Exploring Supervised Learning: A Comparative Analysis on Wine Quality and Breast Cancer

Mar Tejedor Ninou
CS7641 Spring 2024

I. Introduction

This paper explores supervised learning techniques focusing on the exploration and comparison of five different learning algorithms applied to two distinct classification problems: White Wine Quality and Wisconsin Diagnostic Breast Cancer (WDBC). The objective is to analyze the behavior of Decision Trees, Neural Networks, Boosted Decision Trees, Support Vector Machines, and k-Nearest neighbours on these datasets, emphasizing the significance of experimental analysis and hyperparameters tuning. Through this exploration, we aim to provide valuable insights into the behavior of different supervised learning algorithms and their applicability to diverse classification problems.

II. Datasets

The foundation of this research lies in the analysis of two carefully chosen datasets: the White Wine Quality and Wisconsin Diagnostic Breast Cancer (WDBC) datasets.

A. White Wine Quality Dataset

This dataset, with its 4898 samples and 11 features, is interesting for its potential to predict the quality of white wine based on its physicochemical properties. By manipulating this dataset, our analysis aims to uncover how well algorithms can discern and predict the taste quality of wine. The dataset's moderate size and feature dimensionality make it an ideal candidate for assessing algorithmic performance, allowing us to observe how different algorithms respond to varied physical and chemical attributes of wine.

B. Wisconsin Diagnostic Breast Cancer (WDBC) Dataset

In contrast, the WDBC dataset with 569 samples and 32 features focuses on predicting whether a tumor is benign or malignant based on its physical properties. The complexity arises from the higher dimensionality and smaller sample size compared to the White Wine Quality dataset. This dataset provides a challenging scenario to observe how algorithms use physical attributes to accurately classify tumors. Additionally, both datasets share the commonality of being imbalanced, adding a layer of complexity to our analysis.

C. Justification for Dataset Selection

The chosen datasets, White Wine Quality and Wisconsin Diagnostic Breast Cancer (WDBC), present unique characteristics that contribute to a comprehensive analysis of algorithmic behaviors. The variation in sample sizes (4898 vs. 569) and feature dimensions (11 vs. 32) introduces a dynamic landscape, allowing us to assess how algorithms adapt to different dataset structures. The inherent imbalance in both datasets adds a layer of realism to our analysis, mirroring challenges encountered in real-world applications. This shared characteristic enables us to identify commonalities in algorithmic responses to class imbalances.

D. Preprocessing

In preparing the datasets for model training, a key consideration was transforming the target variable to facilitate binary classification. For the Wine Quality dataset, the original target variable denoted wine quality on a scale from 1 to 10. To convert this into a binary classification task, a new column was created, categorizing wines as either ”Good” or ”Bad” quality. Wines with quality ratings between 1 and 5 (excluding 5) were categorized as ”Bad,” while those with quality ratings between 5 and 10 (inclusive) were labeled as ”Good.”

Conversely, the Breast Cancer dataset presented a straightforward case for binary classification. The dataset already included a target variable indicating the nature of detected cancers. The labels ”B” and ”M” represented benign and malignant tumors, respectively. No further transformation was needed in this case, as the dataset inherently supported a binary classification problem.

E. Performance Metric: F1 Score

In the analysis, the F1 score serves as the primary performance metric, strategically chosen for its efficacy in evaluating machine learning models, especially in the context of imbalanced datasets. The F1 score offers a delicate evaluation by achieving a balance between precision and recall, making it particularly valuable in scenarios where class imbalances are prevalent.

Unlike traditional accuracy metrics, the F1 score provides a comprehensive assessment by considering both false positives and false negatives. This characteristic is crucial in situations where misclassifying instances from the minority class can significantly impact model effectiveness. By emphasizing the minimization of errors in both directions, the F1 score addresses the asymmetry present in imbalanced datasets.

III. Methodology

A. Training a Validation Sizes

In the model development process, a consistent approach was adopted for partitioning the datasets into training and validation sets. Specifically, the training set comprised 70% of the samples, while the remaining 30% served as the test set across all algorithms and models.

The 70-30 split provides balance between providing an ample amount of data for training, allowing models to capture underlying patterns, and reserving a substantial portion for testing to generalize to unseen instances. This division aims to prevent overfitting by ensuring that models are not only build by the training data but also exhibit robustness in predicting outcomes for new samples.

B. Cross Validation (CV)

A cross-validation strategy with 5 folds was employed during all the training phases. This strategy provides a more comprehensive assessment of model performance, reducing the impact of data variability, and providing more stable and representative metrics. The decision to CV aligns with the characteristics of the datasets since the

Wine Quality and Breast Cancer datasets, being of small and moderate size, benefit from the balance by cross-validation. Moreover, this approach mitigates the risk of overfitting to a specific training-validation split, ensuring that the models' performance metrics are indicative of their ability to generalize to unseen data.

C. *Grid Search for Optimal Hyperparameters*

At the conclusion of each algorithm, a Grid Search was employed to provide optimal hyperparameters. Grid Search systematically explores hyperparameter combinations, selecting the configuration that maximizes the chosen metric. This approach eliminates guesswork and ensures a comprehensive search for optimal settings. Grid Search is particularly effective for our datasets, where varied characteristics demand specific hyperparameter choices. By conducting Grid Search at the end of each algorithm, we guarantee that the selected hyperparameters align with model requirements, enhancing performance, and adapting to dataset. This allows us to compare our optimal hyperparameters with the ones selected by Grid Search.

IV. Decision Trees (DT)

A. *Pruning: Max Depth*

In our study, we take advantage of pruning through the control of the maximum depth in decision trees. This strategic approach offers multiple benefits, including the reduction of overfitting by preventing the tree from becoming excessively complex and adapting itself too closely to the training data. By limiting the tree's depth, we enhance computational efficiency, ensuring quicker training and inference times, particularly crucial for large datasets and real-time applications.

B. *Criterion: Entropy*

The adoption of entropy as the criterion for decision tree construction is grounded in its efficacy in quantifying impurity and disorder within datasets. This choice is well-suited for the Wine Quality and Breast Cancer datasets, where binary classification tasks are used. Generally, entropy is more sensitive to changes in class proportions in imbalanced datasets. However, it is a good choice when you have imbalanced classes and want the tree to be more sensitive to minority class instances. Also, decision trees using entropy as the criterion usually result in more balanced and equally distributed decision boundaries.

C. *Hyperparameters*

For our datasets, two key hyperparameters, max depth and splitter, have been strategically chosen. As explained before max depth will be used for pruning and splitter will provide different results when splitting nodes. A deeper tree might capture intricate patterns in the training data, but it runs the risk of overfitting. In our case, selecting an optimal max depth provide us a balance between model complexity and generalization. The splitter hyperparameter determines the strategy for selecting the best feature to split a node. The options include 'best' (choosing the best split based on impurity reduction) and 'random' (choosing the best random split). By opting for 'best' in our case, we prioritize making informed splits, against 'random' which allows uninformed splittings.

D. *Learning Curve*

To assess the performance and generalization capabilities of our decision tree models, we conducted a learning curve analysis by varying the size of the training dataset while keeping the optimal hyperparameter choices constant. After tuning hyperparameters and analyzing its results, for the Wine Quality dataset the optimal hyperparameters were set to a maximum depth of 15 and 'best' splitter, and for the Breast Cancer dataset, with a maximum depth of 3 and a 'random' splitter.

