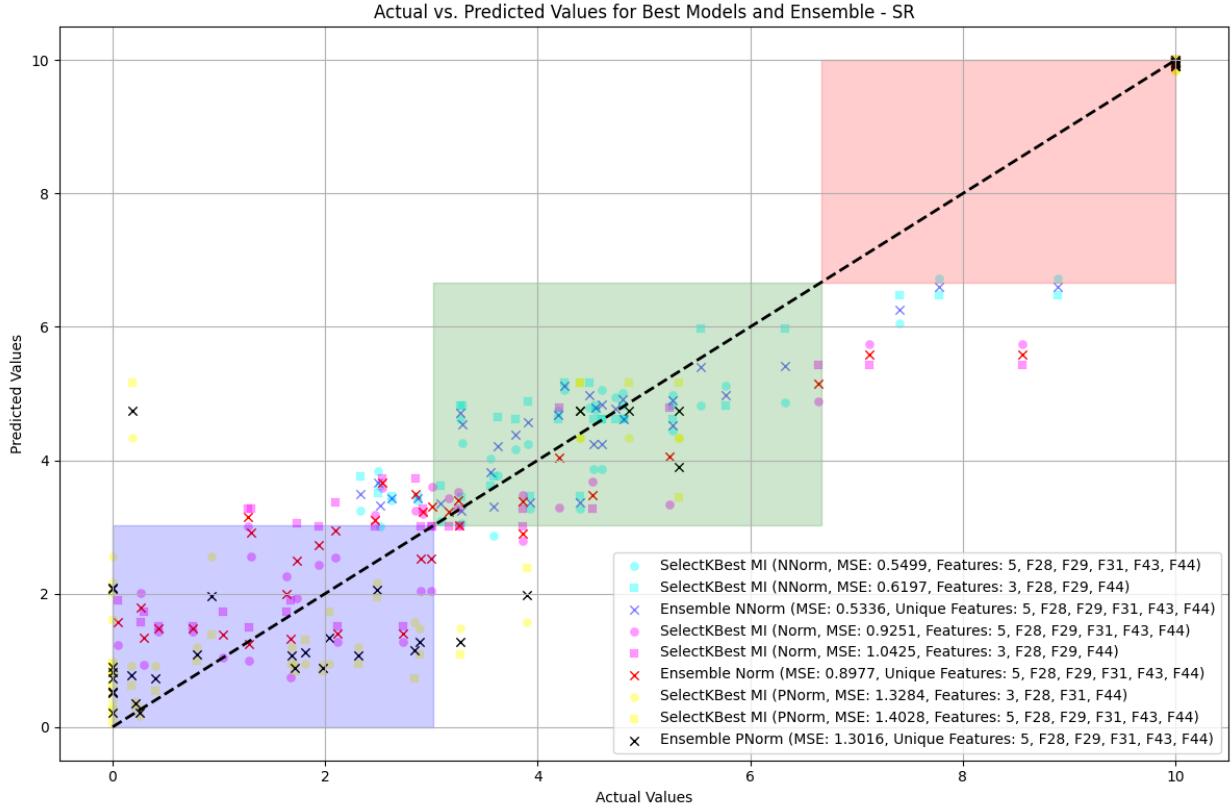


Figure 10: NR Graphs Specific Features

features, while PNorm shows the same. In comparison with using all specific features on FS, our reduced features achieved slightly higher MSEs. Our Benefit scores performed poorly using reduced features. Using NNorm, PR Benefit increased from 0.01 to 0.7844, while NR Benefit increased to 2.6496 from 0.08. Compared to feature reduction using all features, reduced specific features often achieved a higher MSE. In rare cases the MSE was decreased such as with WS using NNorm, D2 achieved an MSE of 0.0838 using four features, while using D4 it achieved 0.2692 with four features. This could be caused by the model having too much data, which hinders its ability to learn. Nonetheless, our overall results demonstrate that using specific features can be beneficial over using all features.

Our ensemble learning results did not increase our performance and often increased the MSE. In the annex, we show our ensemble learning results in figures 62 to 71. They show our validation predictions for the top two models and the ensemble approach for all three normalizations. A glaring issue is the normalized non-normalized predictions, which seem to have a larger error margin. This makes sense since they are the NNorm predictions scaled, which means the errors are also scaled. Nonetheless, we achieved good grouping, such as in figure 10, where three distinct clusters can be seen for Normand PNorm. A cluster near the lower-moderate boundary, near the moderate-higher boundary and in the higher class can be seen. This demonstrates that our approach can cluster regression predictions into classes well.

These results are confirmed in table 25, which shows a summary of our class grouping.

Function	NNorm		Norm		PNorm	
	Overall	Ensemble	Overall	Ensemble	Overall	Ensemble
WS	90.48%	90.48%	97.62%	97.62%	95.24%	90.48%
PR	90.48%	92.86%	78.57%	76.19%	90.48%	90.48%
NR	88.10%	85.71%	80.95%	71.43%	100.00%	95.24%
SR	85.71%	85.71%	80.95%	76.19%	92.86%	92.86%
SFST	90.48%	90.48%	88.10%	88.10%	97.62%	97.62%
PR Benefit	92.86%	90.48%	92.86%	92.86%	88.10%	88.10%
SR Benefit	92.86%	85.71%	90.48%	83.33%	100.00%	100.00%
NR Benefit	92.86%	92.86%	92.86%	92.86%	88.10%	88.10%
WS Benefit	78.57%	78.57%	80.95%	85.71%	92.86%	90.48%
SFST Benefit	71.43%	66.67%	52.38%	54.76%	50.00%	52.38%

Table 25: Summary of Overall and Ensemble Accuracy for Non-Normalized (NNorm), Normalized (Norm) and Pre-Normalized (PNorm) Data for Class Grouping

In this table, we show the overall classification accuracy for each normalization for both the top approach and ensemble. PNorm achieved the overall best accuracy of 89.39%, while NNorm and Norm achieved 87.38% and 83.33%, respectively. Compared to using all specific features, we slightly decreased the performance. Using all features, we achieved 89.426%, 87.083% and 83.472%, while we achieved 89.39%, 87.38% and 83.33% for PNorm, NNorm and Norm, respectively. To handle the SFST Benefit outlier problem, we compared the averages without it. PNorm achieved the best at 92.41%, while NNorm and Norm achieved 89.16% and 87.37%, respectively. These results were comparable to using all features, which achieved 93.925%, 89.155% and 87.138%. However, due to the significant decrease in features used, using reduced specific features achieved impressive results. We achieved similar results on all EFs while using a portion of the specific features.

Tables 44, 46 and 45 show detailed results. In these tables, we show the lower, moderate and higher accuracies of NNorm, Norm and PNorm, respectively. These tables provide us with information such as a lack of sites in the validation dataset. Having an equal number of sites for each rating would be beneficial.

Table 26 shows the best results for the Specific Features dataset.

Function	Model/Data	MSE	Accuracy	Features	Figures
PR	PNorm, Ad-aBoost, Iterative	1.0651	90.48%	F28, F43, F44, F5	62
SR	PNorm, MLP, Custom	1.3284	92.86%	F28, F31, F44	64
NR	PNorm, Ad-aBoost, Iterative	0.0850	100.00%	F43, F44	63
WS	Norm, MLP, Mode	0.1665	97.62%	F22, F28, F31, F43	65
SFST	PNorm, Ad-aBoost, Mean	0.6496	97.62%	F1, F43	66
PR Benefit	PNorm, MLP, Median	0.3265	92.86%	F41, OF19, OF20, OF21, OF22	67
SR Benefit	PNorm, DecisionTree, Median	0.1807	100.00%	F41, OF19, OF21	69
NR Benefit	PNorm, MLP, Custom	0.8918	92.86%	F41, OF9, OF10, OF19, OF21	68
WS Benefit	PNorm, DecisionTree, Mean	2.1603	92.86%	F51, OF8, OF17, OF23, OF24	70
SFST Benefit	NNorm, RANSAC, Interpolated	2.2402	71.43%	OF22, OF25, OF28	71

Table 26: Best Models Using Specific Features

In this table, we present which model for each EF achieved the best MSE and accuracy with reduced features. Excluding SFST Benefit, we achieved a range of accuracies from 90.48% to 100.00% using two to five features. This number of features is a considerable improvement from using all or even just specific features. NR, which usually requires 33 features, was reduced to using only two features to achieve 100.00% accuracy. Similarly, SR Benefit achieved an accuracy of 100.00% using only three features. These results are highly impressive since we only used ~200 sites for training. However, more training would be required since 100.00% should not be taken as a perfect model. Furthermore, using PNorm limits our models to use only sites that can be normalized using the same calibration sites. Our NNorm data also achieved good results with slightly lower accuracy with a similar number of features. These results demonstrate that we can use normalized, non-normalized and pre-normalized results. Each method has its benefits and can be used for specific situations depending on the requirements.

5.3 Dataset Comparison

To more easily compare the differences in results, we present the best-performing models for each EF for both classification and regression. The models that achieved the most interesting results are shown in Tables 26 and 20.

Function	MSE All	Acc. All	MSE Specific	Acc. Specific	Features
PR	0.3807	90.48%	1.0651	90.48%	F28, F43, F44, F45
SR	0.8303	88.10%	1.3284	92.86%	F28, F31, F44
NR	0.7361	85.71%	0.0850	100.00%	F43, F44
WS	0.4642	90.48%	0.1665	97.62%	F22, F28, F31, F43
SFST	0.5688	88.10%	0.6496	97.62%	F1, F43
PR Benefit	1.3627	90.48%	0.3265	92.86%	F41, OF19, OF20, OF21, OF22
SR Benefit	2.0620	95.24%	0.1807	100.00%	F41, OF19, OF21
NR Benefit	3.4702	92.86%	0.8918	92.86%	F41, OF9, OF10, OF19, OF21
WS Benefit	0.9022	80.95%	2.1603	92.86%	F51, OF8, OF17, OF23, OF24
SFST Benefit	2.3641	83.33%	2.2402	71.43%	OF22, OF25, OF28

Table 27: Comparison of Best Model Results Using All and Specific Features

Table 27 highlights the performance differences between models using all features versus specific features. Notably, the accuracy and MSE for several EFs, such as SR and SFST Benefit, show significant variation when using different feature sets. For instance, the NR function saw a substantial improvement in accuracy and a lower MSE when using specific features, whereas the PR Benefit and SR Benefit functions achieved better results with specific features compared to all features.

The performance drop in models using all features for some functions suggests that additional or irrelevant features might have introduced noise, making it harder for the models to learn effectively. On the other hand, the specific feature sets appear to better capture the essential characteristics needed for accurate predictions in certain cases.

This analysis underscores the importance of carefully selecting features based on the specific requirements of each EF. In future work, a more detailed investigation into the feature selection process could help to further refine the models and achieve even better results.

Furthermore, the use of different models and data filling methods, as indicated in the table, suggests that certain combinations are more effective for particular EFs. For instance, the use of AdaBoost with PNorm data consistently yielded strong performance across several functions.

Overall, while the use of all features can sometimes lead to better outcomes, specific features often provide more precise and reliable predictions, particularly when dealing with complex ecological functions.

6 Detailed Analysis of EFs using ML

In this section, we discuss our findings for different EFs separately. Table 28 presents the best results for both classification and regression using all and specific features.

Function	Type	All Features		Specific Features	
		Acc.	Features	Acc.	Features
PR	Class.	85.71%	F43, F45	85.71%	F43, F45
	Reg.	90.48%	F43, F44, F45	90.48%	F28, F43, F44, F5
SR	Class.	80.95%	Hydrogeo., F24, F29, F43, F44, F45	71.42%	OF22, F35, F43, F45, F49
	Reg.	88.10%	F24, F28, F29, F44	92.86%	F28, F31, F44
NR	Class.	80.95%	F43, F44	80.95%	F43, F44, F45
	Reg.	85.71%	F24, F43, F44, F45	100.00%	F43, F44
WS	Class.	90.45%	F43, F46	85.71%	F43, F44
	Reg.	90.48%	F22, F31, F43	97.62%	F22, F28, F31, F43
SFST	Class.	95.24%	F43, F44	95.24%	F1, F43
	Reg.	88.10%	F43, F44	97.62%	F1, F43
PR Benefit	Class.	90.45%	F14, F24, F41, F43, F46, F47	85.71%	OF22, F41
	Reg.	90.48%	F14, F41, F43, OF19	92.86%	F41, OF19, OF20, OF21, OF22
SR Benefit	Class.	100.00%	OF19, OF21, F41, F43	95.24%	OF19, F41
	Reg.	95.24%	F41, OF19, OF21	100.00%	F41, OF19, OF21
NR Benefit	Class.	90.45%	MossCover, OF9, OF10, OF19, F1, F14, F41, F43	90.48%	OF9, OF10, OF19, OF21, F41
	Reg.	92.86%	F14, OF9, OF10, OF19	92.86%	F41, OF9, OF10, OF19, OF21
WS Benefit	Class.	95.24%	OF17, OF23	95.24%	OF17, OF24, F51
	Reg.	80.95%	Hydrogeo., OF17, OF23, OF24	92.86%	F51, OF8, OF17, OF23, OF24
SFST Benefit	Class.	80.95%	F43, F44	57.14%	OF22, OF28
	Reg.	83.33%	F43, F44	71.43%	OF22, OF25, OF28

Table 28: Combined Summary of Accuracy and Features for Classification and Regression, Including Benefits

In this table, the most pertinent model for each EF is highlighted in bold. We show the classification results for each EF for classification and regression with both D2 and D4.

6.1 PR

Phosphorus retention is critical for the health and biodiversity of our planet. It reflects the ability of a wetland to retain phosphorus for long periods. Agricultural activities can increase the quantity of phosphorus in surrounding areas, negatively impacting wetlands. Monitoring the phosphorus retention capacity of a wetland provides valuable insights into land management practices.

PR is one of the most demanding EFs, requiring 22 features from the WESP-AC for the FS and 11 for FS. Reducing the number of features needed to evaluate a site could significantly aid researchers. When analyzing Table 28, PR shows promising results for FS. Both all features and specific features datasets achieved similar accuracy scores.

In classification, both datasets used F43 and F45, achieving an accuracy of 85.71%. For regression, F43, F44 and F45 were utilized to achieve 90.48%, with specific features requiring the addition of F28. These features (F43, F44, F45) respectively define the channel connection and outflow duration, outflow confinement and through-flow resistance. These WESP-AC features impact PR by indicating a wetland's ability to retain or drain water. F28, which measures surface water annual fluctuation, can indicate changes in water level, offering insights into PR. These four features have a significant influence on the results of PR FS.

Phosphorus retention offers substantial benefits to human populations by maintaining the quality of receiving waters. PR Benefit evaluates how beneficial a wetland's phosphorus retention is for humans. A site might be effective at phosphorus retention but not necessarily beneficial to humans and vice versa.

PR Benefit employs 11 features, sharing only one feature with PR: OF22. This distinction highlights the different focuses of FS and BS. The overview table shows that PR BS required more features than FS. The best-performing model required five features, while FS only needed three.

Regression using specific features yielded the best model with an accuracy of 92.86%, using F41, OF19, OF20, OF21 and OF22. F41 determines whether surface water enters the wetland, which is linked to F42 and F43 since it determines if F42 is skipped. OF19, OF20 and OF21 assess water quality, degraded water downstream and upstream, while OF22 determines the wetland's size relative to the catchment area. These features significantly influence BS as they reflect water quality, which is crucial for human benefit.

Interestingly, using all features, an accuracy of 90.48% was achieved with F14, F41, F43 and OF19. F14 and F43 were not originally used to calculate the score in the WESP-AC. F43, representing channel connection and outflow duration, is understandably important for BS. F14, defining the Sphagnum moss extent in the wetland, is notable as phosphorus improves

nitrogen uptake and retention in Sphagnum moss peat [13]. A larger extent of Sphagnum moss indicates better phosphorus retention, offering more human benefits.

This feature demonstrates the advantage of using all features rather than just specific ones. Using all features, we can uncover hidden relationships between features and scores. Overall, we achieved strong results for both FS and BS in classification and regression.

6.2 NR

Nitrate Removal & Retention is also vital for planetary health. Wetlands can remove nitrate by converting it into nitrogen gas with minimal nitrous oxide, a harmful greenhouse gas. This process, known as denitrification, is crucial for mitigating climate change. In the WESP-AC, NR FS uses 33 features, while BS employs 14.

Similar to PR, NR is challenging to predict. For FS, regression proved more effective than classification. Notably, an accuracy of 100.00% was achieved using F43 and F44 with regression on specific features. Interestingly, using all features, the accuracy dropped to 85.71%, suggesting that too many features can confuse the model.

In classification, models performed similarly, achieving 80.95% with F43 and F44 using all features, while specific features also included F45. F43 and F44 play a crucial role in NR, similar to PR, as they define water retention, filling and draining. F24, representing the area without surface water, influences nitrate retention, as areas with less surface water are less effective at retaining and removing nitrate.

Nitrate removal is also highly beneficial to humans by maintaining water quality. BS requires 14 features, with only one feature shared with FS—OF22. Both classification and regression models performed well across datasets. With specific features, regression achieved 92.86% accuracy using F41, OF9, OF10, OF19 and OF20. Similar to PR BS, F41 affects results by determining the wetland's water capacity.

OF9, OF10, OF19 and OF20 are part of the WESP-AC BS calculation, assessing proximity to population centers, distance to domestic wells, water quality and degraded upstream water. These features significantly affect human outcomes related to nitrate removal and retention. The WESP-AC provides more details on how these features impact human health.

Using all features, models performed similarly, achieving slightly better results. Regression with all features reached 92.86% accuracy using F14, OF9, OF10 and OF19. F14's inclusion is interesting, indicating its significance in BS, particularly since Sphagnum moss retains significant nitrate [13]. These results are further supported by the classification results using all features in Table 28.

Moss cover, one of the extra features collected outside the WESP-AC, emerged as an important feature. We demonstrate that NR and PR are quite similar when calculating their respective BS, which makes sense given their shared focus on nutrient retention and removal.

Future models combining data and predicting both scores together to reduce features would

be interesting. Our regression models for both FS and BS performed better than classification, achieving higher accuracy with fewer features while providing direct score predictions.

6.3 SR

Sediment Retention & Stabilization (SR) measures a wetland's effectiveness in intercepting and filtering inorganic sediments, enabling their deposition. It also reduces wave and current energy, helps prevent erosion and stabilizes underlying sediments and soil. This EF FS includes 19 features in the WESP-AC: three OF, 15 F and one S question.

Predicting FS for SR proved more challenging than for other FSs. Classification with specific features achieved only 71.42%, while all features reached 80.95%. However, regression with specific features yielded an accuracy of 92.86%, using F28, F31 and F44. Similar to PR and NR, F44 plays a significant role due to its definition of where major water runoff occurs. F28 and F31 represent the annual water fluctuation and the percentage of ponded water, respectively. These factors directly affect sediment retention, as more water can retain more sediment and a wetland with more ponded water is more effective at retaining and stabilizing sediments.

Using all features, F24 and F29 are included with F28 and F44. F29 defines the predominant depth of the wetland while F24 represents the area without surface water, both clearly affecting a wetland's ability to retain sediments. Overall, SR FS performs very well with regression, while the classification algorithms struggled.