For the Wine Quality dataset, Figure 1, characterized by a low bias scenario, the learning curve exhibited a low training error, indicating that the model captures the underlying patterns well. However, a notable gap between the training and testing curves suggested a high variance. The model demonstrated a strong dependence on the specific training instances, and adding more training data is likely to improve performance by reducing variance and achieving a more accurate generalization.

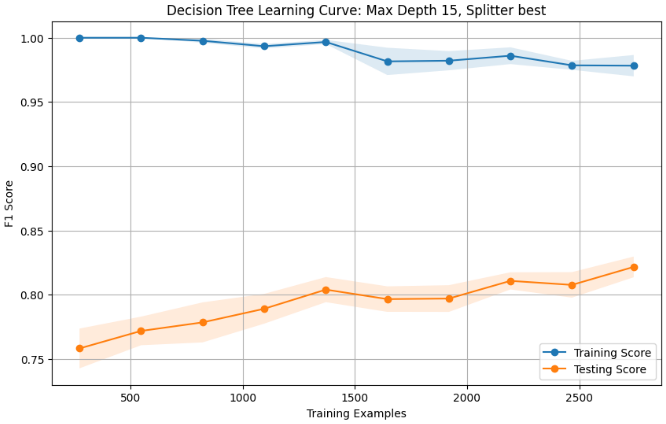

Fig. 1. Wine Quality Decision Tree Learning Curve.

In contrast, the learning curve for the Breast Cancer dataset Figure 2, with its high bias configuration, displayed a convergence of the training and testing curves. This convergence indicated a low variance, suggesting that the model's performance was less dependent on the specific training instances. The small gap between the curves across the entire range of data sizes implied that the model might benefit from increased complexity or more training samples to reduce bias. Smoothing the learning curve through additional data would likely enhance the model's ability to capture underlying patterns in the Breast Cancer dataset.

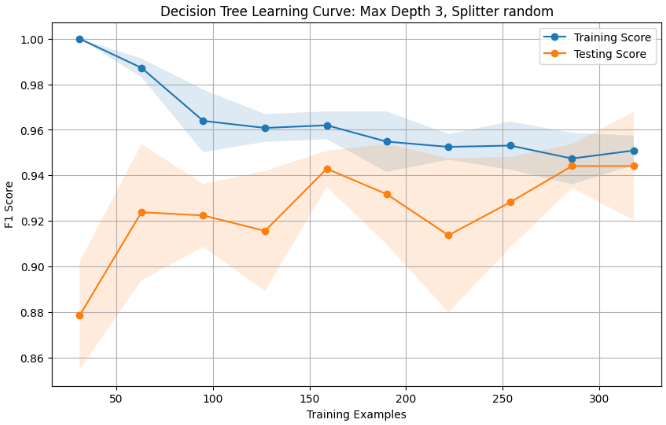

Fig. 2. Breast Cancer Decision Tree Learning Curve.

E. Validation Curve

In our investigation of decision tree models, we performed a validation curve analysis by varying the pruning hyperparameter, specifically the max depth, while keeping all other optimal hyperparameters fixed ('best' and 'random' splitter). Pruning, achieved through controlling the maximum depth of the decision tree, was a crucial consideration in this analysis. For both Wine Quality and Breast Cancer datasets, Figure 3 and Figure 4, the validation curves initially exhibited high bias, indicating suboptimal model performance with a minimal depth. As the max depth increased, the bias decreased, and the model's ability to capture underlying patterns improved. However, at a certain point, the curves displayed diminishing returns, signaling the onset of high variance. In the case of both datasets, this behavior was observed at the end of the max depth range, indicating that overly complex models were not significantly improving on the training set but were leading to overfitting.

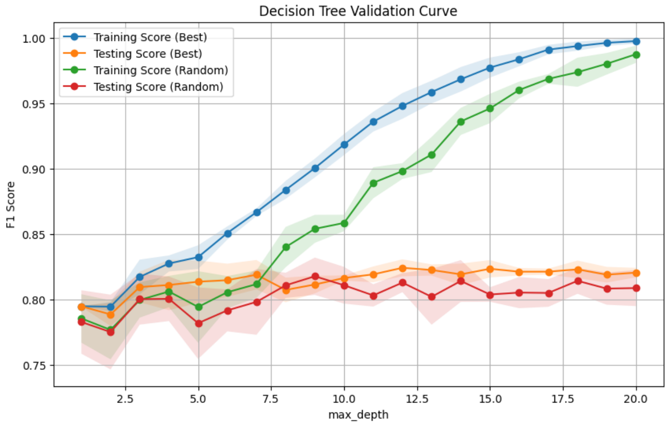

Fig. 3. Wine Quality Decision Tree Validation Curve.

Notably, the choice of splitter had a discernible impact on the smoothness of the validation curves. The 'best' splitter resulted in smoother curves for both datasets, suggesting a more gradual transition between bias and variance. This smoother progression indicated that the model's performance was consistently improving with increased max depth, without abrupt fluctuations.

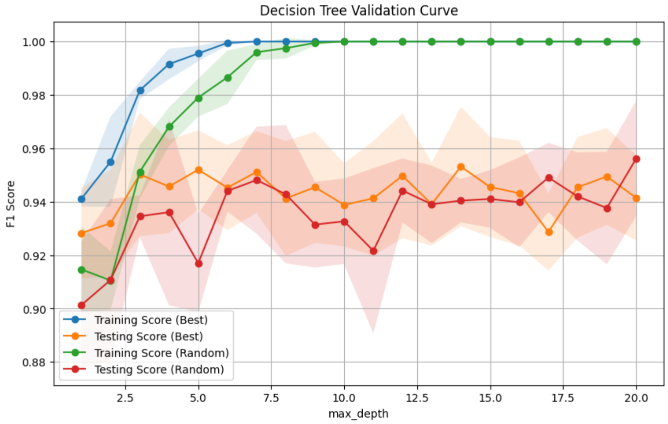

Fig. 4. Breast Cancer Decision Tree Validation Curve.

F. Analysis and Comparison

In our exploration of decision tree models, we observe distinct behaviors influenced by dataset characteristics and hyperparameter tuning. Notably, the relationship between dataset size and maximum tree depth becomes apparent, impacting the model's ability to generalize and susceptibility to overfitting.

For datasets with fewer samples, a lower maximum depth often leads to optimal performance, preventing rapid overfitting. The reasoning lies in the trade-off between model complexity and generalizability—the deeper the tree, the more it adapts itself to the training data, potentially sacrificing performance on unseen instances. This principle is particularly evident when examining the performance of decision trees on datasets with varying sample sizes.

Interestingly, the Breast Cancer dataset, despite having fewer samples, performs remarkably well. This can be attributed to the dataset's richness in features. With more attributes available for tree splits, the model gains greater discriminatory power, allowing it to effectively capture underlying patterns in the data.

Furthermore, the impact of the splitter choice becomes apparent, especially in datasets with a substantial number of features. In the context of large feature spaces, the random splitter demonstrates comparable performance to the best splitter. This is due to the abundance of attributes, providing ample opportunities for effective tree splitting regardless of the starting point.