For SR BS, the score shares only OF24 with FS, similar to PR and NR. Regression with specific features achieved the top accuracy of 100.00%, using F41, OF19 and OF21. These features—tributary channels, water quality and degraded water downstream—are also present in PR and NR for BS calculation. These three features, along with OF20, seem to provide enough information to determine the overall benefit of PR, NR and SR in a wetland.

Using D4 with classification also achieved 100.00% but with the addition of F43. F43, representing channel connection and outflow duration, is frequently used in predicting the benefit of an EF. It appears in all classification models using all features, except for WS BS. Nevertheless, OF19 and F41 seem to be the most important features for SR BS. Regarding FS and BS, both EFs used different features, sharing only F43 in common.

6.4 WS

Water Storage & Delay (WS) evaluates a wetland's effectiveness in storing water or delaying surface water movement, either for long or short periods. Water storage enhances a wetland's ability to perform its SR, NR and PR functions while also slowing erosion. It also helps control flooding by retaining water during heavy rain seasons.

WS FS uses 13 features, while its BS employs only six features, the fewest among all EFs. For FS, we achieved our best accuracy of 97.62% using specific features with regression,

employing F22, F28, F31 and F43. These features represent the percentage of the area seasonally flooded, the annual fluctuation range, the percentage of ponded water and channel connection & outflow duration. As with other EFs, F43 plays an important role in determining a wetland's water storage capacity. Similarly, F22, F28 and F31 are directly related to showing how often and how much the wetland is flooded, ponded or dry.

In classification, we achieved 90.45% using D4 with F43 and F46, while specific features achieved 85.71% with F43. Both F43 and F44 were expected since they are used to calculate the scores. F46, however, was not used in the WESP-AC and represents whether a wetland supports fish. This feature indirectly indicates whether a wetland retains water year-round or just seasonally. A wetland with fish likely retains water long enough to support their habitat. This provides enough information, combined with F43, to classify sites into three classes.

For BS, our top results were achieved using all features with classification, the only EF where the best results were obtained this way. Using all features, we reached 95.24% accuracy with OF17 and OF23, while specific features achieved 95.24% with OF17, OF24 and F51. Using only OF17 and OF23 is promising, as it suggests that we could accurately classify wetlands without the need for on-site visits, given that these are both OF features. OF17 indicates flood damage from non-tidal waters and OF23 represents the unvegetated surface in the contributing area. OF17 affects WS because flood damage suggests nearby human habitation and a higher storage capacity is beneficial for flood reduction. This feature provides substantial information, indicating how close humans live to a wetland. OF23 is also important because vegetation is crucial for water storage and erosion prevention. More vegetation means the wetland has greater WS capabilities.

For regression, we achieved 80.95% using the hydrogeomorphic class, OF17, OF23 and OF24 with all features, while specific features achieved 92.86% using F51, OF8, OF17, OF23 and OF24. The model with specific features remains highly promising, even though it required two more features and achieved slightly lower results. Nonetheless, this model would provide the direct score rather than just the class. Furthermore, it only uses F51, while the others are all OF, significantly reducing on-site time. F51, which represents the type of cover in a 30m radius (impervious or not), could be assessed remotely using satellite images, eliminating the need for on-site visits. This means we could predict BS using five OF features with an accuracy of 92.86%. Overall, the regression results for both FS and BS are highly promising, achieving very good accuracies with few features, particularly for BS, which uses only OF features.

6.5 SFST

Stream Flow Support (SFST) evaluates a wetland's effectiveness in contributing water to streams, particularly to drier parts of the area. SFST FS uses 14 features, while BS employs only six features. Interestingly, FS and BS share no common features, highlighting the distinct focuses of the function and benefit evaluations.

For FS, our best results were achieved using regression with specific features. We achieved

97.62% accuracy using only F1 and F43, while using all features, we reached 88.10% with F43 and F44. Similarly, classification using all features employed F43 and F44, while specific features used F1 and F43. As noted earlier, F43 and F44 are crucial for SFST, as they represent channel connection and outflow duration & outflow confinement. Interestingly, F44 is frequently used in SFST but is not present in the specific features. F1 describes the overall vegetation, water and soil, providing a quick assessment of the SFST function. Overall, we achieved impressive results in both classification and regression with either dataset, making SFST the EF with the overall best performance.

In terms of BS, we encountered some interesting results regarding the datasets used. When using specific features, neither our classification nor regression models performed well. Classification achieved 57.14% accuracy with OF22 and OF28, while regression reached 71.43% with OF22, OF25 and OF28. However, using all features, models performed better, with regression achieving 83.33% accuracy using F43 and F44. Neither of these features is used in the WESP-AC; only F42 is utilized. However, F43 and F44 represent channel connection & outflow duration and outflow confinement, which may influence the benefit of SFST. Classification with all features also used F43 and F44, achieving an accuracy of 80.95%.

The fact that SFST BS uses only six features may cause issues during training, as too many sites may be similar. Similar sites with different ratings can make it difficult for models to differentiate between them. Overall, FS performed very well, while we encountered difficulties in predicting BS.

7 CausalML

CausalML is a Python library that combines CI and ML. CI is the process of determining whether a cause-and-effect relationship exists between variables. Unlike correlation, which only indicates that two variables are related, causal inference seeks to understand whether changes in one variable directly cause changes in another. We aim to use CI to determine which features are more or less important to certain EFs. Researchers could greatly benefit from understanding the relationship between changes in features and their effects on the EF. CausalML provides tools for treatment effect estimation, propensity score modeling, uplift modeling and instrumental variables, integrating well with popular ML libraries like scikit-learn.

We propose training similar algorithms as in the classification and regression sections to determine the importance of features. We train the algorithms using the regression D1 dataset, which includes all features, for each EF. We compare each algorithm individually and globally using two methods to average the results. The first is the average importance across all methods, while the second is a custom point system. The average importance consists of adding all feature importance values and averaging them across the number of algorithms. In our point system, each feature is given a point based on its importance rank for each algorithm. A feature receives the maximum number of points for being the most important in a given algorithm, while it receives the minimum for being the least important.

7.1 Results

We present the results in tables and graphs for better visualization. Tables 47 to 56 present the results for NR, PR, SR, WS, SFST and their benefits respectively. Figures 72 to 81 present the graphical results for these functions.

Table 29 shows the most and least important features for each function.

Function	Top 4 (Avg.)	Bottom 4 (Avg.)	Top 4 (Points)	Bottom 4 (Points)
PR	'F43': 0.977784	'F5': -0.154281	'F43': 1402 points	'F68': 464 points
	'F22': 0.075298	'OF27': -0.046845	'OF22': 1231 points	'F6': 448 points
	'F49': 0.044496	'F6': -0.021769	'F22': 1223 points	'F9': 436 points
	'F45': 0.042024	'OF24': -0.011727	'F25': 1221 points	'OF24': 412 points
NR	'F43': 0.612435	'OF33': -0.006181	'F43': 1343 points	'F32': 377 points
	'F45': 0.117268	'OF6': -0.005082	'F45': 1220 points	'F67': 360 points
	'F44': 0.081121	'F33': -0.004765	'F44': 1151 points	'F33': 358 points
	'F5': 0.053705	'F8': -0.004219	'F3f': 1137 points	'F39': 352 points
WS	'F43': 0.951765	'F12': -0.026529	'F43': 1346 points	'F68': 451 points
	'F5': 0.097695	'F18': -0.020177	'F22': 1265 points	'F18': 439 points
	'F22': 0.091916	'F62': -0.009468	'OF22': 1214 points	'OF8': 398 points
	'F52': 0.048698	'OF20': -0.008094	'F25': 1178 points	'F62': 387 points
SR	'F43': 0.563853	'F58': -0.013924	'F43': 1365 points	'F21': 426 points
	'F45': 0.320945	'OF38': -0.021798	'F45': 1260 points	'S2': 384 points
	'F24': 0.140763	'F3c': -0.026023	'F31': 1135 points	'OF34': 377 points
	'OF21': 0.114374	'OF31': -0.028798	'F24': 1130 points	'OF2': 255 points
SFST	'F43': 1.039090	'F4': -0.014221	'F43': 1305 points	'OF8': 399 points
	'F5': 0.497852	'OF9': -0.019349	'F1': 1149 points	'F58': 378 points
	'OF27': 0.122599	'F36': -0.023610	'F34': 1093 points	'F49': 330 points
	'F45': 0.035529	'F35': -0.027724	'F47': 1076 points	'F3g': 300 points
PR Benefit	'F41': 0.558198	'OF20': -0.008562	'F41': 1351 points	'F20': 424 points
	'OF19': 0.331069	'F48': -0.004729	'OF19': 1285 points	'F13': 472 points
	'OF21': 0.124913	'OF16': -0.003171	'OF21': 1119 points	'F33': 478 points
	'F5': 0.023782	'OF7': -0.003062	'OF24': 1037 points	'OF25': 478 points
NR Benefit	'OF10': 0.417809	'OF27': -0.065078	'OF10': 1342 points	'F36': 356 points
	'OF19': 0.312239	'F3g': -0.014606	'OF19': 1236 points	'OF5': 358 points
	'OF21': 0.205786	'OF25': -0.008854	'F41': 1198 points	'OF3': 372 points
	'F41': 0.075133	'OF6': -0.007846	'OF21': 1189 points	'F25': 433 points
SR Benefit	'OF19': 0.497638	'OF18': -0.002347	'OF19': 1294 points	'F20': 269 points
	'F41': 0.294398	'F20': -0.001792	'F41': 1231 points	'F68': 354 points
	'OF21': 0.268613	'OF5': -0.001777	'OF21': 1143 points	'S2': 407 points
	'F12': 0.045374	'OF22': -0.001403	'F1': 1056 points	'S1': 430 points
WS Benefit	'OF17': 0.671066	'F5': -0.651739	'OF17': 1307 points	'F5': 323 points
	'F53': 0.620027	'F3d': -0.161077	'F52': 1237 points	'F33': 369 points
	'OF20': 0.235132	'F3c': -0.145086	'OF23': 1212 points	'F67': 376 points
	'OF21': 0.218170	'F33': -0.103096	'F51': 1189 points	'F64': 381 points
SFST Benefit	'F43': 0.755861	'F3a': -0.025738	'F43': 1158 points	'F20': 362 points
	'F45': 0.141279	'OF13': -0.020652	'OF18': 1122 points	'F22': 366 points
	'F5': 0.083083	'F3d': -0.015819	'F1': 1100 points	'S2': 416 points
	'OF28': 0.061002	'F3e': -0.012449	'OF28': 1080 points	'F62': 425 points

Table 29: Most and Least Important Features for Various Models

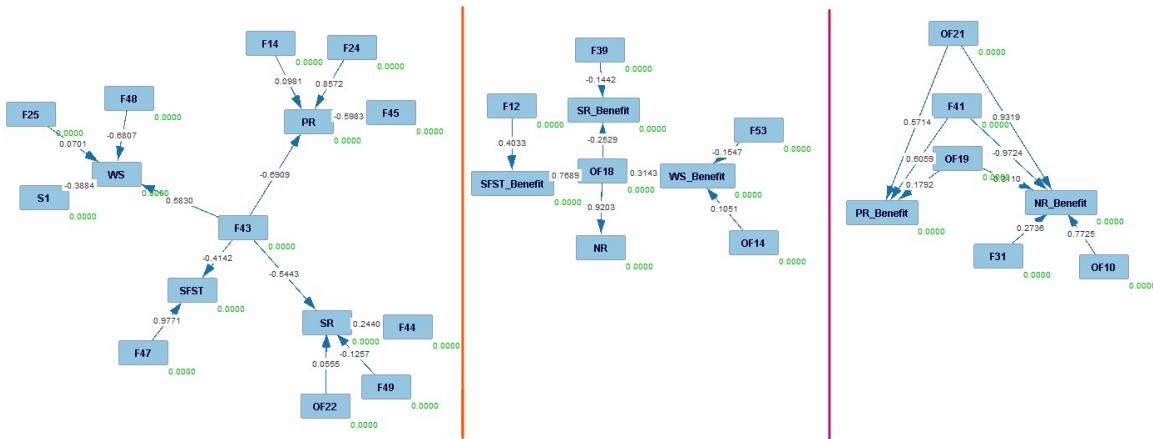
7.2 Discussion

CausalML is a Python library that implements CI and ML. It is often used to determine the effectiveness of a treatment for a specific cause-effect relationship. However, by setting each feature as a treatment, we can determine how important a specific feature is to each EF. Tables 47 to 56 and Figures 72 to 81 present the numerical and graphical results, respectively. Table 29 shows the most and least important features for each function. We also present the average importance and a voting system in the tables. The voting system consists of providing points to each feature proportionate to how important they are when using an algorithm. This voting system was implemented to prevent a specific algorithm from disproportionately influencing the average importance. The figures present both of these metrics combined for each EF. They demonstrate the importance of using both methods, as some average importance values are very small but have high point totals. Similar to our previous results using ML, F43 appears to be highly important for most EFs.

8 Tetrads

Tetrad is CI modeling software that is widely used to discover causal relationships between variables. We employed this software to try to determine causal relationships between features for specific EFs. Our goal was to utilize the features and non-normalized scores to determine these relationships. However, after testing various algorithms, parameters and more, we discovered that no model was able to fit the data. An example of the graphical results can be seen in Figure 11.

After multiple iterations of Tetrad yielding no results, we turned our efforts towards ML. Further exploration with CI or Tetrad specialists could be beneficial for this aspect of the project.



9 Future Directions

9.1 Data

This project presents several opportunities for future improvement, particularly in the area of data management and utilization. A more detailed analysis of the existing dataset could greatly enhance both our models and our understanding of the underlying processes. By thoroughly analyzing the input variables and output possibilities, we can gain deeper insights into the behavior of certain results and model performance.

Expanding the dataset by incorporating new data or employing data augmentation techniques to increase the number of sites could significantly benefit our models. Introducing data from different provinces or countries would enhance the generalizability and robustness of the models, making them more reliable across diverse regions. Since visiting sites for data collection can be resource-intensive, leveraging calibration sites during training could provide a practical and efficient alternative.

Exploring different normalization techniques alongside model training could also yield promising results. Currently, the normalization process is based on minimum and maximum values, which sometimes causes new sites to fall outside the expected bounds. Implementing alternative normalization methods, such as Z-score normalization or percentile-based scaling, could ensure that new sites remain within appropriate bounds, thereby improving model accuracy and applicability.

Incorporating additional features could further refine our models by reducing the number of input variables required. Features such as provincial and federal classifications have already shown potential in improving the prediction of ecosystem function (EF) scores and ratings. Additionally, integrating historical data could enable long-term tracking of wetland performance, providing valuable insights into temporal changes and trends.

Future work could also explore the integration of remote sensing data to complement field data. This could reduce the need for on-site visits and provide a more comprehensive view of wetland conditions over time. Moreover, developing a dynamic dataset that updates in real-time as new data becomes available would allow for continuous model improvement and more accurate predictions.

9.2 Machine Learning

There are multiple avenues for advancing the machine learning algorithms used in this project, particularly in classification and regression tasks. One promising direction is the implementation of more sophisticated neural network models. Currently, we employ a simple neural network with a limited number of layers, which leaves considerable room for enhancement. Incorporating deeper architectures, experimenting with different types of layers (such as convolutional, recurrent, or attention mechanisms) could be interesting. Furthermore,

exploring various methods of layer concatenation and architectural design could significantly boost model performance.

In future work, it would be beneficial to explore different ensemble learning techniques, such as a voting system. Techniques such as bagging, boosting, or stacking could be investigated to determine their efficacy in predicting EF scores. Additionally, implementing transfer learning, where a pre-trained model on a large dataset is fine-tuned on our specific data, could accelerate the training process and improve model accuracy, especially since the available data is limited.

Hyperparameter optimization techniques, other than grid search, such as random search or Bayesian optimization, could be employed to fine-tune model parameters. This could potentially lead to further improvements in performance. Furthermore, incorporating advanced algorithms like Gradient Boosting Machines (GBM) or Extreme Gradient Boosting (XGBoost) could be explored as alternatives or complements to neural networks.

9.3 Causal Inference

While much of our work has focused on machine learning, there are several potential improvements for causal inference (CI) that could be explored. First, incorporating additional algorithms and expanding the dataset used in CausalML could improve the quality and reliability of our results. A thorough, in-depth analysis of these results could further enhance our understanding and interpretation of the causal relationships at play.

For the Tetrad software, our initial attempts did not yield conclusive results. Engaging with Tetrad experts to review our methodology and findings could be highly beneficial. Expert consultation could help us validate our approach, refine our methods, and potentially unlock new insights that were previously inaccessible.

Future work could also involve integrating causal inference with machine learning models to create hybrid models that leverage the strengths of both approaches. For example, causal inference could be used to identify the most influential features, which could then be used to inform feature selection in machine learning models. Additionally, exploring other causal inference frameworks and tools, such as DoWhy or Pyro, could provide alternative perspectives and methodologies that may yield better results.

Moreover, developing a more comprehensive causal model that accounts for potential confounders and interactions between variables could enhance the robustness of our findings. Finally, validating the causal relationships identified through field experiments or observational studies could provide empirical support for the model predictions and strengthen the overall conclusions of the study.

10 Conclusion

This report presented a detailed analysis of various ecosystem functions (EFs) using a combination of classification and regression models, integrated with causal inference techniques and ensemble learning methods. Our objective was to evaluate the effectiveness of different feature sets in predicting EF scores, to assess the potential of specific features versus all available features and to explore the causal relationships between features and their effects on the EFs using advanced tools like CausalML and Tetrad. Our study began by outlining the importance of ecosystem functions, which are critical to understanding the health and sustainability of natural environments. The accurate prediction and assessment of these functions are vital for environmental management, conservation efforts and policy-making. We employed various machine learning techniques to predict the scores of different EFs, including Phosphorus Retention (PR), Nitrate Removal & Retention (NR), Sediment Retention & Stabilization (SR), Water Storage & Delay (WS) and Stream Flow Support (SFST). These EFs were analyzed not only in terms of their direct functions but also their benefits (BS) to human populations.

10.1 Classification Analysis

We began with the classification of EFs, where we compared the performance of models trained on all features versus those trained on a reduced set of specific features. The classification models aimed to categorize sites based on their EF scores into predefined classes, enabling us to determine how well the models could generalize across different datasets. Our results demonstrated that while classification using all features generally outperformed the models trained on specific features, certain EFs, such as PR and WS, showed robust performance even with a reduced feature set. This indicates that for these functions, key features carry significant predictive power, allowing for more efficient modeling with fewer data points. However, for other EFs like SR and SFST, the reduction in features led to a noticeable drop in classification accuracy, highlighting the importance of maintaining a comprehensive set of features for complex functions. The results also revealed that ensemble learning techniques did not universally enhance model performance, with some EFs benefiting from ensemble methods while others experienced reduced accuracy.

10.2 Regression Analysis

Following the classification analysis, we delved into regression models to predict the exact EF scores. Regression provided a more nuanced understanding of the relationships between features and EF outcomes, allowing us to generate continuous predictions rather than categorical classifications. The regression models, similar to the classification models, were tested on both the full set of features and the reduced specific features dataset. Our findings indicated that regression models generally performed better when utilizing all features, particularly for complex EFs such as SR and SFST, where the comprehensive dataset allowed for

more accurate predictions. Interestingly, for certain EFs like PR and NR, the models trained on specific features performed comparably to those trained on all features, suggesting that the most influential features for these functions were included in the reduced dataset. Additionally, our analysis showed that regression models often outperformed classification models in terms of predictive accuracy, making them more suitable for detailed EF assessments.