To fine-tune our decision tree models, we conducted a Grid Search shown in Table I, confirming that the chosen hyperparameters yield optimal results. Both datasets demonstrate suitability for decision tree models, with the optimized hyperparameters ensuring robust performance. This observation underscores the versatility of decision trees across datasets of varying sizes and characteristics.

TABLE I
DECISION TREE GRID SEARCH PERFORMANCE

Grid Search	**White Wine**	**Breast Cancer**
Best Max Depth	15	4
Best Splitter	Best	Random
Best Accuracy	0.907	0.973
Wall Time	11.280	5.602

V. NEURAL NETWORKS (NN)

A. Hyperparameters

Configuring Neural Networks for our datasets involved pivotal choices in two key hyperparameters: hidden layer sizes and activation. The Hidden Layer Sizes parameter dictates the network's structure, influencing its ability to discern specific patterns. By experimenting with different sizes, we aim to find the right balance, ensuring the network's complexity aligns optimally with our dataset. Simultaneously, the Activation Function choice introduces non-linearity to the network. Selecting functions like 'relu' or 'logistic' allows us to capture the non-linear relationships in our data.

B. Learning Curve

The learning curve for the Wine Quality dataset exhibited a peculiar pattern, Figure 5. Initially, the model's performance seemed to worsen through the first half of the training samples shown in the testing line, raising concerns of overfitting. However, as the dataset size increased, the score showed a substantial improvement. The curve's early convergence may suggest a rapid fitting to noise. Further analysis using the error rate per iteration graph, Figure 6, revealed a noteworthy reduction in error with additional iterations. Yet, a saturation point was observed, indicating that more iterations might enhance the model's performance.

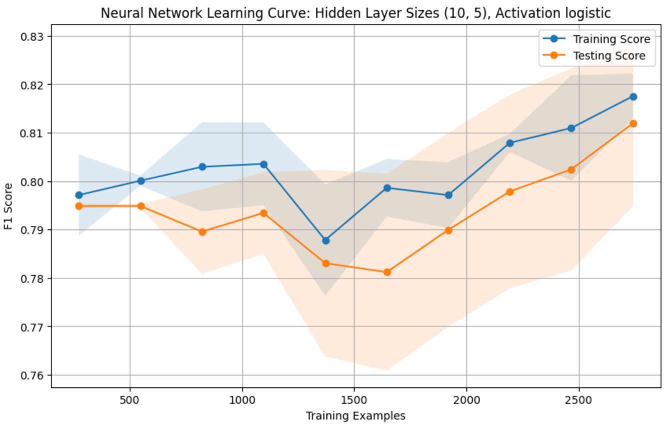

Fig. 5. Wine Quality Neural Network Learning Curve.

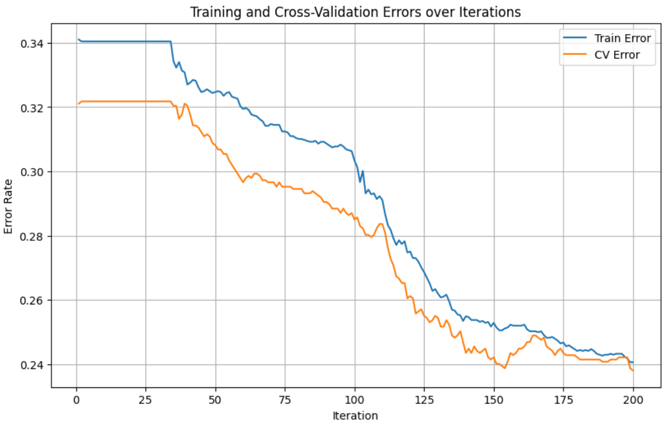

Fig. 6. Wine Quality Neural Network Error per Iteration Curve.

The learning curve for the Breast Cancer dataset displayed interesting dynamics, Figure 7. The model's improvement settled after processing more than 200 samples, indicating a potential high bias at the beginning. Notably, the training set score initiated at a high level, suggesting a initial biased model too. However, the variance at the end is very low which predicts well performance if we added more unseen data. Examining the error rate per iteration, Figure8, revealed a less smooth trajectory compared to the other datasets. The fluctuations in error rates indicated that more iterations did not consistently lead to lower errors, hinting at potential challenges like getting stuck in local minima. However, beyond 80 iterations, the error approached zero, underscoring a well-trained model. Adjusting the maximum iteration parameter to 80-100 appeared viable for this dataset.

C. Validation Curve

Validation curves for the Quality Wine dataset revealed interesting patterns, Figure 9. Comparing Logistic and ReLU activations, Logistic outperformed when the model was less complex. The peak testing score was observed with smaller hidden layer sizes (less than 20) and more than one hidden layer. Notably, the absence of significant variance suggested no apparent overfitting. This indicated that a simpler model structure and specific activation functions could lead to optimal performance in the Wine Quality dataset.

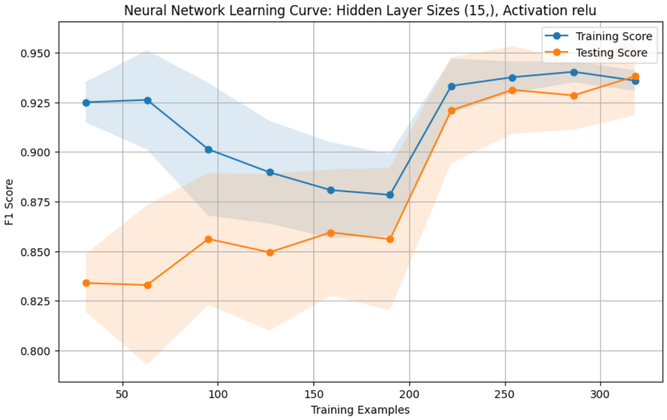

Fig. 7. Breast Cancer Neural Network Learning Curve.

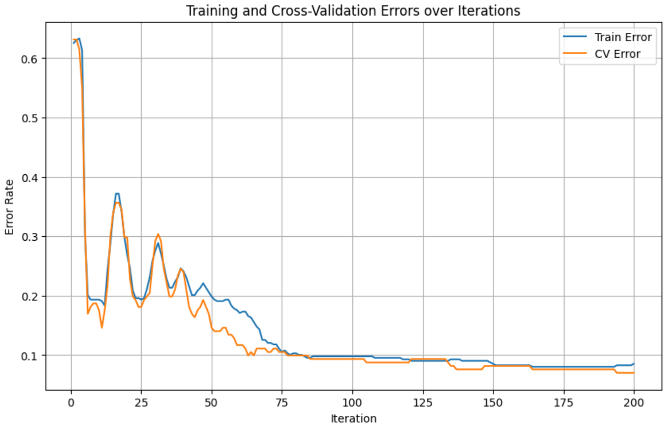

Fig. 8. Breast Cancer Neural Network Error per Iteration Curve.

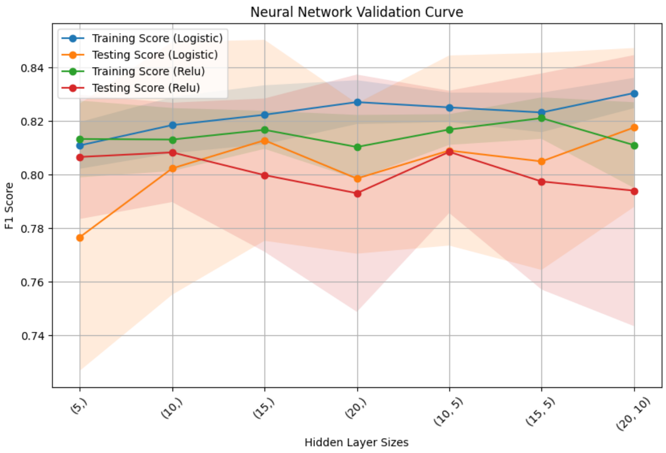

Fig. 9. Wine Quality Neural Network Validation Curve.

For the Breast Cancer dataset, the validation curves demonstrated clear trends, Figure 10. Logistic activation proved more unstable, with a huge drop in score (0.2) for a hidden layer size of (10,5). Generally, both activations showcased minimal variance and performed better with hidden layer sizes larger than 5 across different depths. The stability in performance indicated a robust model, particularly when hidden layer sizes were sufficiently large. This highlighted the importance of choosing suitable hyperparameters to ensure model stability and optimal performance.

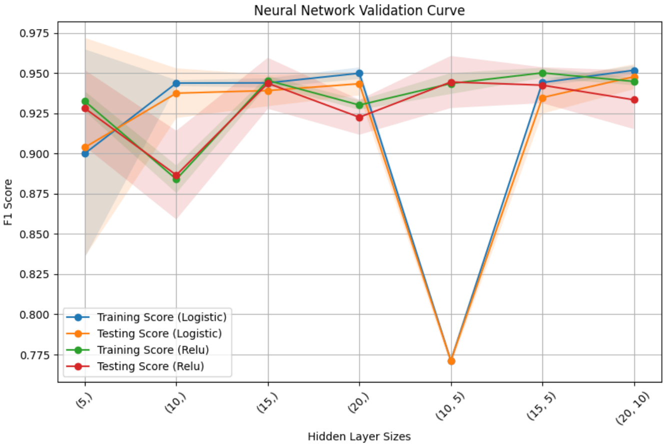

Fig. 10. Breast Cancer Neural Network Validation Curve.

D. Analysis and Comparison

For the Wine Quality dataset, the constraint of 200 maximum iterations poses a challenge, potentially hindering the neural network from reaching minimal error. The backpropagation algorithm updating the weights does not have enough iteration to find the optimal weights for each perceptron. This limitation likely contributes to the observed performance discrepancy compared to decision trees.

Turning our attention to the Breast Cancer dataset, the initial decline in performance indicates early overfitting, where the neural network captures noise or specific patterns in the small training subset. As more samples are introduced, the model adapts, leading to improved generalization updating the weights of the neural network. The chosen layer size (10,5) shows a clear example of limited representational capacity, the chosen layer size might not have enough capacity to effectively capture the underlying patterns in the data. In this case, increasing the layer size or adopting a different architecture might be beneficial.

Comparing neural networks with decision trees reveals noteworthy distinctions. Neural networks, with their flexibility, are advantageous for capturing intricate relationships but may be prone to overfitting, especially with limited data. Decision trees, characterized by simplicity, demonstrate effective generalization, particularly for certain datasets. The need for larger datasets to optimize neural network performance makes them more susceptible to overfitting with smaller datasets, while decision trees exhibit less sensitivity to dataset size.

The validation through Grid Search shown in Table II confirms the optimality of our initially chosen parameters for both datasets. The proportional difference in wall clock time aligns with expectations based on dataset training sizes.

In conclusion, our findings emphasize the critical role of parameter tuning and considerations of model complexity for neural networks. The delicate trade-offs between model complexity and dataset size become apparent in comparisons with decision trees. The robustness of our chosen parameters, validated through Grid Search, ensures their practicality for the specific challenges posed by the given datasets.

TABLE II
NEURAL NETWORK GRID SEARCH PERFORMANCE

Grid Search	**White Wine**	**Breast Cancer**
Best Layers Size	(10,5)	(15,)
Best Activation	Logistic	Relu
Best Accuracy	0.7554	0.9173
Wall Time	267.668	43.1041

VI. BOOSTING

A. Type of Boosting: Ada Boosting

AdaBoost, or Adaptive Boosting, is a highly regarded boosting algorithm for its versatility, adaptability to weak classifiers, and effectiveness in handling imbalanced datasets. It is for this reason that we chose it as the boosting for our project. Its iterative approach focuses on correcting the errors of weak learners by assigning different weights to instances, reducing overfitting and improving generalization. This properties will provide good analysis fro both of our datasets.

B. Hyperparameters

In configuring AdaBoosting for our dataset, we focused on two key hyperparameters: number of estimators and learning rate. The former dictates the number of decision tree weak learners in the ensemble, striking a balance between complexity and generalization. A lower learning rate emphasizes a gradual learning process, enhancing stability and reliability. Additionally, we kept the max depth of decision trees consistently set to 1. This deliberate choice ensures the ensemble comprises shallow learners, preventing overfitting and enhancing generalization capabilities.

C. Learning Curve

In our exploration of AdaBoost models, we conducted a learning curve analysis by maintaining optimal hyperparameter choices and varying the size of the training dataset. For the Wine Quality dataset, the learning curve analysis revealed low bias and low variance, Figure 11. The model demonstrated exceptional performance even with a smaller training dataset, indicating a robust ability to capture the underlying patterns. The consistent convergence of the training and testing curves suggested that the model generalized well across varying dataset sizes, showcasing the effectiveness of AdaBoost in this low-bias, low-variance scenario.

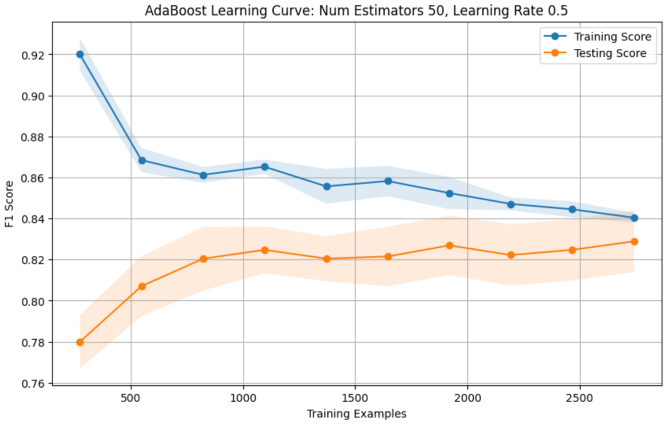

Fig. 11. Wine Quality AdaBoost Learning Curve.

In contrast, the learning curve for the Breast Cancer dataset exhibited high bias (Figure 12), with the training score remaining consistently perfect, indicating overfitting. The absence of errors in the training set implied that additional training instances would likely improve the model's performance by further reducing overfitting. Interestingly, the testing set performed admirably even with the limited training data, displaying a consistent and positive trajectory in F1 score. This behavior indicated that while the model had overfitting and could benefit from additional training instances, it was already demonstrating strong generalization on the testing set.