10.3 Phosphorus Retention (PR)

Phosphorus Retention is a crucial EF as it plays a significant role in maintaining the ecological balance by preventing excess phosphorus from entering water bodies, which can lead to eutrophication. Our analysis showed that PR performed well with both classification and regression models, especially when using specific features. Key features such as F43 (Channel Connection and Outflow Duration), F44 (Outflow Confinement) and F45 (Through Flow Resistance) consistently appeared as important predictors, demonstrating their significance in phosphorus retention. These features, which are related to the wetland's hydrological connectivity and water retention capacity, directly influence the ability of a wetland to retain phosphorus over time. Interestingly, when focusing on PR Benefit, which evaluates the human benefits of phosphorus retention, additional features such as OF19 (Water Quality Downstream) and OF22 (Wetland Size Relative to Catchment) became crucial. This distinction between function and benefit underscores the complexity of environmental modeling, where the ecological role of a feature might differ from its perceived human value. Regression models, particularly those using specific features, provided high accuracy, suggesting that PR can be effectively predicted with a carefully selected subset of features, enabling more efficient environmental assessments.

10.4 Nitrate Removal & Retention (NR)

Nitrate Removal & Retention is another vital EF, particularly in its role in reducing nitrogen pollution through the process of denitrification. Our study revealed that NR is more challenging to predict than PR, requiring a broader set of features to achieve high accuracy. However, similar to PR, features related to water retention and connectivity, such as F43, F44 and F45, were consistently important across models. The regression models showed that NR could be predicted with high accuracy when using these key features, especially with specific feature sets. For NR Benefit, which considers the impact on human populations, the most significant features included those related to water quality and proximity to human activities, such as OF9 (Distance to Domestic Wells) and OF10 (Nearest Population Center). This indicates that human-related factors play a significant role in determining the benefit of nitrate retention, highlighting the need for models that integrate both environmental and human dimensions. The successful prediction of NR Benefit using regression models with a reduced feature set demonstrates the potential for efficient monitoring and management practices.

10.5 Sediment Retention & Stabilization (SR)

Sediment Retention & Stabilization is crucial for preventing soil erosion, maintaining water quality and supporting aquatic habitats. Our analysis found that SR was particularly challenging to predict, especially with classification models using specific features, which often resulted in lower accuracy compared to other EFs. Key features such as F28 (Annual Water Fluctuation), F31 (Percentage of Ponded Water) and F44 were vital in predicting SR, indicating that the ability of wetlands to stabilize sediments is closely tied to their hydrological characteristics. The regression models provided better predictions for SR, especially when using all available features, which suggests that sediment retention is a complex process influenced by multiple factors. For SR Benefit, features related to water quality and hydrological connectivity, such as OF19 and F41 (Tributary Channels), were essential, reflecting the importance of both environmental and human-related factors in determining the benefits of sediment retention. The mixed results for SR highlight the complexity of this function and suggest that future models may need to incorporate more comprehensive datasets or advanced modeling techniques to improve accuracy.

10.6 Water Storage & Delay (WS)

Water Storage & Delay is a fundamental EF that helps to mitigate flooding, sustain base flows during dry periods and support other EFs like sediment retention and nutrient removal. Our findings indicated that WS was one of the most predictable EFs, with regression models achieving high accuracy, particularly when using specific features. Key features for WS included F22 (Percentage of Area Seasonally Flooded), F28 and F43, all of which are directly related to the wetland's capacity to store and manage water. These results suggest that WS is strongly influenced by the physical and hydrological properties of the wetland, making it a relatively straightforward function to model. Interestingly, the WS Benefit could be accurately predicted using only a few OF features, such as OF17 (Flood Damage from Non-Tidal Waters) and OF23 (Unvegetated Surface in the Contributing Area). This suggests that for WS, it may be possible to conduct effective assessments with minimal field data, relying instead on observable or remotely sensed information. The ability to predict WS Benefit with high accuracy using limited features highlights the potential for simplifying environmental assessments, particularly in large-scale or remote regions where field data collection is challenging.

10.7 Stream Flow Support (SFST)

Stream Flow Support is essential for maintaining base flows in streams, especially during dry periods and for supporting aquatic life. Our analysis showed that SFST was one of the most challenging EFs to predict, particularly its benefit to human populations. The SFST function itself was best predicted using features like F1 (Overall Vegetation, Water and Soil) and F43, which are indicative of the wetland's ability to contribute water to streams. However, the SFST Benefit was difficult to model accurately, especially when using

specific features, with significant drops in accuracy observed. This suggests that the factors influencing SFST Benefit are more complex and perhaps less directly observable than those influencing the SFST function. The disparity in model performance between the SFST function and its benefit highlights the need for further research into the drivers of stream flow support, particularly in understanding how human activities and land-use changes affect this critical function.

10.8 Causal Inference with CausalML

To further enhance our understanding of feature importance and the underlying relationships between features and EFs, we employed CausalML, a Python library that integrates causal inference with machine learning. This approach allowed us to move beyond correlation and explore causality, identifying which features have a direct impact on EF outcomes. Through CausalML, we were able to rank the features based on their importance, using both average importance scores and a custom point system. The results reinforced our earlier findings, with features such as F43 consistently emerging as crucial across multiple EFs. The causal analysis provided valuable insights into the mechanics of each EF, highlighting the features that drive the most significant changes in EF scores. This understanding is crucial for environmental management, as it allows for more targeted interventions and resource allocation.

10.9 Challenges with Tetrad

In an attempt to explore the causal relationships further, we utilized Tetrad, a software tool designed for discovering causal relationships between variables. Despite multiple iterations and adjustments to the models, Tetrad did not yield any significant results that could be integrated into our analysis. The lack of fitting models led us to reconsider our approach and focus more on the insights provided by CausalML. Future work in this area could benefit from collaboration with specialists in causal inference to fully leverage Tetrad's capabilities.

10.10 Overall Conclusion

In conclusion, this report has provided a comprehensive analysis of ecosystem functions using advanced machine learning techniques and causal inference tools. Our findings underscore the importance of selecting the right set of features for accurate predictions, particularly when dealing with complex environmental functions like SR and SFST. While feature reduction can streamline models and reduce data requirements, it is not without risks, as the exclusion of important variables can significantly impact model performance. The integration of causal inference techniques like CausalML offers a promising avenue for better understanding the dynamics at play within these ecosystems, providing a more informed basis for environmental decision-making. Moving forward, future research should continue to explore the balance between model complexity and accuracy, the potential of ensemble

methods and the application of causal inference in environmental science. By doing so, we can improve our ability to model, understand and ultimately protect the vital ecosystem functions that sustain our planet.

11 References

- [1] W. J. Junk, M. Brown, I. C. Campbell, M. Finlayson, B. Gopal, L. Ramberg, and B. G. Warner, “The comparative biodiversity of seven globally important wetlands: a synthesis,” *Aquatic Sciences*, vol. 68, pp. 400–414, 2006.
- [2] A. Pott, A. K. Oliveira, G. A. Damasceno-Junior, and J. S. Silva, “Plant diversity of the pantanal wetland,” *Brazilian journal of biology*, vol. 71, pp. 265–273, 2011.
- [3] The Wetlands Conservancy, “Animals,” 2024, accessed: 2024-07-30. [Online]. Available: <https://www.wetlandsconservancy.org/animals>
- [4] I. Kačergytė, D. Arlt, Å. Berg, M. Žmihorski, J. Knape, Z. M. Rosin, and T. Pärt, “Evaluating created wetlands for bird diversity and reproductive success,” *Biological Conservation*, vol. 257, p. 109084, 2021.
- [5] R. K. Das, “Fish diversity and the conservation status of a wetland of cooch behar district, west bengal, india,” *Journal of Threatened Taxa*, vol. 10, no. 3, pp. 11 423–11 431, 2018.
- [6] J. Grime, “Benefits of plant diversity to ecosystems: immediate, filter and founder effects,” *Journal of Ecology*, vol. 86, no. 6, pp. 902–910, 1998.
- [7] Washington State Department of Ecology, “Wetlands and climate change,” 2024, accessed: 2024-07-30. [Online]. Available: <https://ecology.wa.gov/water-shorelines/wetlands/tools-resources/wetlands-climate-change#:~:text=Wetlands%20are%20vulnerable%20to%20climate%20change&text=Coastal%20wetlands%20will%20also%20be,manage%20water%20quality%20and%20flooding>.
- [8] M. Alldred, S. B. Baines, and S. Findlay, “Effects of invasive-plant management on nitrogen-removal services in freshwater tidal marshes,” *PloS one*, vol. 11, no. 2, p. e0149813, 2016.
- [9] D. P. Batzer, C. R. Jackson, and M. Mosner, “Influences of riparian logging on plants and invertebrates in small, depressional wetlands of georgia, usa,” *Hydrobiologia*, vol. 441, pp. 123–132, 2000.
- [10] F. Kleinschroth and J. R. Healey, “Impacts of logging roads on tropical forests,” *Biotropica*, vol. 49, no. 5, pp. 620–635, 2017.
- [11] D. Mao, Z. Wang, J. Wu, B. Wu, Y. Zeng, K. Song, K. Yi, and L. Luo, “China’s wetlands loss to urban expansion,” *Land degradation & development*, vol. 29, no. 8, pp. 2644–2657, 2018.

- [12] C. Li, H. Wang, X. Liao, R. Xiao, K. Liu, J. Bai, B. Li, and Q. He, "Heavy metal pollution in coastal wetlands: A systematic review of studies globally over the past three decades," *Journal of Hazardous Materials*, vol. 424, p. 127312, 2022.
- [13] B. Williams, D. Silcock, and M. Young, "Nitrogen-phosphorus interactions in peat," *Écologie*, vol. 30, no. 1, p. 19, 1999.

12 Annexes

12.1 Datasets

12.2 Features

Feature Name	Type	Description
Provincial Class	Class (6)	This classification divides wetlands into six distinct types based on their ecological characteristics. It aids in the management and conservation of wetland resources.
Provincial Class	Class (10)	This extended classification categorizes wetlands into ten types, providing a more detailed understanding of wetland diversity for conservation strategies.
Water Regime	Class (4)	This indicator classifies water regimes into four types, describing the hydrological conditions of a wetland. It is essential for wetland management.
Vegetation Type	Class (7)	This classification identifies seven types of specific vegetation present in wetlands. It helps in assessing the ecological status of wetland areas.
Veg. Cover	Class (5)	This classification indicates the percentage cover of specific vegetation types, divided into five classes. It provides insights into vegetation distribution.
Woody Canopy	Class (4)	This classification describes the percentage cover of high woody canopy (over 5 meters), divided into four classes. It is useful for understanding forest structure in wetlands.
% Moss Cover	Class (5)	This classification specifies the percentage cover of moss in wetlands, divided into five classes. It is important for ecological assessments.
Phragmites	Class (3)	This feature indicates the presence of Phragmites (an invasive plant species) in a wetland, classified as yes or no. It is crucial for invasive species management.
Soil Type	Class (3)	This classification identifies the soil types in wetlands, divided into three classes. It helps in understanding soil characteristics and their impact on wetland ecology.
Surface Water	Class (7)	This classification indicates the percentage of surface water present in a wetland, divided into seven classes. It is important for hydrological studies.
Depth Sat.	Class (4)	This feature measures the depth of water saturation in soil, divided into four classes. It helps in understanding soil moisture conditions.
Depth Moss	Float	This feature measures the average depth of living moss in centimeters. It provides insights into the health and growth of moss in wetlands.
Organic Depth	Float	This feature measures the average depth of organic material in soil in centimeters. It is crucial for understanding soil composition and fertility.
Hydrogeomorphic	Class (11)	This classification divides wetlands into eleven hydrogeomorphic classes, based on their geomorphology and hydrology. It is essential for wetland characterization and management.

Table 30: Feature Descriptions for Various Wetland Attributes

12.3 Machine Learning Algorithms

Algorithm	Type	Parameters
Ridge	Class./Reg.	<ul style="list-style-type: none"> • <code>alpha</code>: [0.1, 0.5, 1.0] • <code>solver</code>: ['auto', 'svd', 'cholesky', 'lsqr', 'sparse_cg', 'sag', 'saga', 'lbfgs']
Decision Tree	Class./Reg.	<ul style="list-style-type: none"> • <code>criterion</code>: ['gini', 'entropy', 'log_loss'] (Class.), ['squared_error', 'friedman_mse', 'absolute_error', 'poisson'] (Reg.) • <code>splitter</code>: ['best', 'random'] • <code>min_samples_split</code>: [2, 3, 4, 5] • <code>max_features</code>: [None, 'sqrt', 'log2']
Random Forest	Class./Reg.	<ul style="list-style-type: none"> • <code>n_estimators</code>: [50, 100, 200] • <code>criterion</code>: ['gini', 'entropy', 'log_loss'] (Class.), ['squared_error', 'friedman_mse', 'absolute_error', 'poisson'] (Reg.) • <code>min_samples_split</code>: [2, 5] • <code>max_features</code>: ['sqrt', 'log2']
Gradient Boosting	Class./Reg.	<ul style="list-style-type: none"> • <code>loss</code>: ['log_loss', 'deviance', 'exponential'] (Class.), ['squared_error', 'absolute_error', 'huber', 'quantile'] (Reg.) • <code>learning_rate</code>: [0.001, 0.01, 0.1] • <code>n_estimators</code>: [50, 100, 200] • <code>warm_start</code>: [True, False]
AdaBoost	Class./Reg.	<ul style="list-style-type: none"> • <code>n_estimators</code>: [50, 100, 200] • <code>learning_rate</code>: [0.001, 0.01, 0.1, 1.0] • <code>algorithm</code>: ['SAMME', 'SAMME.R'] (Class.), <code>loss</code>: ['linear', 'square', 'exponential'] (Reg.)

K-Neighbors	Class./Reg.	<ul style="list-style-type: none"> <code>n_neighbors</code>: [5, 10, 15, 20] <code>weights</code>: ['uniform', 'distance'] <code>algorithm</code>: ['auto', 'ball_tree', 'kd_tree', 'brute'] <code>leaf_size</code>: [30, 50, 70] <code>metric</code>: ['euclidean', 'manhattan', 'minkowski']
MLP (Neural Network)	Class./Reg.	<ul style="list-style-type: none"> <code>hidden_layer_sizes</code>: [(50, 50, 50), (100, 100, 100), (100, 100, 100, 100)] <code>activation</code>: ['identity', 'logistic', 'tanh', 'relu'] <code>solver</code>: ['lbfgs', 'sgd', 'adam'] <code>learning_rate</code>: ['constant', 'invscaling', 'adaptive']
Logistic Regression	Classifier	<ul style="list-style-type: none"> <code>penalty</code>: ['l1', 'l2', 'elasticnet', 'none'] <code>C</code>: [0.1, 0.5, 1.0, 5.0, 10.0] <code>solver</code>: ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'] <code>max_iter</code>: [100, 200, 300]
SGD	Class./Reg.	<ul style="list-style-type: none"> <code>loss</code>: ['hinge', 'log', 'modified_huber', 'squared_hinge', 'perceptron'] (Class.), ['squared_error', 'huber', 'epsilon_insensitive', 'squared_epsilon_insensitive'] (Reg.) <code>penalty</code>: ['l2', 'l1', 'elasticnet'] <code>learning_rate</code>: ['constant', 'optimal', 'invscaling', 'adaptive'] <code>warm_start</code>: [True, False]
Support Vector Machines (SVM)	Class./Reg.	<ul style="list-style-type: none"> <code>C</code>: [0.1, 1.0, 10.0] <code>kernel</code>: ['linear', 'poly', 'rbf', 'sigmoid'] <code>degree</code>: [1, 3, 5] <code>gamma</code>: ['scale', 'auto']

Gaussian Naive Bayes	Classifier	<ul style="list-style-type: none"> <code>var_smoothing</code>: [1e-9, 1e-8, 1e-7]
Linear Discriminant Analysis	Classifier	<ul style="list-style-type: none"> <code>solver</code>: ['svd', 'lsqr', 'eigen'] <code>shrinkage</code>: [None, 'auto', 0.1, 0.5, 1.0]
ElasticNet	Regressor	<ul style="list-style-type: none"> <code>l1_ratio</code>: [0.25, 0.5, 0.75] <code>fit_intercept</code>: [True, False] <code>precompute</code>: [True, False] <code>copy_X</code>: [True, False] <code>warm_start</code>: [True, False] <code>positive</code>: [True, False] <code>selection</code>: ['cyclic', 'random']
Bayesian Ridge	Regressor	<ul style="list-style-type: none"> <code>alpha_1</code>: [1e-7, 1e-6, 1e-5] <code>alpha_2</code>: [1e-7, 1e-6, 1e-5] <code>lambda_1</code>: [1e-7, 1e-6, 1e-5] <code>lambda_2</code>: [1e-7, 1e-6, 1e-5]
Kernel Ridge	Regressor	<ul style="list-style-type: none"> <code>alpha</code>: [0.00001, 0.0001, 0.001, 0.01] <code>kernel</code>: ['linear', 'poly', 'rbf', 'sigmoid', 'precomputed'] <code>degree</code>: [1, 2, 3, 5, 10] <code>coef0</code>: [0.0, 0.5, 1.0]
Linear Regression	Regressor	<ul style="list-style-type: none"> <code>fit_intercept</code>: [True, False] <code>copy_X</code>: [True, False] <code>positive</code>: [True, False]
RANSAC Regressor	Regressor	<ul style="list-style-type: none"> <code>min_samples</code>: [None, 1, 2, 5, 10] <code>max_trials</code>: [1, 10, 50, 100, 150] <code>loss</code>: ['absolute_error', 'squared_error']

Theil-Sen Regressor	Regressor	<ul style="list-style-type: none"> • <code>max_subpopulation</code>: [1, 10, 100, 500] • <code>n_subsamples</code>: [None, 1, 5, 10, 25]
------------------------	-----------	--