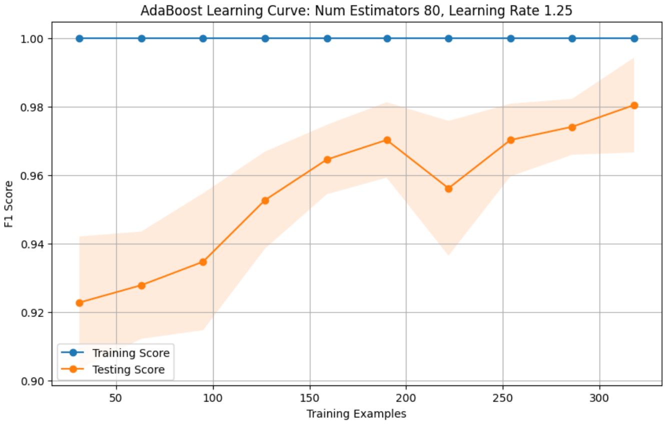

Fig. 12. Breast Cancer AdaBoost Learning Curve.

D. Validation Curve

For the Wine Quality dataset, varying the number of estimators revealed interesting patterns on the validation curve, Figure 13. The variance increased with the number of weak learners, indicating that excessively complex models led to diminishing returns. Notably, the testing set's performance peaked between 30 and 60 estimators, showcasing the optimal trade-off between bias and variance. On the other hand, Figure 14, altering the learning rate highlighted a relationship with variance; larger learning rates led to increased variance and a decline in overall performance. This finding emphasized the importance of carefully tuning the learning rate to strike a balance between bias and variance.

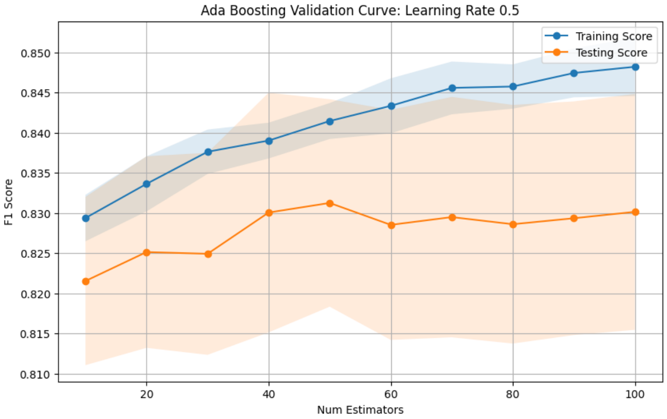

Fig. 13. Wine Quality AdaBoost Validation Curve Number of Estimators.

Turning our attention to the Breast Cancer dataset, both the number of estimators and learning rate validation curves exhibited substantial bias, Figure 15 and Figure 16. The training score remained perfect, indicating overfitting as we already saw on the learning curve. However, as the number of estimators and learning rate increased, the variance reduced marginally, providing a modest improvement in model generalization. Interestingly, the limited number of samples occasionally led to unpredictable outcomes, such as the shift between learning rates of 1 and 1.25. Even though overfitting on training data is observed, the testing results demonstrate adaptability, indicating that the model exhibits robust generalization capabilities despite the apparent overfitting on the training data.

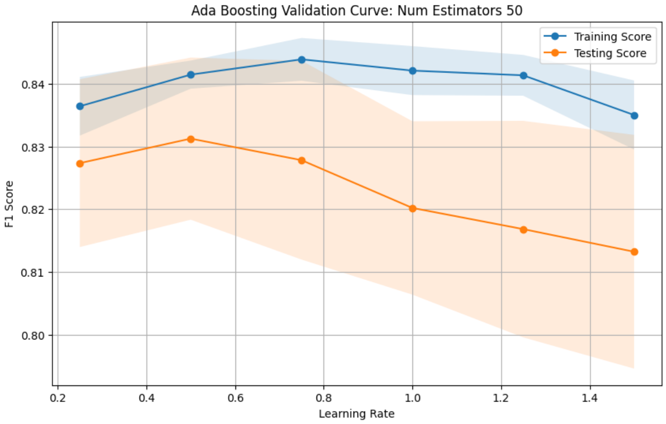

Fig. 14. Wine Quality AdaBoost Validation Curve Learning Rate.

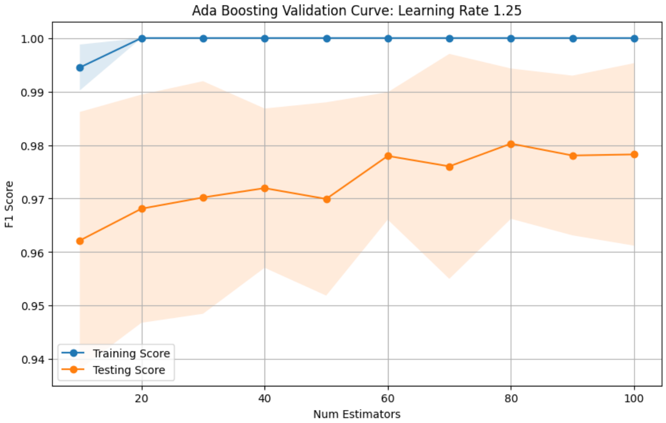

Fig. 15. Breast Cancer AdaBoost Validation Curve Number of Estimators.

E. Analysis and Comparison

In analyzing the AdaBoost algorithm's performance on the Wine Quality and Breast Cancer datasets, certain trends and behaviors emerge. AdaBoost's adaptive weighting mechanism becomes more pronounced with increased training data, assigning more significant emphasis to misclassified samples in subsequent iterations. This adaptability is especially crucial for focusing on challenging cases when the sample size is small and we need to improve overall model performance. We can see this in Figure 17, provided by Grid search, where the heat map for Wine Quality shows how smaller learning rate achieves slowly a better score. However, the performance on the Breast Cancer dataset with a limited training dataset of 300 samples introduces a different set of observations that contradicts the general perception that for a small dataset a

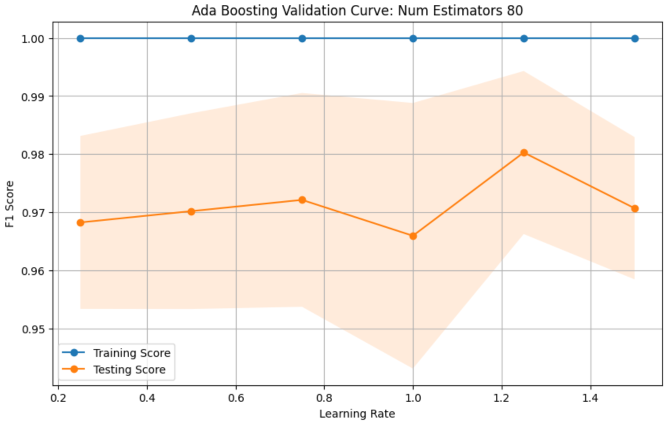

Fig. 16. Breast Cancer AdaBoost Validation Curve Learning Rate.

low number of estimators and low learning rate the model perform better, Figure 18. This is because, the aggressive weight updating of AdaBoost, particularly with a high learning rate, enables quick adaptation to the dataset's characteristics. This aggressiveness, along with a large number of estimators, to avoid underfitting, contributes to capturing intricate relationships in a more complex model. Emphasis on hard-to-classify instances is heightened with a high learning rate, significantly impacting challenging instances in small datasets.

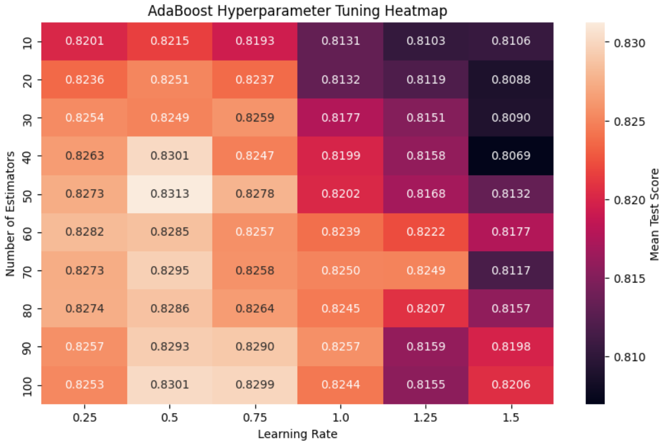

Fig. 17. Wine Quality AdaBoost Hyperperameter Tuning Heat Map.

Comparing AdaBoost to regular decision trees reveals specific insights. AdaBoost tends to outperform regular decision trees in scenarios with limited data, showcasing its adaptability. In medium-sized datasets, both methods can control overfitting, with AdaBoost potentially offering a slight advantage. However, performance in Wine Quality is much better with a single decision tree taking advantage of the max depth of the tree. Grid Search optimization validates the chosen parameters as optimal for achieving good accuracy in both datasets, Table III. The wall clock time considerations emphasize the trade-off between complexity and computational efficiency, with AdaBoost exhibiting intermediate time requirements compared to decision trees and neural networks.

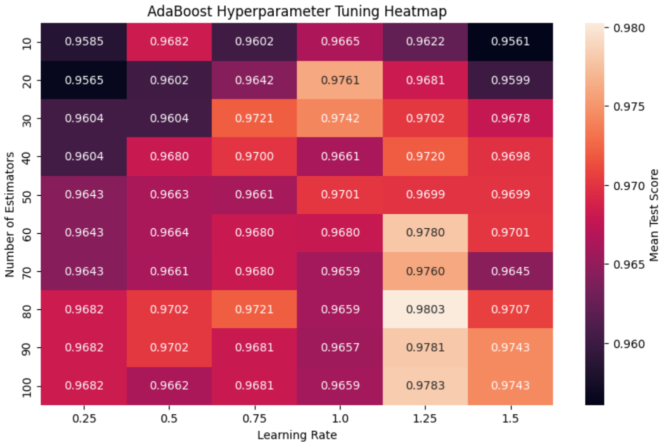

Fig. 18. Breast Cancer AdaBoost Hyperperameter Tuning Heat Map.

TABLE III
TABLE TYPE STYLES

Grid Search	White Wine	Breast Cancer
Best Number Estimators	100	80
Best Learning Rate	0.75	1.25
Best Accuracy	0.8313	0.9803
Wall Time	186.146	85.8866

VII. SUPPORT VECTOR MACHINES (SVM)

A. Hyperparameters

In adapting Support Vector Machines (SVM) to our dataset, we strategically chose two key hyperparameters: kernel and C (Cost). The kernel selection involved exploring diverse options—linear, polynomial, and radial basis function (RBF). Simultaneously, the choice of the C hyperparameter, governing the balance between a smooth decision boundary and accurate classification, was adjusted to navigate the dataset's details.

B. Learning Curve

The learning curve for the Quality Wine dataset unveiled a distinctive performance pattern, Figure 19. Initially, the training score showed substantial improvement with the first 1000 samples in the training set. However, beyond this point, a significant decline in performance occurred, indicating a pronounced case of overfitting. This overfitting trend was mirrored in the testing set, with a notable drop in score after 1000 examples. The occurrence of this overfitting phenomenon early in the training process led to subsequent underfitting, suggesting that the model struggled to generalize effectively to additional samples. Despite the initial improvement, the presence of high variance between the testing and training sets around the 1000-sample mark underscored challenges in achieving robust generalization.

In contrast, the learning curve for the Breast Cancer dataset showcased a different narrative, Figure 20. The model exhibited high bias and low variance, indicative of a well-fitted and stable performance. The convergence of the training and testing curves suggested that the model had reached an optimal state with the available data. The absence of a significant gap between the training and testing curves hinted that additional samples might not yield substantial improvements, implying that the current dataset size was

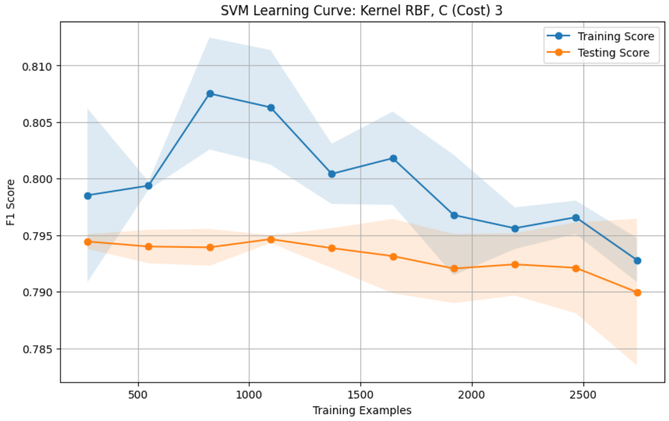

Fig. 19. Wine Quality Support Vector Machine Learning Curve.

adequate for training. This scenario emphasized the suitability of the dataset and the robustness of the SVM model.

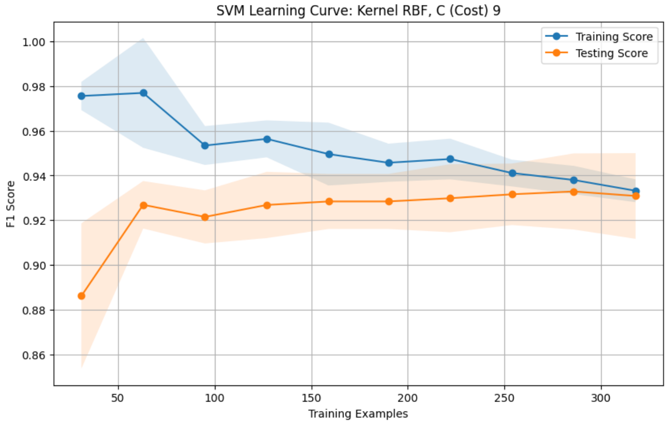

Fig. 20. Breast Cancer Support Vector Machine Learning Curve.

In summary, the SVM learning curve analyses provided detailed insights into the behavior of the model on both datasets. While the Wine Quality dataset exhibited challenges related to overfitting and subsequent underfitting, the Breast Cancer dataset showcased a well-fitted model with minimal variance. These findings contribute valuable considerations for refining model training strategies and understanding the dataset's adequacy in the SVM context.

C. Validation Curve

Validation curves for the Wine Quality dataset revealed intriguing patterns, Figure 21. The Linear kernel outperformed other types, showcasing better scores. The optimal model configuration emerged when training and testing scores were almost the same, indicating a well-fitted model. Importantly, the absence of gaps between the training and testing scores suggested low variance and no indications of overfitting or underfitting. This scenario pointed towards a stable and robust SVM model for the Wine Quality dataset.