Table 31: Machine Learning Algorithms

12.4 Classification

12.4.1 All Features

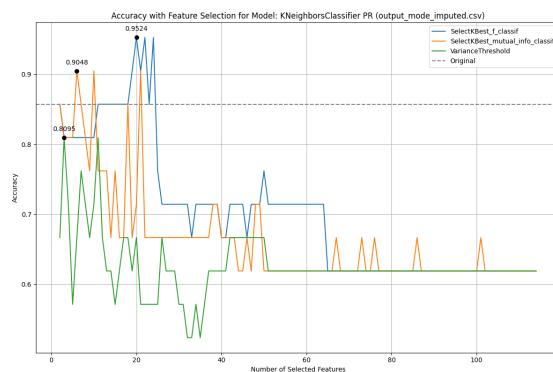


Figure 12: PR

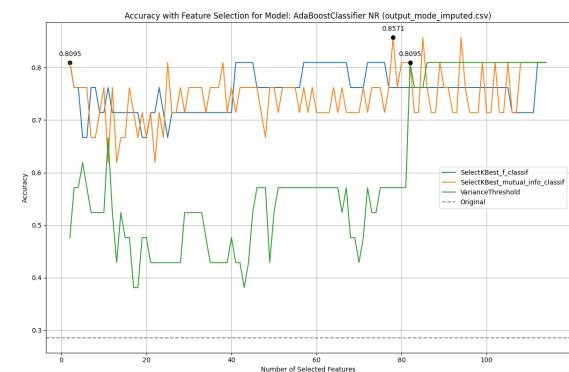


Figure 13: NR

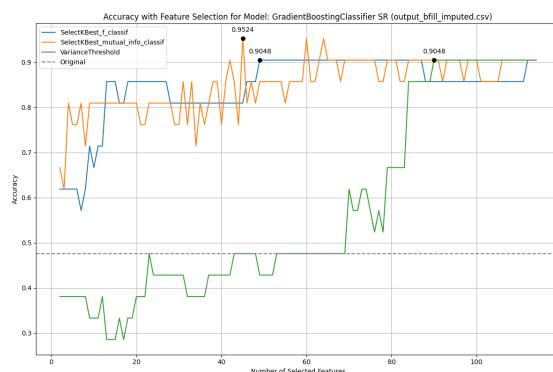


Figure 14: SR

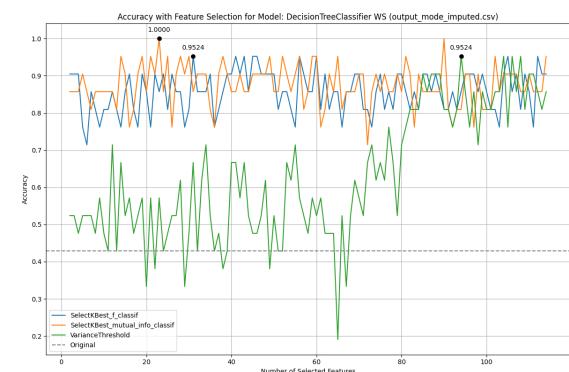
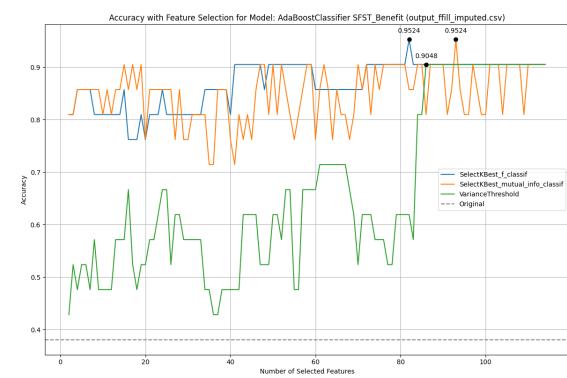
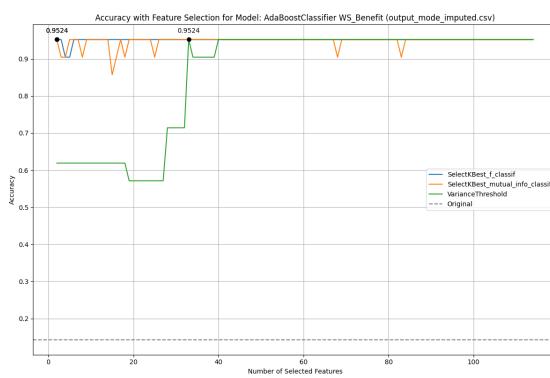
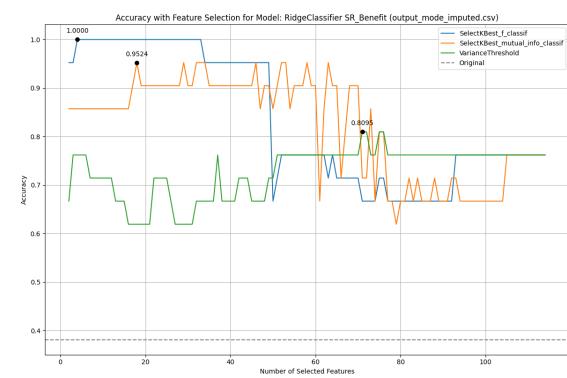
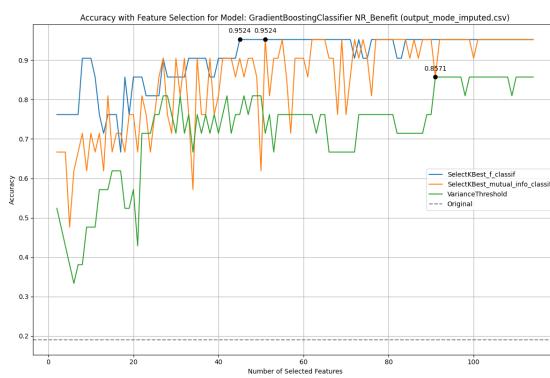
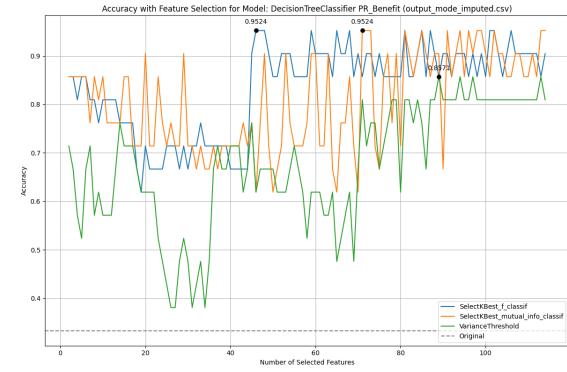
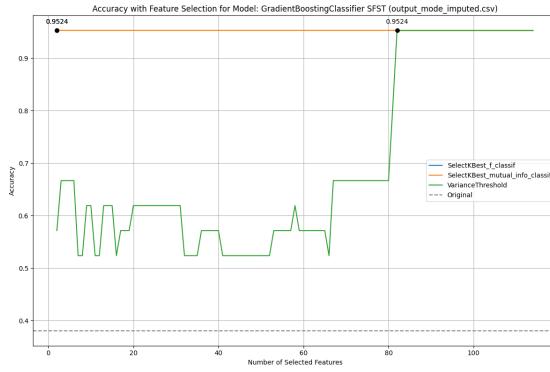


Figure 15: WS



12.4.2 Specific Features

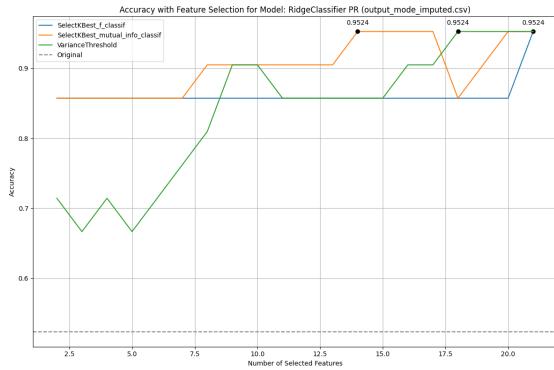


Figure 22: PR

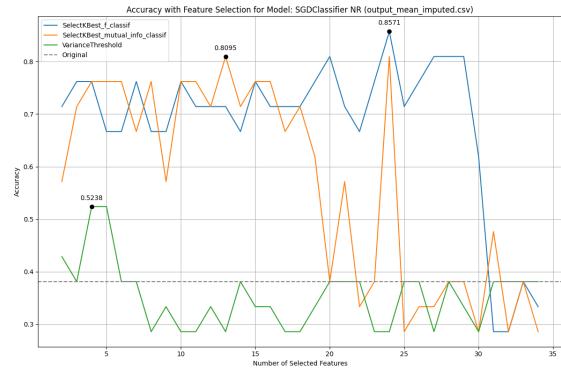


Figure 23: NR

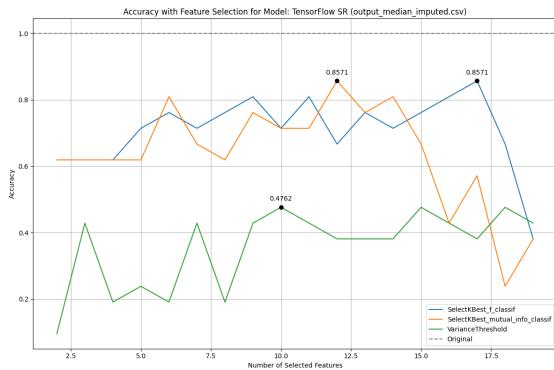


Figure 24: SR

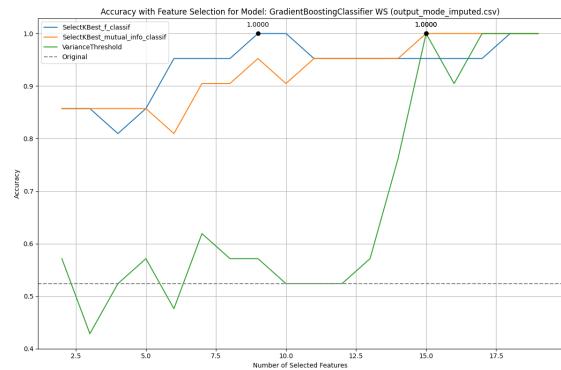


Figure 25: WS

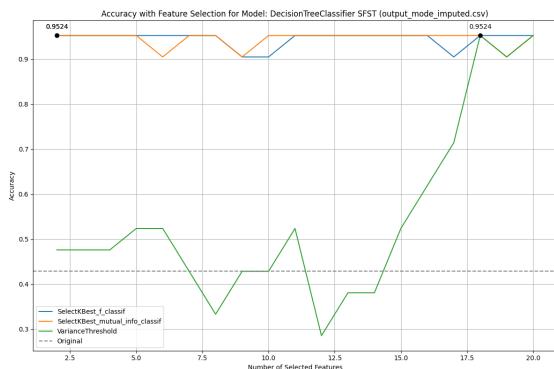


Figure 26: SFST

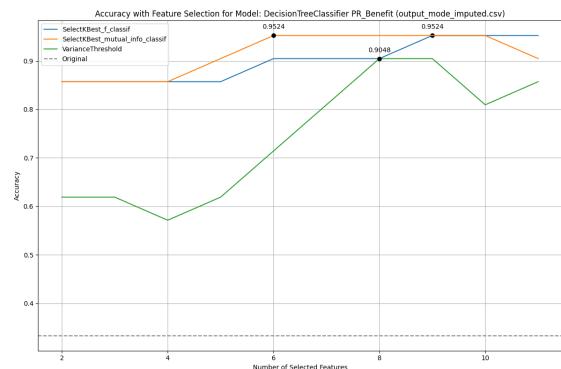


Figure 27: PR Benefit

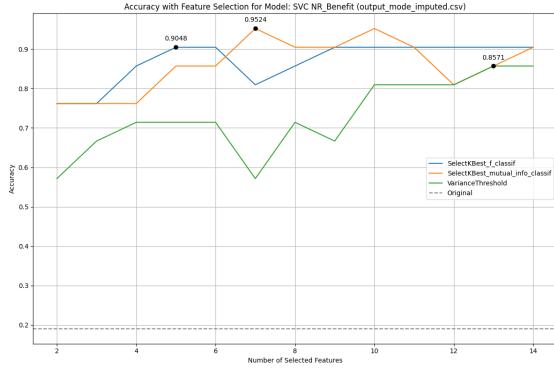


Figure 28: NR Benefit

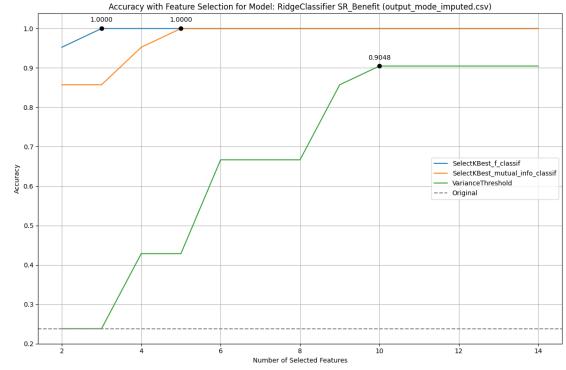


Figure 29: SR Benefit

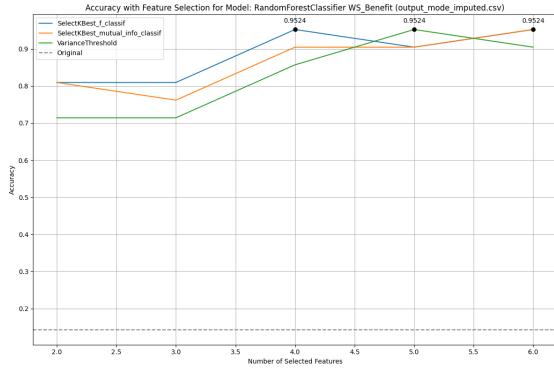


Figure 30: WS Benefit

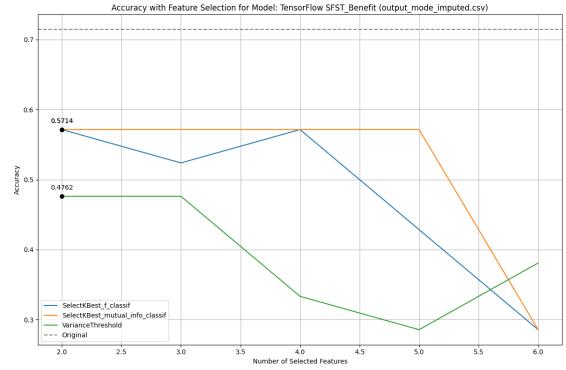


Figure 31: SFST Benefit

12.5 Regression Models

12.5.1 All Features

Feature	NNorm MSE	Norm MSE	Model	Dataset
NR	0.17	0.48	AdaBoostRegressor	Interpolated
PR	0.12	0.19	AdaBoostRegressor	custom
SR	0.37	0.62	AdaBoostRegressor	bfill
SFST	0.20	0.33	AdaBoostRegressor	ffill
WS	0.25	0.49	AdaBoostRegressor	knn
NR Benefit	0.39	0.45	MLPRegressor	ffill
PR Benefit	0.22	0.23	SGDRegressor	iterative
SR Benefit	0.17	0.22	AdaBoostRegressor	bfill
SFST Benefit	0.51	0.98	AdaBoostRegressor	custom
WS Benefit	1.19	1.21	AdaBoostRegressor	bfill

Table 32: Best MSE for Non-Normalized and Normalized Data with Algorithm and Dataset

Feature	PNorm MSE	Model	Dataset
NR	0.31	AdaBoostRegressor	knn
PR	0.21	AdaBoostRegressor	iterative
SR	0.46	AdaBoostRegressor	knn
SFST	0.26	AdaBoostRegressor	custom
WS	0.60	AdaBoostRegressor	custom
NR Benefit	0.28	MLPRegressor	knn
PR Benefit	0.49 & MLPRegressor	ffill	
SR Benefit	0.07	SGDRegressor	mode
SFST Benefit	0.71	AdaBoostRegressor	iterative
WS Benefit	0.63	AdaBoostRegressor	mean

Table 33: Best MSE for Pre-Normalized Data

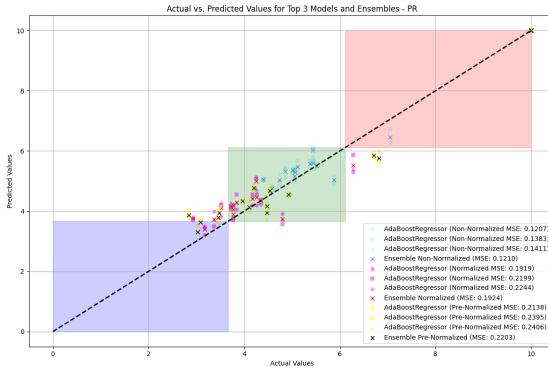


Figure 32: PR

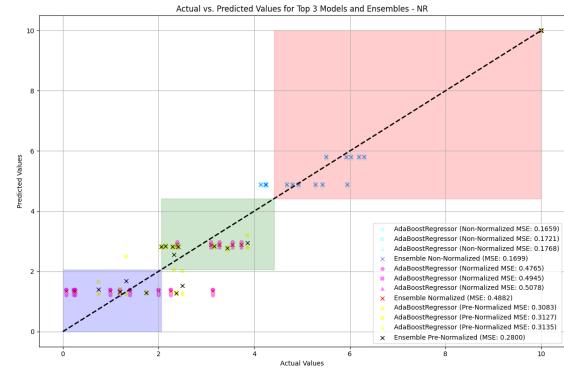


Figure 33: NR

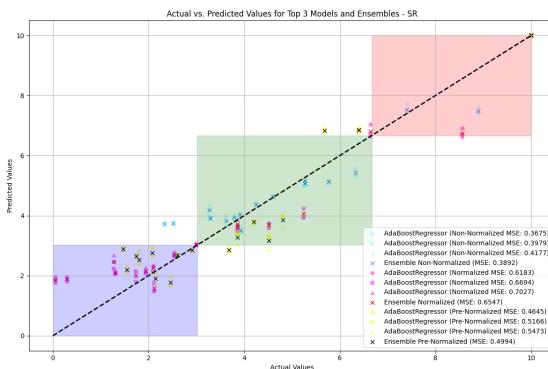


Figure 34: SR

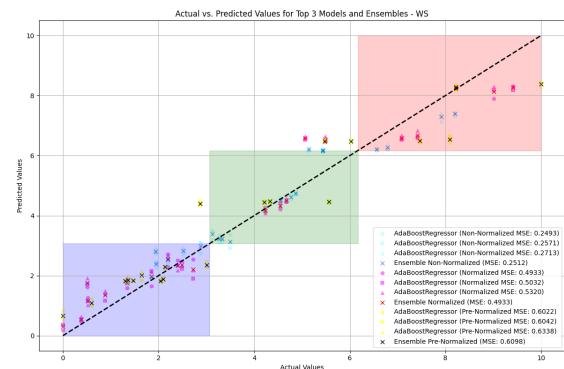


Figure 35: WS

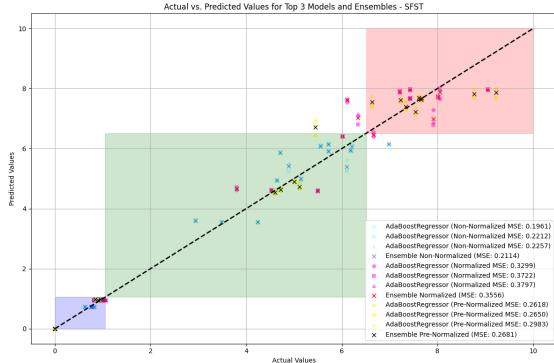


Figure 36: SFST

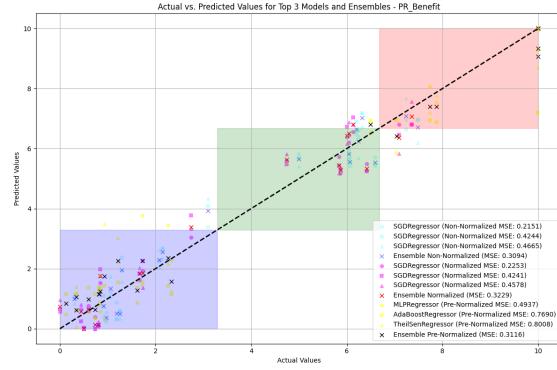


Figure 37: PR Benefit

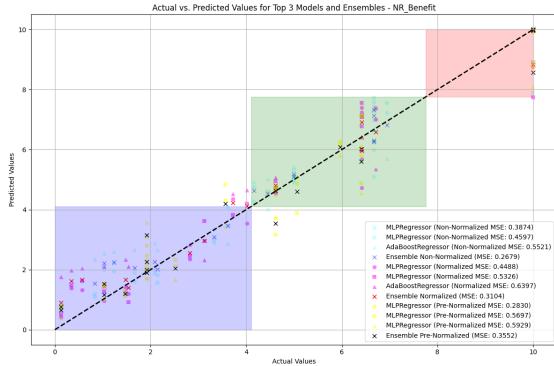


Figure 38: NR Benefit

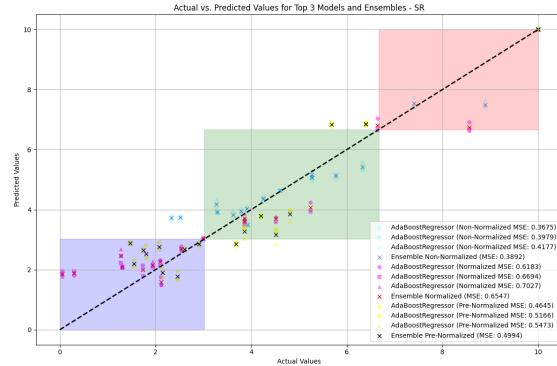


Figure 39: SR Benefit

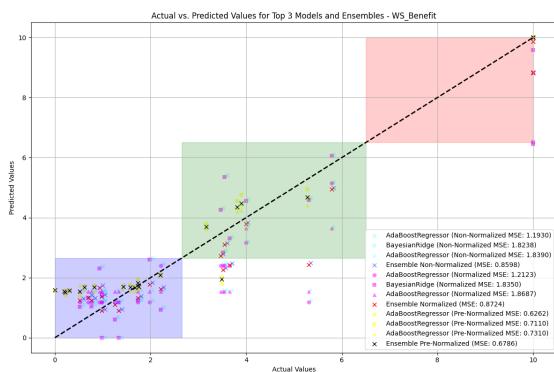


Figure 40: WS Benefit

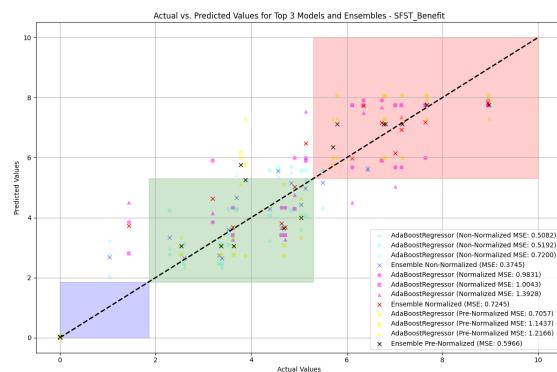


Figure 41: SFST Benefit

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
PR	75.00%	100.00%	87.50%	75.00%	100.00%	87.50%
NR	85.71%	85.71%	100.00%	85.71%	85.71%	100.00%
SR	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
WS Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
PR Benefit	100.00%	85.71%	75.00%	100.00%	85.71%	75.00%
NR Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SR Benefit	100.00%	88.89%	100.00%	100.00%	88.89%	100.00%
SFST Benefit	87.50%	100.00%	100.00%	87.50%	100.00%	100.00%

Table 34: Class-wise Accuracy for Non-Normalized Data: Best Model vs. Ensemble

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	100.00%	100.00%	100.00%	83.33%	100.00%
PR	75.00%	100.00%	87.50%	75.00%	100.00%	87.50%
NR	85.71%	85.71%	100.00%	85.71%	85.71%	100.00%
SR	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST	100.00%	100.00%	100.00%	100.00%	100.00%	0.00%
WS Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
PR Benefit	100.00%	85.71%	100.00%	100.00%	85.71%	100.00%
NR Benefit	90.00%	100.00%	100.00%	90.00%	100.00%	100.00%
SR Benefit	100.00%	88.89%	100.00%	100.00%	88.89%	100.00%
SFST Benefit	87.50%	83.33%	100.00%	87.50%	83.33%	100.00%

Table 35: Class-wise Accuracy for Normalized Data: Best Model vs. Ensemble

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
PR	0.00%	100.00%	100.00%	0.00%	92.31%	100.00%
NR	0.00%	0.00%	100.00%	0.00%	0.00%	100.00%
SR	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST	100.00%	100.00%	100.00%	100.00%	100.00%	0.00%
WS Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
PR Benefit	100.00%	85.71%	100.00%	100.00%	85.71%	75.00%
NR Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SR Benefit	100.00%	88.89%	100.00%	100.00%	88.89%	100.00%
SFST Benefit	87.50%	100.00%	100.00%	87.50%	100.00%	100.00%

Table 36: Class-wise Accuracy for Pre-Normalized Data: Best Model vs. Ensemble

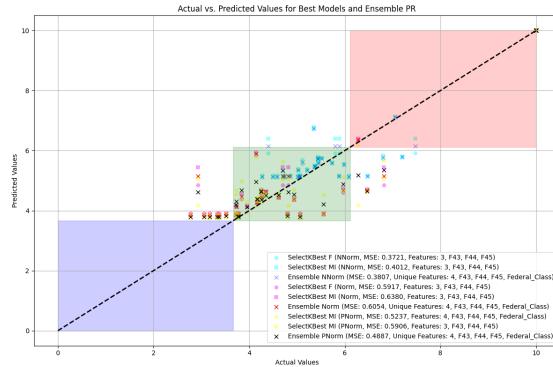


Figure 42: PR

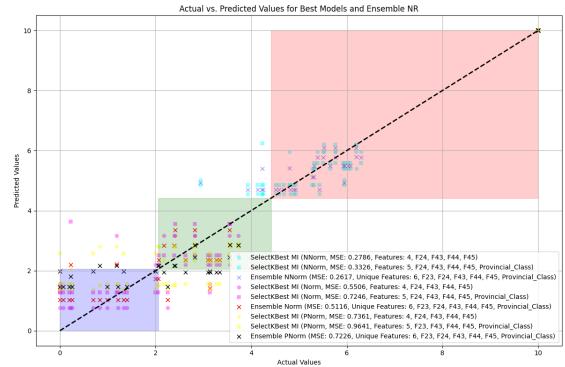


Figure 43: NR

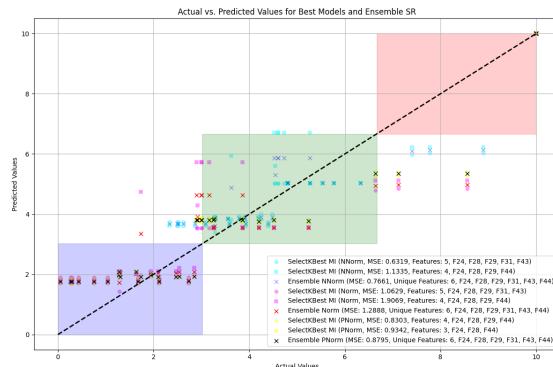


Figure 44: SR

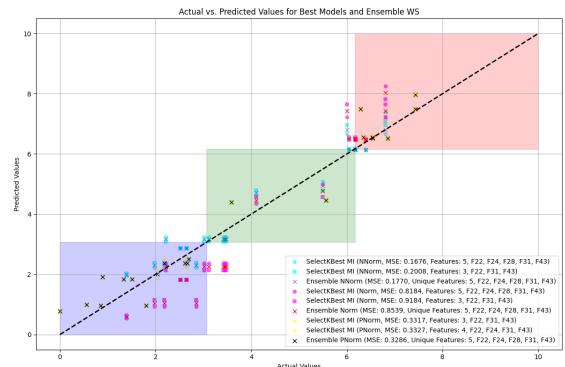


Figure 45: WS

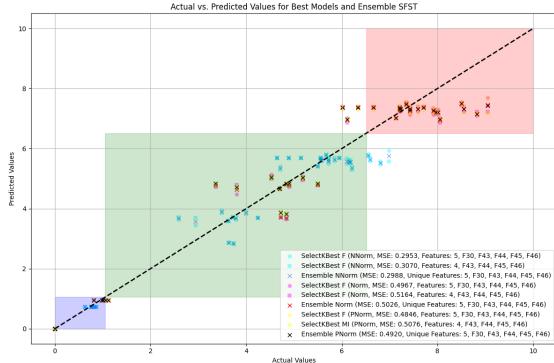


Figure 46: SFST

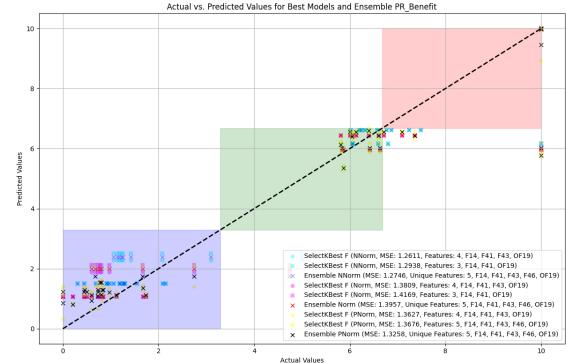


Figure 47: PR Benefit

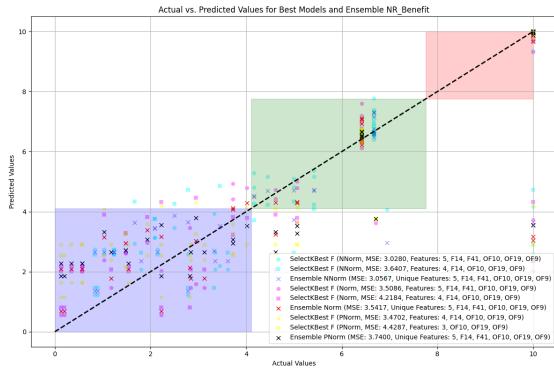


Figure 48: NR Benefit

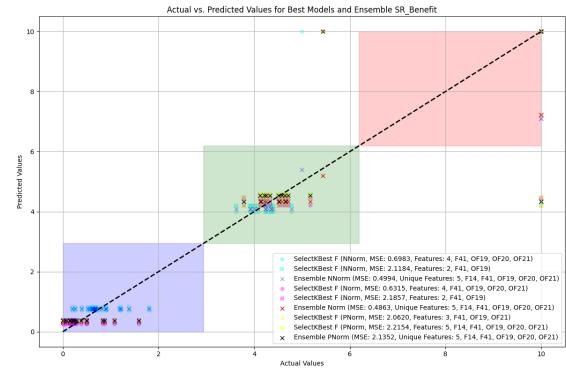


Figure 49: SR Benefit

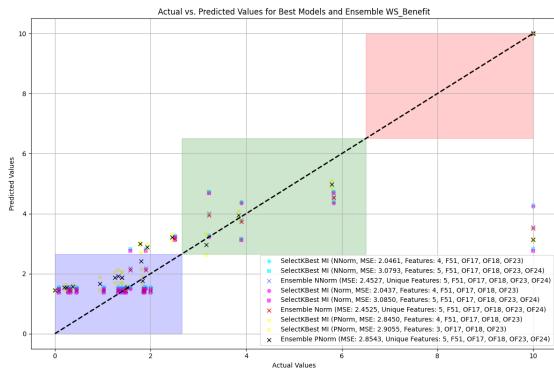


Figure 50: WS Benefit

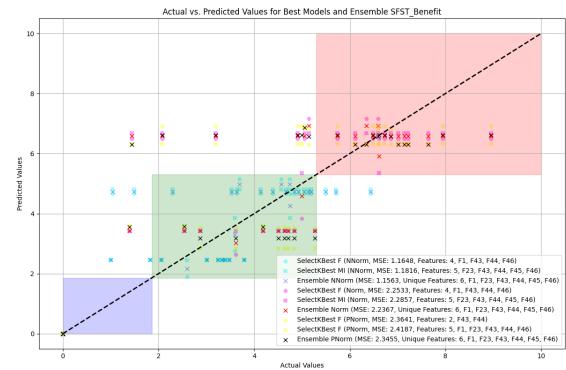


Figure 51: SFST Benefit

Function	Top Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	50.00%	100.00%	100.00%	50.00%	100.00%
PR	0.00%	100.00%	87.50%	0.00%	100.00%	87.50%
NR	57.14%	71.43%	100.00%	28.57%	85.71%	100.00%
SR	90.00%	80.00%	83.33%	90.00%	100.00%	83.33%
SFST	100.00%	42.86%	100.00%	100.00%	42.86%	100.00%
WS_Benefit	100.00%	28.57%	100.00%	100.00%	28.57%	100.00%
PR_Benefit	100.00%	42.86%	50.00%	100.00%	85.71%	50.00%
NR_Benefit	80.00%	100.00%	75.00%	100.00%	100.00%	75.00%
SR_Benefit	20.00%	88.89%	0.00%	20.00%	88.89%	0.00%
SFST_Benefit	87.50%	50.00%	100.00%	87.50%	50.00%	100.00%

Table 37: Comparison of Top Model and Ensemble Accuracy for Non-Normalized Data

Function	Top Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	80.00%	100.00%	100.00%	60.00%	100.00%
PR	N/A	100.00%	100.00%	N/A	100.00%	87.50%
NR	N/A	0.00%	100.00%	N/A	0.00%	100.00%
SR	0.00%	100.00%	71.43%	0.00%	100.00%	100.00%
SFST	100.00%	100.00%	0.00%	100.00%	100.00%	0.00%
WS_Benefit	100.00%	28.57%	100.00%	81.82%	42.86%	100.00%
PR_Benefit	100.00%	85.71%	50.00%	100.00%	57.14%	50.00%
NR_Benefit	75.00%	77.78%	75.00%	100.00%	77.78%	75.00%
SR_Benefit	100.00%	88.89%	0.00%	100.00%	88.89%	0.00%
SFST_Benefit	87.50%	100.00%	0.00%	87.50%	100.00%	0.00%

Table 38: Comparison of Top Model and Ensemble Accuracy for Pre-Normalized Data

12.5.2 Specific Features

Feature	NNorm MSE	Norm MSE	Model	Dataset
NR	0.15	0.43	AdaBoostRegressor	ffill
PR	0.11	0.18	AdaBoostRegressor	custom
SR	0.08	0.13	MLPRegressor	custom
SFST	0.16	0.26	AdaBoostRegressor	knn
WS	0.10	0.19	MLPRegressor	mode
NR_Benefit	0.08	0.09	MLPRegressor	mode
PR_Benefit	0.01	0.01	TheilSenRegressor	custom
SR_Benefit	0.07	0.09	RANSACRegressor	custom
SFST_Benefit	1.66	3.22	RANSACRegressor	interpolated.csv
WS_Benefit	0.53	0.54	RandomForestRegressor	custom

Table 39: Best MSE for Non-Normalized and Normalized Data with Algorithm and Dataset

Feature	Model	Dataset	MSE
NR	AdaBoostRegressor	iterative	0.24
PR	AdaBoostRegressor	iterative	0.24
SR	MLPRegressor	custom	0.27
SFST	AdaBoostRegressor	mean	0.20
WS	MLPRegressor	knn	0.08
NR_Benefit	MLPRegressor	custom	0.07
PR_Benefit	TheilSenRegressor	knn	0.01
SR_Benefit	SGDRegressor	mean	0.01
SFST_Benefit	Ridge	iterative	4.37
WS_Benefit	DecisionTreeRegressor	mean	0.11