In the Breast Cancer dataset, Figure 22, Linear kernel demonstrated better performance again, with higher variance than the dataset before. This scenario implied that the model was capturing the underlying patterns effectively. The convergence of training and testing scores suggested that additional complexity in the model did not yield substantial improvements. This finding aligns with the notion that the dataset size was adequate for the SVM model to generalize optimally. Both datasets showcased scenarios where an increase in model complexity increasing Cost did not led to simultaneous rises in significant improvement.

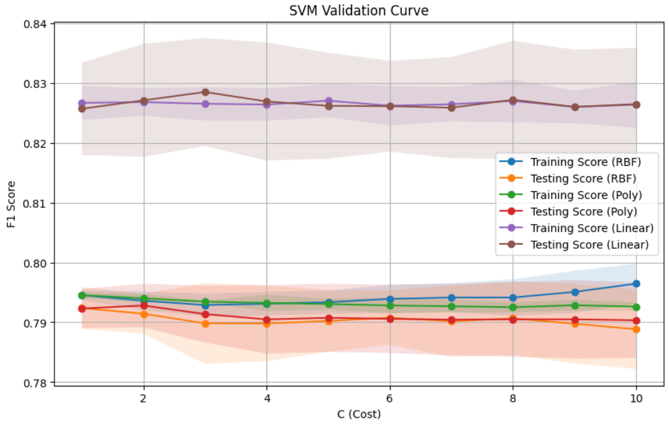

Fig. 21. Wine Quality Support Vector Machine Validation Curve.

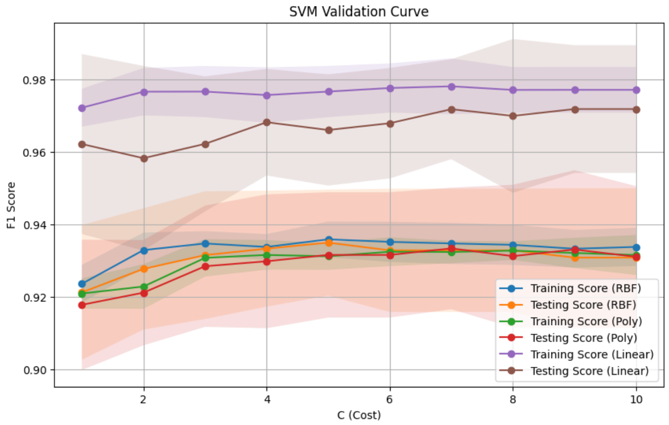

Fig. 22. Breast Cancer Support Vector Machine Validation Curve.

D. Analysis and Comparison

The SVM performance on the Wine Quality dataset exhibits early overfitting around 1000 samples, the reason can be attributed to the use of complex kernels, like BRF used for the learning curve. Kernels with high degrees or intricate transformations may lead to a decision boundary that closely adapts to the training instances. However, with more instances, the score drops significantly, indicating a lack of generalization. Notably, the validation curve reveals that a simple linear kernel outperforms others, emphasizing the effectiveness of a linear approach for this dataset.

For the Breast Cancer dataset, the learning curve shows good performance, but the validation curve suggests that a linear kernel provides a better option. This underscores the idea that smaller datasets imply simpler problems that a linear kernel can easily model. Additionally, in smaller datasets, a higher regularization parameter C proves beneficial. A higher C makes the SVM less regularized, enhancing flexibility to closely fit the training data, preventing underfitting, and better handling noise and outliers. The increased complexity afforded by higher C allows the SVM to capture nuanced relationships within smaller datasets.

TABLE IV
SVM Grid Search Performance

Grid Search	White Wine	Breast Cancer
Best Kernel	Linear	Linear
Best C	3	9
Best Accuracy	0.754	0.968
Wall Time	1331.76	145.777

Comparisons with Grid Search results, Table IV, confirm that a linear kernel is the optimal choice for both datasets. The analysis highlights the importance of choosing C based on dataset characteristics, with a higher C proving advantageous for smaller datasets. The wall clock time for SVM is notably higher than other algorithms due to computational complexity, involving solving a quadratic programming problem, and the kernel trick, especially with more complex kernels like RBF. However, because of the computational demands, SVM results do not surpass those of the other algorithms analyzed. The increased complexity may not be justified for these datasets, as Decision Trees and Boosting algorithms demonstrate comparable or better performance without the same computational overhead.

VIII. K-Nearest Neighbour (kNN)

A. Hyperparameters

In configuring the k-Nearest Neighbours (kNN) algorithm for our dataset, our focus centered on two critical hyperparameters: number of neighbours and weights. The choice of the number of neighbours is essential, our aim is to ensure that the model remains adaptable, considering enough neighbours for robust predictions without becoming overly sensitive. Similarly, the consideration of weights (Uniform and Distance) provides versatility in the model's approach.

B. Learning Curve

For the Wine Quality dataset, Figure 23, the learning curve revealed high variance, evident from the substantial gap between the training and testing sets. While the model demonstrated low bias, as indicated by the relatively high training score, the marginal decrease in the testing score toward the end of the samples suggested a limited potential for improvement with additional data. This behavior indicated that the model might not benefit significantly from an increase in the training dataset size, reflecting a trade-off between variance reduction and marginal performance gains.

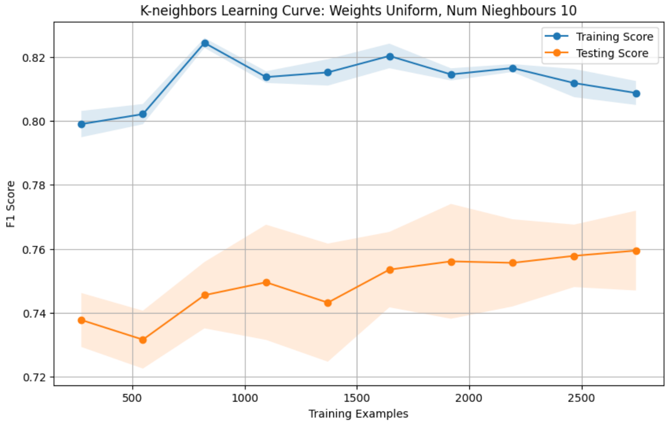

Fig. 23. Wine Quality K-Nearest Neighbour Learning Curve.

In contrast, the learning curve for the Breast Cancer dataset exhibited low variance, Figure 24, with a minimal gap between the training and testing sets. The model displayed both low bias and high testing performance, indicating robust generalization capabilities. The consistent and positive trajectory of the learning curve highlighted the model's ability to maintain excellent performance across varying training set sizes, suggesting that the dataset was sufficient for effective learning without introducing overfitting concerns.

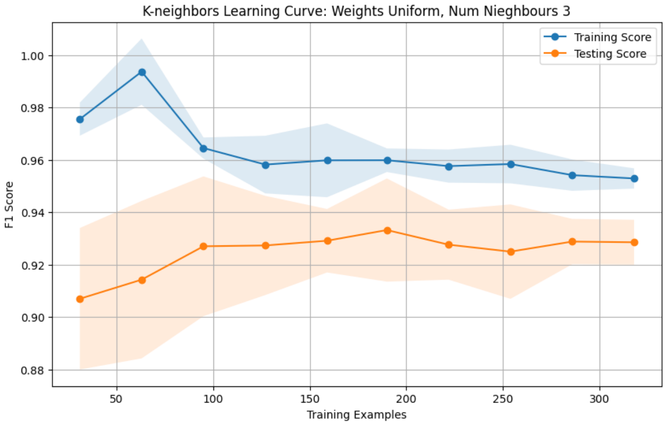

Fig. 24. Breast Cancer K-Nearest Neighbour Learning Curve.