Table 40: Best MSE for PNorm Data

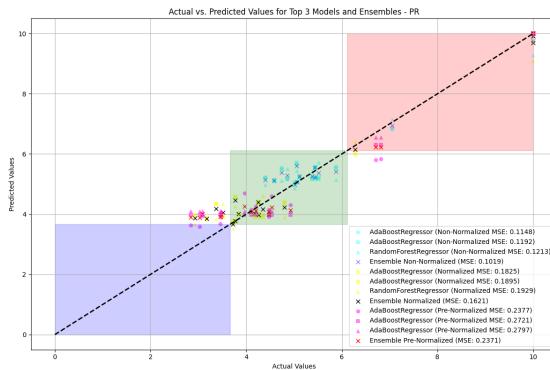


Figure 52: PR

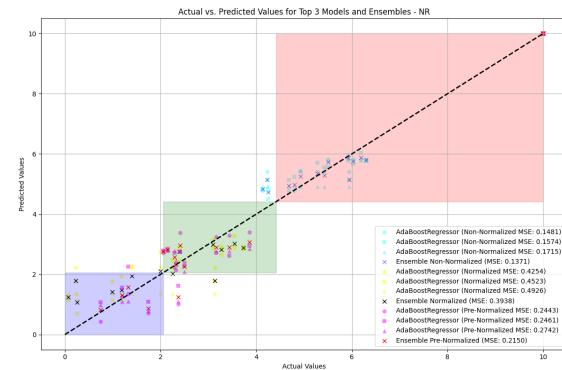


Figure 53: NR

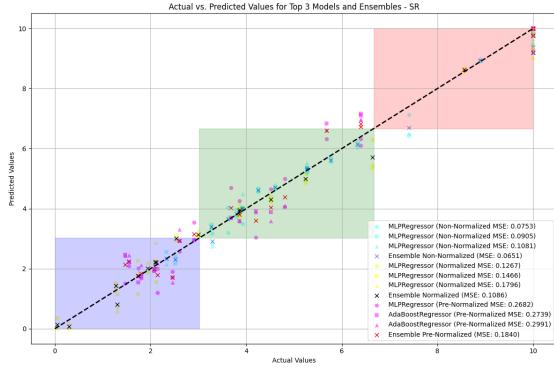


Figure 54: SR

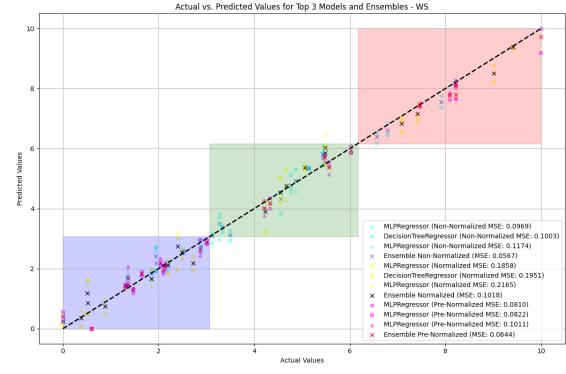


Figure 55: WS

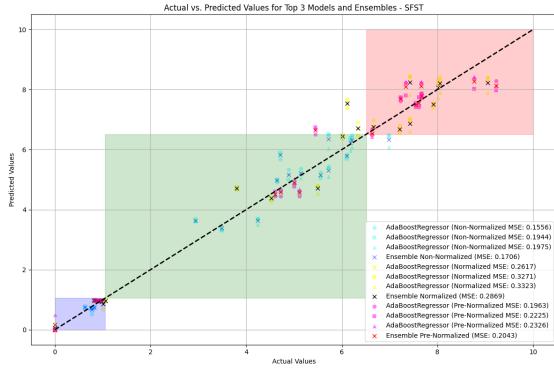


Figure 56: SFST

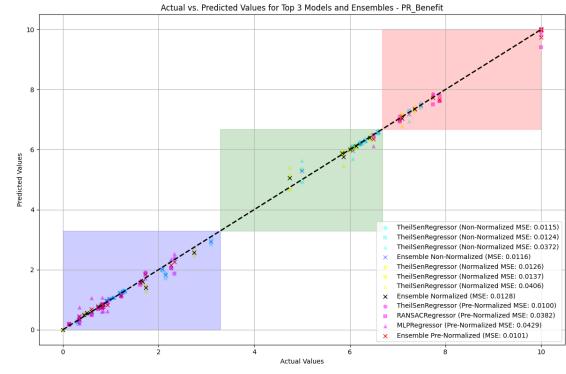


Figure 57: PR Benefit

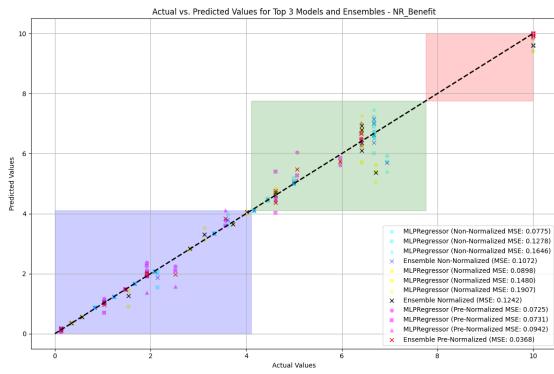


Figure 58: NR Benefit

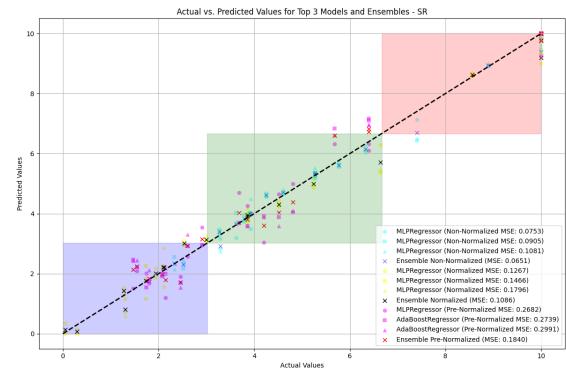


Figure 59: SR Benefit

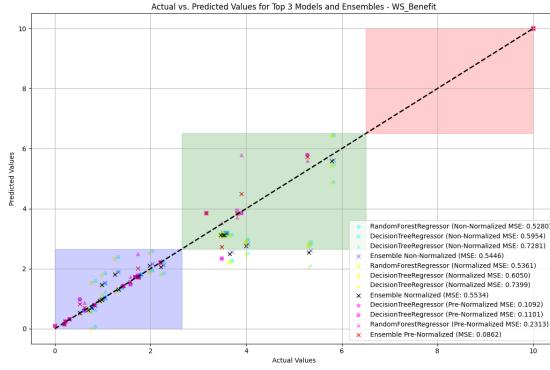


Figure 60: WS Benefit

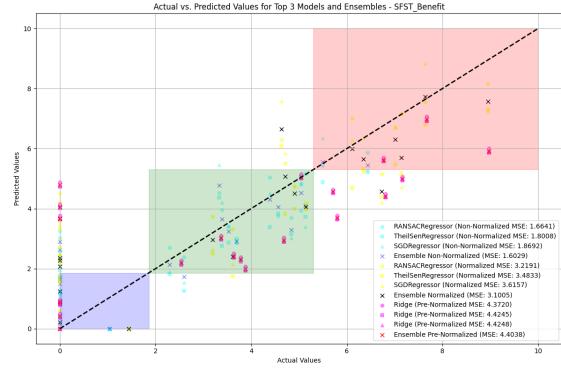


Figure 61: SFST Benefit

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	92.86%	90.48%	85.71%	92.86%	90.48%	85.71%
PR	N/A	92.31%	87.50%	N/A	96.15%	87.50%
NR	N/A	66.67%	91.67%	N/A	0.00%	100.00%
SR	20.00%	96.15%	90.91%	0.00%	100.00%	90.91%
SFST	100.00%	100.00%	0.00%	100.00%	100.00%	0.00%
WS Benefit	92.86%	36.36%	100.00%	96.43%	27.27%	100.00%
PR Benefit	100.00%	77.78%	90.00%	100.00%	77.78%	80.00%
NR Benefit	94.44%	100.00%	75.00%	94.44%	100.00%	75.00%
SR Benefit	100.00%	92.86%	60.00%	100.00%	92.86%	0.00%
SFST Benefit	66.67%	75.00%	66.67%	73.33%	62.50%	66.67%

Table 41: Class-wise Accuracy for Non-Normalized Data: Best Model vs. Ensemble

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	75.00%	100.00%	95.24%	62.50%	100.00%
PR	95.24%	40.00%	100.00%	95.24%	40.00%	100.00%
NR	100.00%	N/A	100.00%	92.31%	N/A	100.00%
SR	96.00%	71.43%	100.00%	96.00%	71.43%	100.00%
SFST	100.00%	100.00%	95.24%	100.00%	100.00%	95.24%
WS Benefit	96.55%	75.00%	100.00%	93.10%	75.00%	100.00%
PR Benefit	100.00%	100.00%	44.44%	100.00%	100.00%	44.44%
NR Benefit	100.00%	64.29%	100.00%	100.00%	85.71%	50.00%
SR Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST Benefit	23.53%	80.00%	65.00%	29.41%	80.00%	65.00%

Table 42: Class-wise Accuracy for Normalized Data: Best Model vs. Ensemble

Feature	Best Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	92.86%	100.00%	100.00%	92.86%	100.00%
PR	0.00%	100.00%	85.71%	0.00%	100.00%	78.57%
NR	66.67%	81.25%	100.00%	53.33%	68.75%	100.00%
SR	78.26%	88.89%	80.00%	78.26%	66.67%	80.00%
SFST	100.00%	66.67%	100.00%	100.00%	66.67%	100.00%
WS Benefit	96.77%	12.50%	100.00%	100.00%	25.00%	100.00%
PR Benefit	100.00%	92.31%	66.67%	100.00%	92.31%	66.67%
NR Benefit	100.00%	88.89%	75.00%	100.00%	88.89%	75.00%
SR Benefit	100.00%	0.00%	60.00%	100.00%	0.00%	0.00%
SFST Benefit	69.23%	46.67%	42.86%	61.54%	53.33%	42.86%

Table 43: Class-wise Accuracy for Pre-Normalized Data: Best Model vs. Ensemble

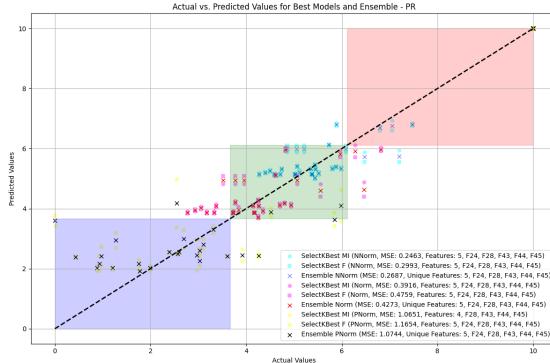


Figure 62: PR

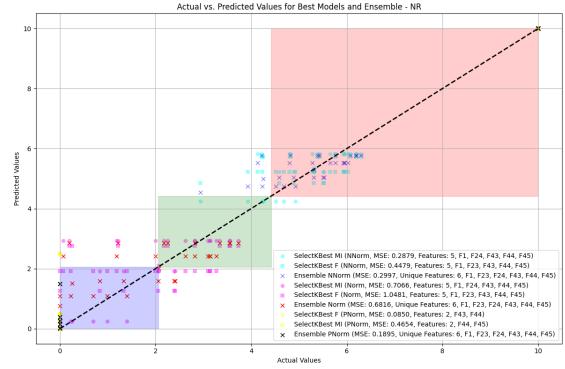


Figure 63: NR

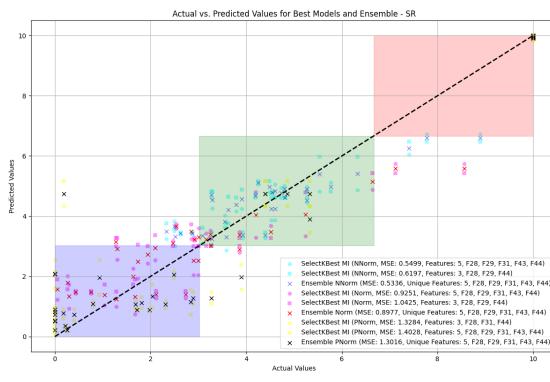


Figure 64: SR

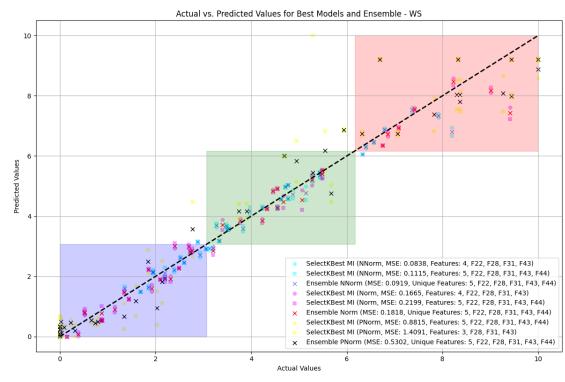


Figure 65: WS

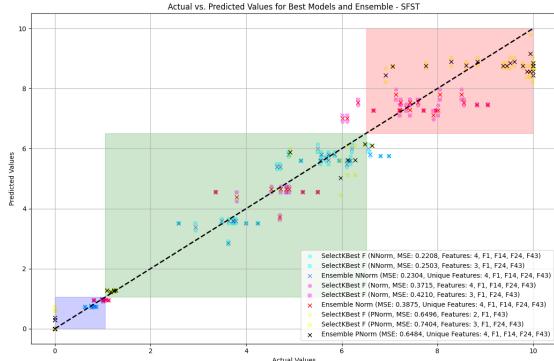


Figure 66: SFST

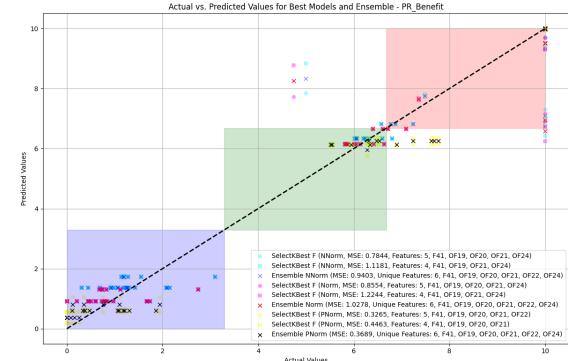


Figure 67: PR Benefit

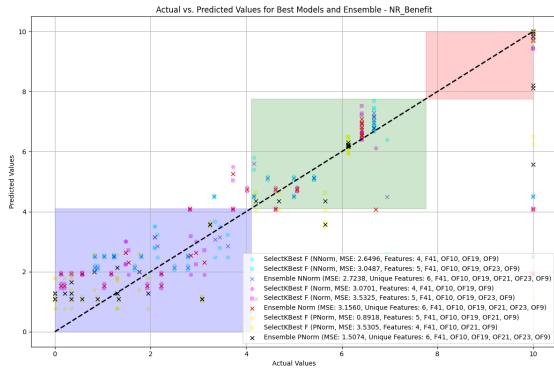


Figure 68: NR Benefit

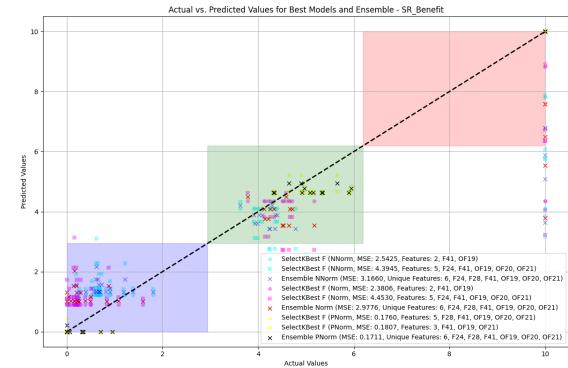


Figure 69: SR Benefit

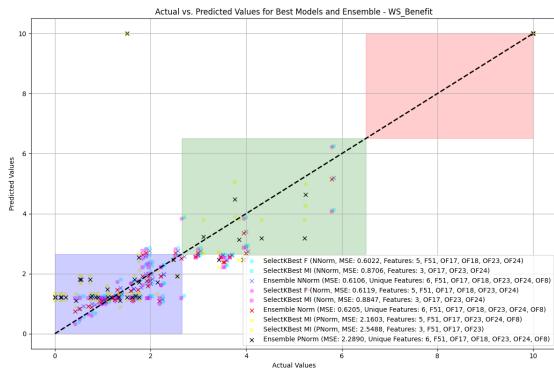


Figure 70: WS Benefit

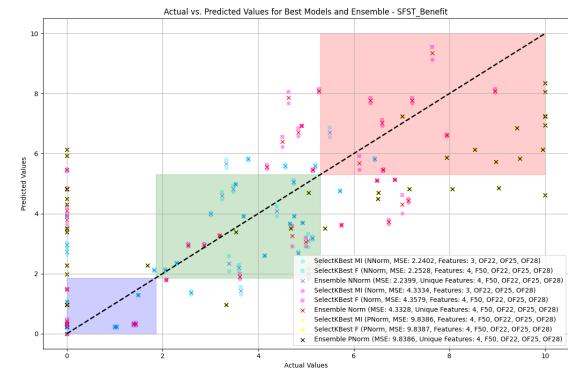


Figure 71: SFST Benefit

Function	Top Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	92.86%	90.48%	85.71%	92.86%	90.48%	85.71%
PR	N/A	92.31%	87.50%	N/A	96.15%	87.50%
NR	N/A	66.67%	91.67%	N/A	0.00%	100.00%
SR	20.00%	96.15%	90.91%	0.00%	100.00%	90.91%
SFST	100.00%	100.00%	0.00%	100.00%	100.00%	0.00%
WS Benefit	92.86%	36.36%	100.00%	96.43%	27.27%	100.00%
PR Benefit	100.00%	77.78%	90.00%	100.00%	77.78%	80.00%
NR Benefit	94.44%	100.00%	75.00%	94.44%	100.00%	75.00%
SR Benefit	100.00%	92.86%	60.00%	100.00%	92.86%	60.00%
SFST Benefit	66.67%	75.00%	66.67%	73.33%	62.50%	66.67%

Table 44: Comparison of Top Model and Ensemble Accuracy for Non-Normalized (NNorm) Data

Function	Top Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	92.86%	100.00%	100.00%	92.86%	100.00%
PR	95.24%	40.00%	100.00%	95.24%	40.00%	100.00%
NR	100.00%	N/A	100.00%	92.31%	N/A	100.00%
SR	96.00%	71.43%	100.00%	96.00%	71.43%	100.00%
SFST	100.00%	100.00%	95.24%	100.00%	100.00%	95.24%
WS Benefit	96.55%	75.00%	100.00%	93.10%	75.00%	100.00%
PR Benefit	100.00%	100.00%	44.44%	100.00%	100.00%	44.44%
NR Benefit	100.00%	64.29%	100.00%	100.00%	85.71%	50.00%
SR Benefit	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
SFST Benefit	23.53%	80.00%	65.00%	29.41%	80.00%	65.00%

Table 45: Comparison of Top Model and Ensemble Accuracy for Normalized (Norm) Data

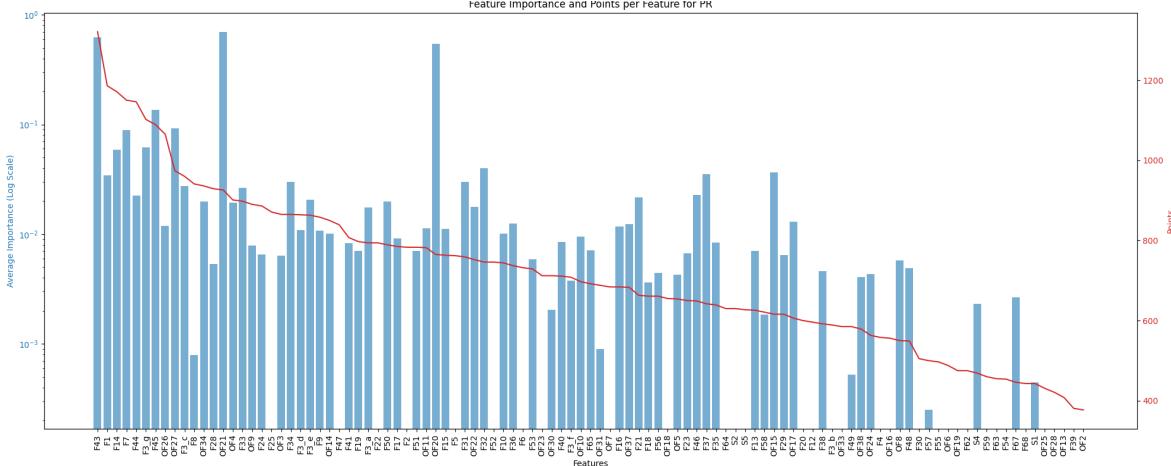


Figure 72: PR

Function	Top Model			Ensemble		
	Lower	Moderate	Higher	Lower	Moderate	Higher
WS	100.00%	92.86%	100.00%	100.00%	92.86%	100.00%
PR	0.00%	100.00%	85.71%	0.00%	100.00%	78.57%
NR	66.67%	81.25%	100.00%	53.33%	68.75%	100.00%
SR	78.26%	88.89%	80.00%	78.26%	66.67%	80.00%
SFST	100.00%	66.67%	100.00%	100.00%	66.67%	100.00%
WS Benefit	96.77%	12.50%	100.00%	100.00%	12.50%	100.00%
PR Benefit	100.00%	92.31%	66.67%	100.00%	92.31%	66.67%
NR Benefit	100.00%	88.89%	75.00%	100.00%	88.89%	75.00%
SR Benefit	100.00%	0.00%	100.00%	100.00%	0.00%	100.00%
SFST Benefit	69.23%	46.67%	42.86%	61.54%	53.33%	42.86%

Table 46: Comparison of Top Model and Ensemble Accuracy for Post-Normalized (PNorm) Data

12.6 CausalML

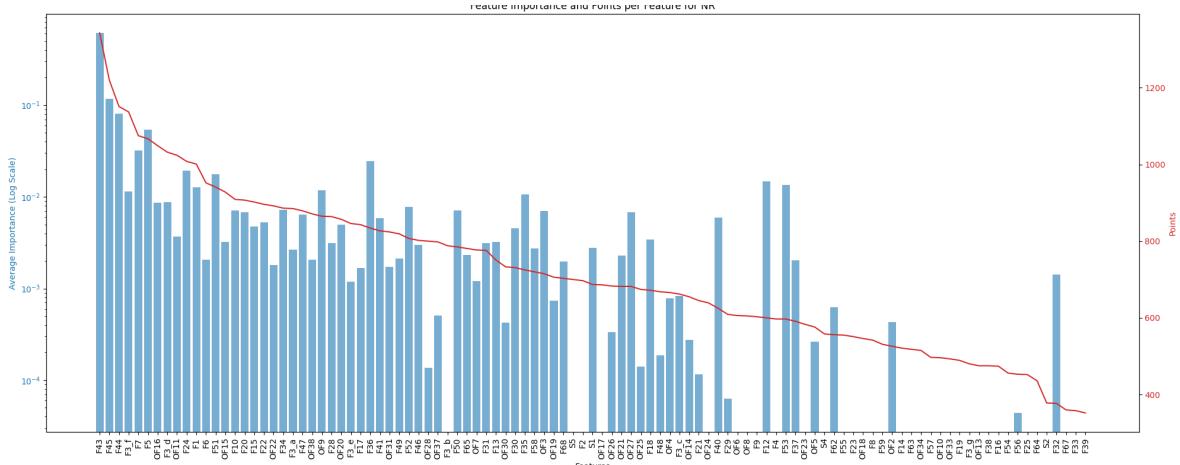


Figure 73: NR

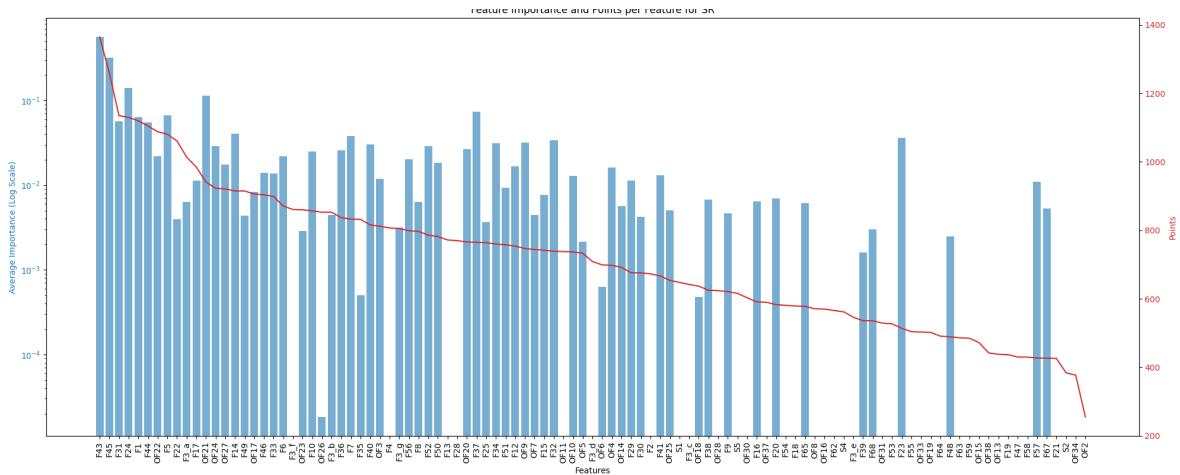


Figure 74: SR

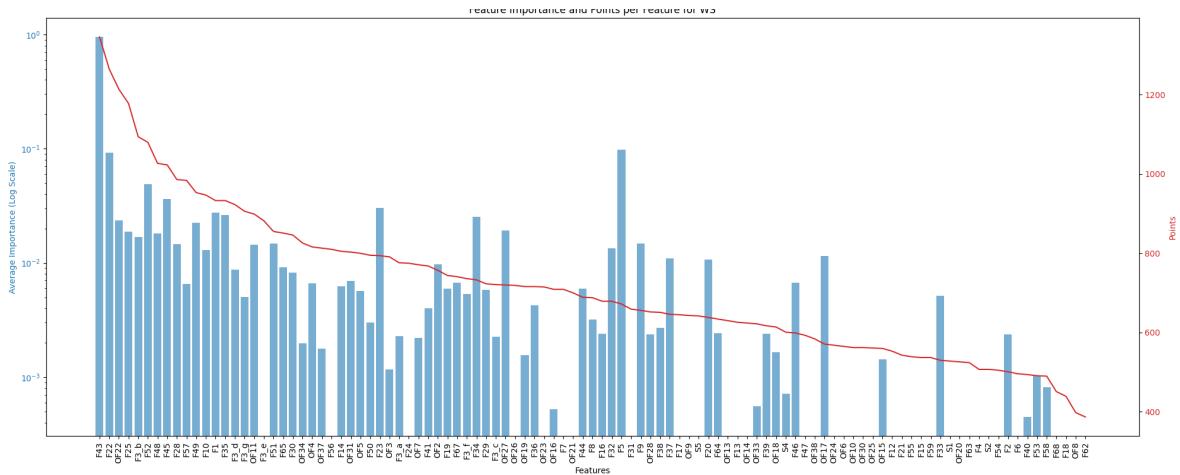


Figure 75: WS

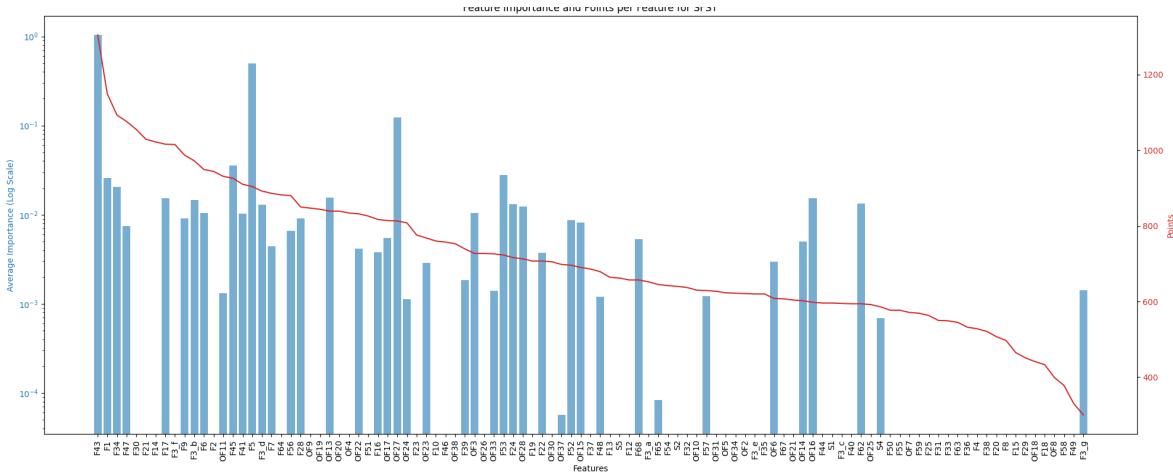


Figure 76: SFST

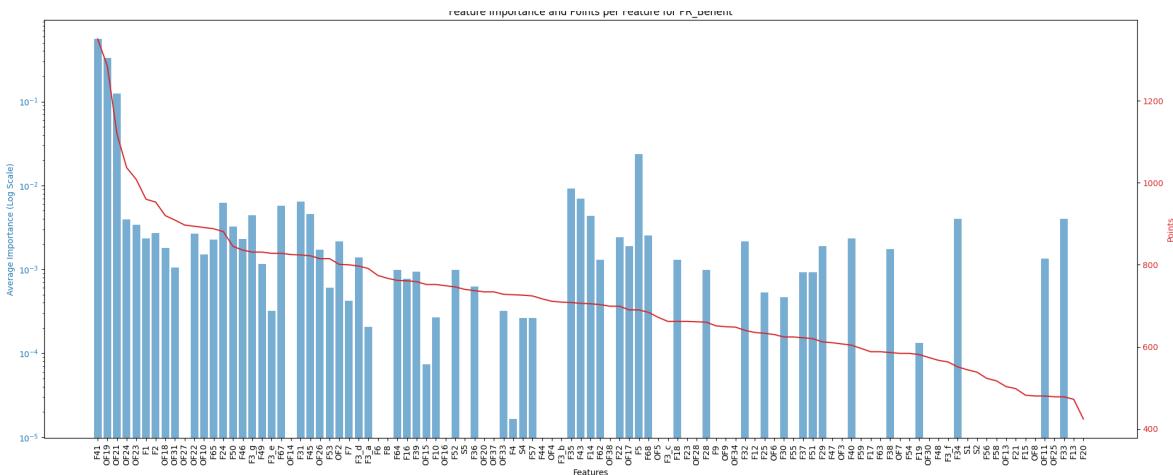


Figure 77: PR Benefit

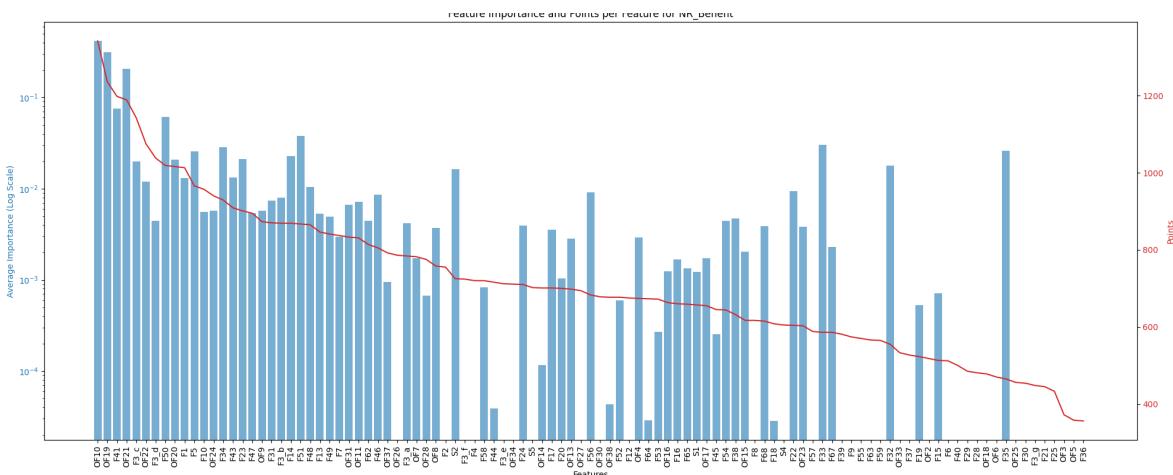


Figure 78: NR Benefit

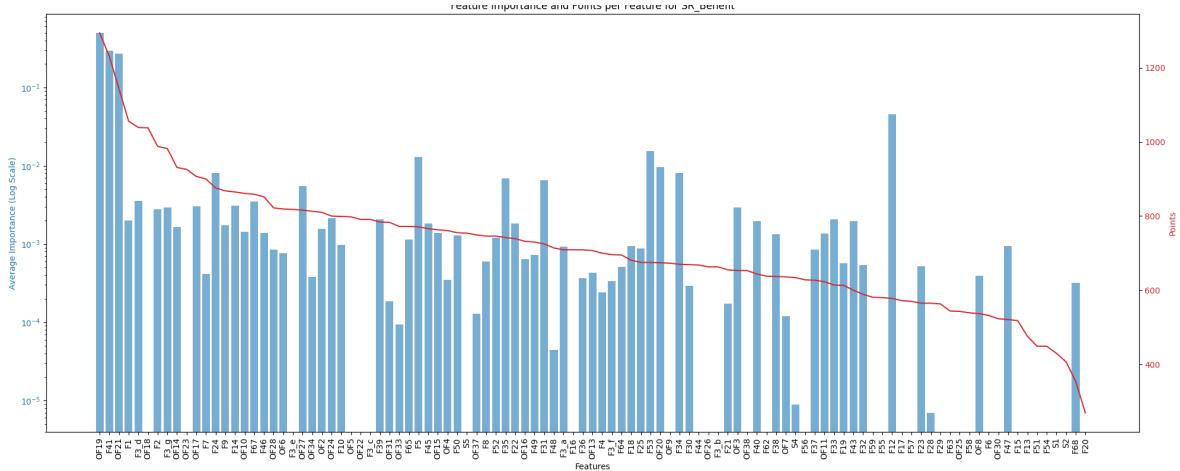


Figure 79: SR Benefit

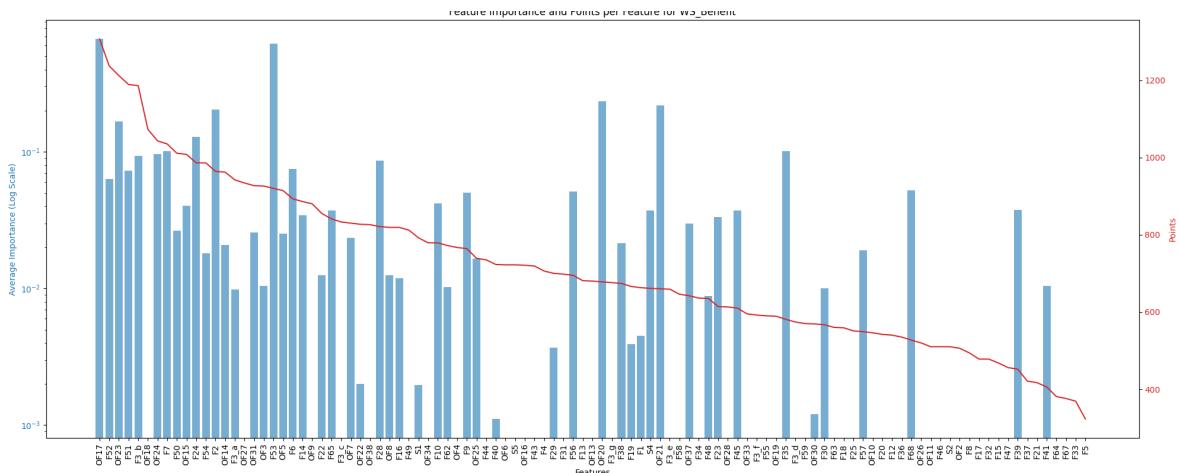


Figure 80: WS Benefit

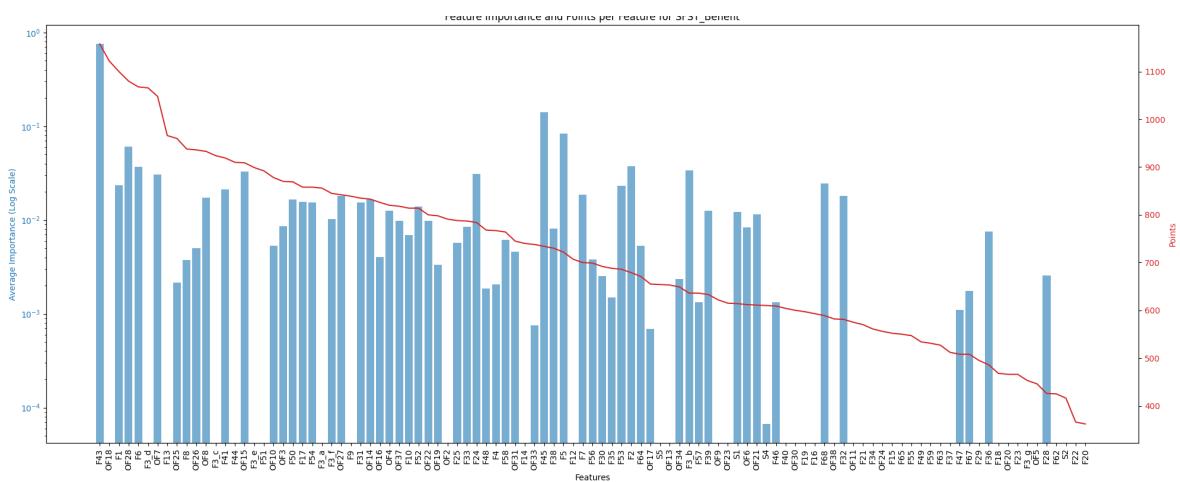


Figure 81: SFST Benefit

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F43': 0.536779 'F45': 0.192897	'OF8': -0.012789 'OF13': -0.009090
DecisionTreeRegressor	'F43': 1.515596 'F29': 0.037213	'F65': -0.022642 'F24': -0.017126
GradientBoostingRegressor	'F43': 1.346256 'OF26': 0.021221	'F3d': -0.003275 'F13': -0.000601
RandomForestRegressor	'F43': 1.182576 'OF26': 0.042053	'F13': -0.001210 'F3f': -0.000922
AdaBoostRegressor	'F43': 1.431664 'OF26': 0.020958	'F15': -0.001777 'F49': -0.001246
KNeighborsRegressor	'F5': 0.708910 'F3d': 0.003255	'OF27': -0.055648 'F41': 0.000000
MLPRegressor	'OF27': 1.309096 'F14': 0.515494	'F5': -2.923432 'F25': -0.309051
ElasticNet	'F43': 0.441983 'F45': 0.245119	'OF2': 0.000000 'F39': 0.000000
SVR	'F5': 0.048764 'OF27': 0.010900	'OF18': -0.000016 'F3g': -0.000009
BayesianRidge	'F43': 0.383303 'F45': 0.171036	'OF8': -0.004480 'OF38': -0.003056
KernelRidge	'F43': 0.542174 'F45': 0.191382	'OF8': -0.013750 'OF13': -0.008700
LinearRegression	'F43': 0.590500 'F7': 0.202867	'OF8': -0.012748 'OF6': -0.011382
RANSACRegressor	'OF21': 9.560983 'OF20': 7.545941	'F52': -0.194904 'OF16': -0.166460
TheilSenRegressor	'F43': 0.393848 'F45': 0.244852	'OF28': -0.006563 'OF8': -0.006284
Average Importance	'Feature OF21': 0.702052 'Feature F43': 0.625989 'Feature OF20': 0.544127 'Feature F45': 0.135707	'Feature F25': -0.009865 'Feature OF16': -0.011711 'Feature OF6': -0.011840 'Feature F5': -0.138349
Voting System	'Feature F43': 1321 points 'Feature F1': 1186 points 'Feature F14': 1171 points 'Feature F7': 1150 points	'Feature OF28': 421 points 'Feature OF13': 408 points 'Feature F39': 381 points 'Feature OF2': 377 points

Table 47: Features for PR Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F43': 0.390725 'F44': 0.165844	'OF6': -0.020725 'OF33': -0.013250
DecisionTreeRegressor	'F43': 1.774561 'F24': 0.057620	'F3e': -0.008175 'F49': -0.005408
GradientBoostingRegressor	'F43': 1.620630 'F24': 0.022750	'F3c': -0.000362 'F12': -0.000226
RandomForestRegressor	'F43': 0.817959 'F12': 0.218402	'F47': -0.011895 'F21': -0.009079
AdaBoostRegressor	'F43': 1.720332 'F24': 0.038324	'F50': -0.000827 'F12': -0.000806
KNeighborsRegressor	'F5': 0.533441 'OF27': 0.061891	'OF13': -0.000245 'OF9': -0.000204
MLPRegressor	'F5': 0.153295 'F45': 0.111910	'F2': -0.017481 'F14': -0.014245
ElasticNet	'F43': 0.506674 'F45': 0.266837	'OF2': 0.000000 'F38': 0.000000
SVR	'F5': 0.025143 'OF27': 0.006574	'OF5': -0.000005 'F3g': -0.000004
BayesianRidge	'F43': 0.428843 'F45': 0.148416	'OF6': -0.007226 'OF34': -0.006675
KernelRidge	'F43': 0.406266 'F44': 0.176189	'OF6': -0.021310 'OF8': -0.014864
LinearRegression	'F43': 0.346340 'F44': 0.199381	'OF6': -0.027631 'OF33': -0.018511
RANSACRegressor	'F45': 0.448435 'F44': 0.221599	'F33': -0.038320 'F8': -0.023706
TheilSenRegressor	'F43': 0.316211 'F44': 0.201227	'OF6': -0.027462 'OF33': -0.015219
Average Importance	'Feature F43': 0.612435 'Feature F45': 0.117268 'Feature F44': 0.081121 'Feature F5': 0.053705	'Feature F8': -0.004219 'Feature F33': -0.004765 'Feature OF6': -0.005082 'Feature OF33': -0.006181
Voting System	'Feature F43': 1343 points 'Feature F45': 1220 points 'Feature F44': 1151 points 'Feature F3f': 1137 points	'Feature F32': 377 points 'Feature F67': 360 points 'Feature F33': 358 points 'Feature F39': 352 points

Table 48: Features for NR Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F43': 1.147478 'F45': 0.091115	'OF24': -0.025517 'F24': -0.016204
DecisionTreeRegressor	'F43': 1.754513 'F22': 0.147253	'F20': -0.014787 'F29': -0.009592
GradientBoostingRegressor	'F43': 1.511286 'F22': 0.110634	'F20': -0.003293 'F46': -0.001353
RandomForestRegressor	'F43': 1.575029 'F22': 0.124296	'F20': -0.006019 'F2': -0.002894
AdaBoostRegressor	'F43': 1.391152 'F22': 0.044818	'OF6': -0.003275 'F51': -0.002221
KNeighborsRegressor	'F5': 0.559163 'OF27': 0.011390	'OF13': -0.001688 'OF9': -0.001394
MLPRegressor	'F5': 0.805548 'OF27': 0.188689	'F45': -0.084892 'F3e': -0.080038
ElasticNet	'F43': 0.898375 'F25': 0.020736	'OF27': -0.003515 'OF18': -0.000012
SVR	'F5': 0.036869 'F43': 0.000169	'OF27': -0.001506 'OF18': -0.000024
BayesianRidge	'F43': 0.726329 'F3b': 0.036111	'F68': -0.006553 'F6': -0.005987
KernelRidge	'F43': 1.163232 'F52': 0.098785	'OF24': -0.025737 'F24': -0.017471
LinearRegression	'F43': 1.185014 'F52': 0.186445	'OF24': -0.041529 'F24': -0.022150
RANSACRegressor	'F43': 0.787310 'F22': 0.434157	'F12': -0.331444 'F18': -0.283120
TheilSenRegressor	'F43': 1.195570 'F52': 0.204452	'OF24': -0.036281 'OF21': -0.026817
Average Importance	'Feature F43': 0.951765 'Feature F5': 0.097695 'Feature F22': 0.091916 'Feature F52': 0.048698	'Feature OF20': -0.008094 'Feature F62': -0.009468 'Feature F18': -0.020177 'Feature F12': -0.026529
Voting System	'Feature F43': 1346 points 'Feature F22': 1265 points 'Feature OF22': 1214 points 'Feature F25': 1178 points	'Feature F68': 451 points 'Feature F18': 439 points 'Feature OF8': 398 points 'Feature F62': 387 points

Table 49: Features for WS Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F43': 1.145591 'F1': 0.032092	'F49': -0.007396 'F15': -0.004882
DecisionTreeRegressor	'F43': 1.956398 'F48': 0.016230	'OF3': -0.012433 'F47': -0.006130
GradientBoostingRegressor	'F43': 1.658540 'F21': 0.016892	'OF18': -0.003418 'F2': -0.001442
RandomForestRegressor	'F43': 1.507879 'F44': 0.017893	'F65': -0.002316 'OF15': -0.001130
AdaBoostRegressor	'F43': 1.680548 'F21': 0.008374	'OF18': -0.005203 'F8': -0.003271
KNeighborsRegressor	'F5': 0.650557 'OF27': 0.009803	'OF13': -0.001657 'OF9': -0.001100
MLPRegressor	'F5': 6.260377 'OF27': 1.699317	'F35': -0.410244 'F36': -0.313736
ElasticNet	'F43': 0.809538 'F45': 0.102403	'OF18': -0.000019 'OF2': 0.000000
SVR	'F5': 0.026701 'F43': 0.000142	'OF18': -0.000016 'OF5': -0.000004
BayesianRidge	'F43': 0.811364 'F34': 0.018542	'F15': -0.002554 'OF8': -0.002456
KernelRidge	'F43': 1.142568 'F1': 0.031358	'F49': -0.007572 'F15': -0.004914
LinearRegression	'F43': 1.219160 'F1': 0.046488	'F49': -0.009860 'F31': -0.005465
RANSACRegressor	'F43': 1.518405 'F34': 0.129466	'F36': -0.016735 'F48': -0.012570
TheilSenRegressor	'F43': 1.308138 'F1': 0.044073	'F49': -0.008300 'F20': -0.005366
Average Importance	'Feature F43': 1.039090 'Feature F5': 0.497852 'Feature OF27': 0.122599 'Feature F45': 0.035529	'Feature F4': -0.014221 'Feature OF9': -0.019349 'Feature F36': -0.023610 'Feature F35': -0.027724
Voting System	'Feature F43': 1305 points 'Feature F1': 1149 points 'Feature F34': 1093 points 'Feature F47': 1076 points	'Feature OF8': 399 points 'Feature F58': 378 points 'Feature F49': 330 points 'Feature F3g': 300 points

Table 50: Features for SFST Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F41': 0.703613 'OF19': 0.405519	'OF20': -0.055471 'F48': -0.012283
DecisionTreeRegressor	'F41': 0.933778 'OF19': 0.555612	'OF30': -0.028011 'OF11': -0.001123
GradientBoostingRegressor	'F41': 0.877237 'OF19': 0.507981	'OF5': -0.004187 'F65': -0.001427
RandomForestRegressor	'F41': 0.949723 'OF19': 0.528025	'F3c': -0.001661 'OF11': -0.000769
AdaBoostRegressor	'F41': 1.009670 'OF19': 0.518803	'F17': -0.000691 'OF30': -0.000016
KNeighborsRegressor	'F5': 0.288869 'OF14': 0.000170	'OF27': -0.042150 'OF26': -0.000768
MLPRegressor	'F41': 0.069968 'F45': 0.054308	'OF27': -0.066358 'OF16': -0.048331
ElasticNet	'F43': 0.073112 'F41': 0.045888	'OF5': -0.010997 'OF13': -0.001769
SVR	'OF27': 0.010856 'F5': 0.009396	'OF5': -0.000006 'F3a': -0.000004
BayesianRidge	'F41': 0.154028 'OF21': 0.064929	'OF25': -0.018223 'F15': -0.013258
KernelRidge	'F41': 0.691473 'OF19': 0.396402	'OF20': -0.043222 'F48': -0.011175
LinearRegression	'F41': 0.751667 'OF19': 0.556543	'OF20': -0.031105 'S2': -0.012739
RANSACRegressor	'F41': 0.850172 'OF19': 0.528539	'OF21': -0.031337 'F48': -0.030173
TheilSenRegressor	'F41': 0.777552 'OF19': 0.557333	'OF20': -0.040786 'F48': -0.007471
Average Importance	'Feature F41': 0.558198 'Feature OF19': 0.331069 'Feature OF21': 0.124913 'Feature F5': 0.023782	'Feature OF7': -0.003062 'Feature OF16': -0.003171 'Feature F48': -0.004729 'Feature OF20': -0.008562
Voting System	'Feature F41': 1351 points 'Feature OF19': 1285 points 'Feature OF21': 1119 points 'Feature OF24': 1037 points	'Feature OF25': 478 points 'Feature F33': 478 points 'Feature F13': 472 points 'Feature F20': 424 points

Table 51: Features for PR Benefit Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'OF19': 0.671544 'OF21': 0.468303	'OF20': -0.035512 'F62': -0.005597
DecisionTreeRegressor	'OF19': 0.844500 'OF21': 0.403633	'OF7': -0.000485 'OF24': -0.000437
GradientBoostingRegressor	'OF19': 0.821696 'OF21': 0.666593	'OF3': -0.001737 'F52': -0.001574
RandomForestRegressor	'F41': 0.813527 'F12': 0.655125	'OF18': -0.076716 'OF23': -0.049900
AdaBoostRegressor	'OF19': 0.806301 'OF21': 0.631670	'F12': -0.000092 'OF30': -0.000044
KNeighborsRegressor	'F5': 0.184602 'OF13': 0.000152	'OF27': -0.019385 'F35': -0.000665
MLPRegressor	'F24': 0.038007 'OF19': 0.036784	'OF5': -0.044193 'OF20': -0.043179
ElasticNet	'OF27': 0.043120 'F34': 0.041926	'OF13': -0.000296 'OF2': 0.000000
SVR	'OF27': 0.027782 'F5': 0.006478	'OF5': -0.000007 'F3a': -0.000003
BayesianRidge	'OF21': 0.129610 'OF20': 0.092130	'F15': -0.013227 'OF16': -0.012066
KernelRidge	'OF19': 0.660613 'OF21': 0.500996	'OF20': -0.022330 'OF38': -0.003980
LinearRegression	'OF19': 0.898653 'OF21': 0.714826	'OF20': -0.023416 'S2': -0.008183
RANSACRegressor	'OF19': 0.906854 'F41': 0.409029	'F32': -0.004250 'OF22': -0.002480
TheilSenRegressor	'OF19': 0.891270 'F41': 0.396657	'F33': -0.004966 'F34': -0.003339
Average Importance	'Feature OF19': 0.497638 'Feature F41': 0.294398 'Feature OF21': 0.268613 'Feature F12': 0.045374	'Feature OF22': -0.001403 'Feature OF5': -0.001777 'Feature F20': -0.001792 'Feature OF18': -0.002347
Voting System	'Feature OF19': 1294 points 'Feature F41': 1231 points 'Feature OF21': 1143 points 'Feature F1': 1056 points	'Feature S1': 430 points 'Feature S2': 407 points 'Feature F68': 354 points 'Feature F20': 269 points

Table 52: Features for SR Benefit Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F45': 0.354192 'F43': 0.223956	'OF38': -0.029008 'OF33': -0.021786
DecisionTreeRegressor	'F43': 1.823320 'F25': 0.050814	'OF22': -0.006792 'OF28': -0.006266
GradientBoostingRegressor	'F43': 1.535820 'OF22': 0.023043	'F23': -0.000647 'F65': -0.000644
RandomForestRegressor	'F43': 1.094504 'F44': 0.097645	'F31': -0.001984 'F25': -0.001022
AdaBoostRegressor	'F43': 1.700848 'F22': 0.019314	'OF2': -0.002059 'S4': -0.001717
KNeighborsRegressor	'F5': 0.430873 'OF27': 0.118516	'OF13': -0.000089 'F41': 0.000000
MLPRegressor	'F43': 0.150892 'F45': 0.147232	'F3d': -0.018913 'F17': -0.014442
ElasticNet	'F43': 0.426430 'F45': 0.233256	'OF18': -0.000013 'OF2': 0.000000
SVR	'OF27': 0.023293 'F5': 0.008348	'OF18': -0.000016 'F54': -0.000006
BayesianRidge	'F43': 0.310904 'F45': 0.239989	'OF38': -0.012652 'F54': -0.009201
KernelRidge	'F45': 0.349817 'F43': 0.231780	'OF38': -0.025551 'OF33': -0.022170
LinearRegression	'F45': 0.380422 'F44': 0.171212	'F3c': -0.042531 'OF38': -0.036337
RANSACRegressor	'F45': 2.335171 'F24': 1.789909	'OF31': -0.330208 'F3c': -0.245296
TheilSenRegressor	'F45': 0.448860 'F52': 0.173472	'F3c': -0.036481 'OF38': -0.032400
Average Importance	'Feature F43': 0.563853 'Feature F45': 0.320945 'Feature F24': 0.140763 'Feature OF21': 0.114374	'Feature F58': -0.013924 'Feature OF38': -0.021798 'Feature F3c': -0.026023 'Feature OF31': -0.028798
Voting System	'Feature F43': 1365 points 'Feature F45': 1260 points 'Feature F31': 1135 points 'Feature F24': 1130 points	'Feature F21': 426 points 'Feature S2': 384 points 'Feature OF34': 377 points 'Feature OF2': 255 points

Table 53: Features for SR Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'OF10': 0.434765 'OF19': 0.414841	'F29': -0.029499 'OF33': -0.025170
DecisionTreeRegressor	'OF10': 0.796662 'OF19': 0.563827	'F51': -0.001892 'OF23': -0.000782
GradientBoostingRegressor	'OF10': 0.671728 'OF19': 0.464892	'F65': -0.004832 'OF3': -0.003717
RandomForestRegressor	'OF10': 0.836535 'OF19': 0.444950	'OF3': -0.003660 'OF5': -0.003092
AdaBoostRegressor	'OF10': 0.708734 'OF19': 0.393413	'OF3': -0.009379 'OF5': -0.007220
KNeighborsRegressor	'F5': 0.327369 'OF27': 0.003880	'OF26': -0.002144 'F24': -0.001972
MLPRegressor	'F35': 0.309029 'F34': 0.088340	'OF27': -0.878888 'F3g': -0.151826
ElasticNet	'OF9': 0.106174 'OF10': 0.081926	'OF5': -0.039847 'OF27': -0.021988
SVR	'F5': 0.012487 'OF9': 0.000024	'OF27': -0.005404 'OF5': -0.000020
BayesianRidge	'OF10': 0.142219 'OF9': 0.092250	'OF3': -0.025231 'OF5': -0.021868
KernelRidge	'OF10': 0.424293 'OF21': 0.421672	'F29': -0.034079 'OF33': -0.024907
LinearRegression	'OF21': 0.607966 'OF19': 0.595742	'OF33': -0.018405 'F3g': -0.016241
RANSACRegressor	'OF10': 0.845077 'F50': 0.680531	'OF20': -0.141968 'OF21': -0.098952
TheilSenRegressor	'OF19': 0.630024 'OF21': 0.585880	'OF33': -0.017230 'F7': -0.015492
Average Importance	'Feature OF10': 0.417809 'Feature OF19': 0.312239 'Feature OF21': 0.205786 'Feature F41': 0.075133	'Feature OF6': -0.007846 'Feature OF25': -0.008854 'Feature F3g': -0.014606 'Feature OF27': -0.065078
Voting System	'Feature OF10': 1342 points 'Feature OF19': 1236 points 'Feature F41': 1198 points 'Feature OF21': 1189 points	'Feature F25': 433 points 'Feature OF3': 372 points 'Feature OF5': 358 points 'Feature F36': 356 points

Table 54: Features for NR Benefit Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'OF17': 0.901439 'OF23': 0.333440	'OF37': -0.087702 'F20': -0.063504
DecisionTreeRegressor	'OF17': 1.460172 'OF23': 0.207532	'F23': -0.006009 'OF5': -0.003486
GradientBoostingRegressor	'OF17': 1.489140 'OF23': 0.131793	'F41': -0.003486 'F65': -0.002410
RandomForestRegressor	'F53': 7.896065 'OF24': 0.623784	'OF18': -2.847741 'OF22': -0.226609
AdaBoostRegressor	'OF17': 1.300579 'OF23': 0.082517	'F21': -0.001078 'OF34': -0.000603
KNeighborsRegressor	'OF27': 0.359802 'F5': 0.237931	'F35': -0.000053 'F45': -0.000035
MLPRegressor	'F2': 2.912182 'F6': 0.921916	'F5': -9.126598 'F3c': -2.098251
ElasticNet	'OF17': 0.411531 'F54': 0.084062	'F5': -0.020583 'F31': -0.001072
SVR	'OF27': 0.028316 'OF5': 0.000032	'F5': -0.001905 'OF18': -0.000007
BayesianRidge	'OF17': 0.851256 'OF23': 0.199564	'F20': -0.033465 'OF11': -0.023769
KernelRidge	'OF17': 0.835658 'F24': 0.376621	'OF37': -0.072615 'F20': -0.055666
LinearRegression	'OF17': 0.888735 'OF23': 0.385073	'OF37': -0.178323 'OF21': -0.095820
RANSACRegressor	'OF20': 3.351361 'OF21': 3.174862	'S2': -0.562348 'OF6': -0.521871
TheilSenRegressor	'OF17': 0.850844 'OF23': 0.332914	'OF37': -0.128644 'F20': -0.045430
Average Importance	'Feature OF17': 0.671066 'Feature F53': 0.620027 'Feature OF20': 0.235132 'Feature OF21': 0.218170	'Feature F33': -0.103096 'Feature F3c': -0.145086 'Feature F3d': -0.161077 'Feature F5': -0.651739
Voting System	'Feature OF17': 1307 points 'Feature F52': 1237 points 'Feature OF23': 1212 points 'Feature F51': 1189 points	'Feature F64': 381 points 'Feature F67': 376 points 'Feature F33': 369 points 'Feature F5': 323 points

Table 55: Features for WS Benefit Models

Model	Top 2 Important Features	Bottom 2 Important Features
Ridge	'F43': 1.068454 'F50': 0.068869	'OF20': -0.020697 'OF11': -0.012037
DecisionTreeRegressor	'F43': 1.323816 'OF28': 0.159743	'F65': -0.007496 'OF16': -0.006533
GradientBoostingRegressor	'F43': 1.196512 'OF28': 0.082922	'F47': -0.001079 'F18': -0.000807
RandomForestRegressor	'OF28': 0.538508 'OF22': 0.071115	'F43': -1.649695 'OF18': -0.549475
AdaBoostRegressor	'F43': 0.969779 'OF28': 0.094944	'OF5': -0.004574 'F3f': -0.002689
KNeighborsRegressor	'F5': 0.763124 'OF27': 0.157415	'OF13': -0.000378 'OF9': -0.000315
MLPRegressor	'F45': 1.611972 'F3b': 0.504369	'F3a': -0.517319 'OF2': -0.487867
ElasticNet	'F43': 0.821764 'OF18': 0.086786	'OF2': 0.000000 'F39': 0.000000
SVR	'F5': 0.059692 'OF27': 0.027247	'F3g': -0.000010 'OF2': -0.000007
BayesianRidge	'F43': 0.769779 'OF18': 0.039686	'OF20': -0.017260 'OF11': -0.013994
KernelRidge	'F43': 1.045855 'F50': 0.051678	'OF20': -0.029024 'OF11': -0.014729
LinearRegression	'F43': 1.074091 'F50': 0.103631	'S2': -0.015593 'OF20': -0.012721
RANSACRegressor	'F43': 2.910780 'F45': 0.534127	'F9': -0.129027 'F23': -0.081989
TheilSenRegressor	'F43': 1.144618 'F6': 0.082759	'OF21': -0.062422 'OF20': -0.053491
Average Importance	'Feature F43': 0.755861 'Feature F45': 0.141279 'Feature F5': 0.083083 'Feature OF28': 0.061002	'Feature F3e': -0.012449 'Feature F3d': -0.015819 'Feature OF13': -0.020652 'Feature F3a': -0.025738
Voting System	'Feature F43': 1158 points 'Feature OF18': 1122 points 'Feature F1': 1100 points 'Feature OF28': 1080 points	'Feature F62': 425 points 'Feature S2': 416 points 'Feature F22': 366 points 'Feature F20': 362 points

Table 56: Features for SFST Benefit Models