C. Validation Curve

For both datasets, Figure 25 and Figure 26, employing distance weights in the kNN algorithm led to a noticeable overfitting pattern. Despite achieving a perfect score on the training set, indicating that the model perfectly memorized the training data, the performance on the testing set remained robust. This suggests that, while the model might be overfitting the training data, its generalization capabilities to unseen instances were not compromised.

The susceptibility of the kNN algorithm with distance weighting to overfitting, especially with small values of K, was evident in both datasets. The small size of the datasets exacerbated this effect, causing the algorithm to be overly sensitive to individual instances and leading to a perfect training score across different hyperparameter values. This behavior is a characteristic of the kNN algorithm, especially when the value of K is small, as it tends to capture noise and outliers in the training data.

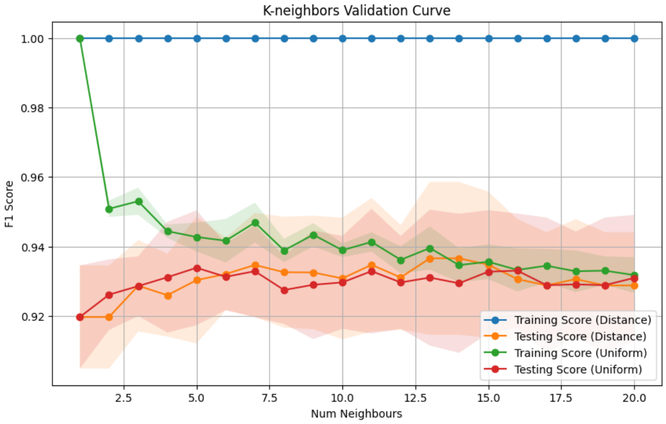

Fig. 25. Wine Quality K-Nearest Neighbour Validation Curve.

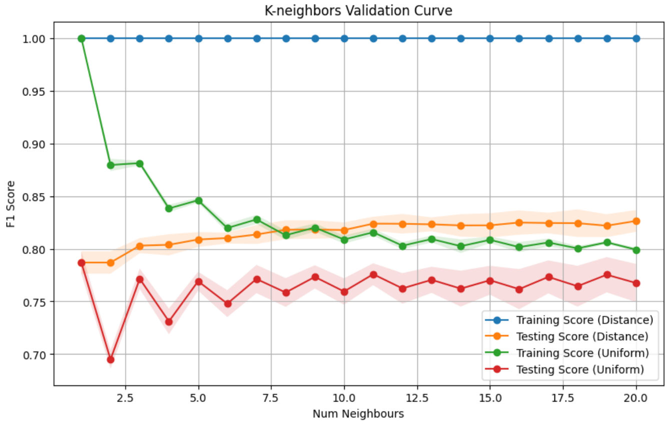

Fig. 26. Breast Cancer K-Nearest Neighbour Validation Curve.

D. Analysis and Comparison

The performance of k-Nearest Neighbours (KNN) on the Wine Quality and Breast Cancer datasets showcases its adaptability to diverse data characteristics. On the Wine Quality dataset, KNN demonstrates high variance, emphasizing the significance of selecting an appropriate number of neighbours. Increasing the number of neighbours enhances the generalization of KNN, preventing overfitting and handling outliers effectively. However, the use of distance weights in KNN reveals a peculiar phenomenon where the model memorizes the training data, resulting in a perfect score for training samples. This behavior doesn't necessarily translate to superior generalization on new, unseen data. Despite Grid Search suggesting distance weights as optimal, Table V, a practical choice might be uniform distance weights to prevent completely overfitting on training sets. On the Breast Cancer dataset, the small size on training samples poses challenges in achieving optimal generalization; Grid Search, Table V, might indicate distance weights as preferable, but the limited dataset size encourages adopting uniform distance weights to mitigate overfitting.

The wall clock time for KNN remains proportional to the dataset size, showcasing computational efficiency compared to more complex algorithms like neural networks and SVM. KNN's timing is competitive with decision trees and AdaBoosting, making it a viable option for datasets of moderate size. In comparing KNN with other algorithms, its simplicity and ease of use are highlighted. While KNN may not exhibit the same level of performance as Decision Tree or AdaBoosting, its adaptability and trade-off between model complexity and performance make it a practical choice for easy-moderately complex problems. The observed behavior of distance weights emphasizes the importance of understanding KNN's details for optimal use.

TABLE V
TABLE TYPE STYLES

Grid Search	**White Wine**	**Breast Cancer**
Best Weights	Distance	Distance
Best Num Neighbours	20	13
Best Accuracy	0.93487	0.9929
Wall Time	50.457	10.3803

Note that 'distance' weights perform overfitting on training set and even it is set as optimal for Grid Search we concluded that 'uniform' weights would allow us to generalize better to new unseen data.

IX. Conclusions

In this study, we conducted a comprehensive analysis and comparison of five machine learning algorithms—decision tree, AdaBoost, neural networks, SVM, and k-NN—across two distinct datasets: the White Wine Quality Dataset and the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. The aim was to unravel the detailed behavior of each algorithm, showing strengths, weaknesses, and optimal configurations.

The decision tree exhibited sensitivity to dataset characteristics, with its performance influenced by the interplay of sample size and feature dimensions. Notably, a smaller dataset benefited from shallower trees to avoid overfitting, while a larger feature space necessitated feature selection. Random splitter showed comparable results to best splitter, emphasizing the algorithm's robustness. Decision Tree was one of the algorithms that provided better results.

Neural networks exhibited a good fit for both datasets, albeit constrained by the set maximum iterations. Early overfitting in the learning curve suggested a need for more iterations to reach minimal error. Model complexity and adaptability played pivotal roles, and hidden layer layout was the most important point.

AdaBoost demonstrated a keen ability to adapt to the dataset's complexity, particularly thriving on larger training sets. The ensemble's performance benefited from the aggregation of weak learners, showcasing robustness to noise and outliers. Aggressive weight updating, model flexibility, and an emphasis on hard-to-classify instances distinguished AdaBoost's success, especially on medium-sized datasets like the ones we used. Along with Decision Tree, AdaBoost provided great results.

SVM's efficacy was evident, particularly with a linear kernel on both datasets. The choice of kernel, along with the regularization parameter C, proved vital. A higher C worked well for smaller datasets, showcasing SVM's adaptability and sensitivity to dataset characteristics. Computational complexity remained a challenge, especially as the dataset size increased, and compared to other algorithms.

k-NN demonstrated proficiency in handling varying dataset sizes. The distance-weighted variant raised concerns of overfitting due to perfect training scores, emphasizing the importance of cautious model selection. The algorithm's simplicity and dependency on neighbours made it computationally efficient, especially for smaller datasets.

In our general observations, several key insights emerged. The F1 score proved to be a robust evaluation metric, demonstrating its effectiveness in handling imbalances present in the datasets. Smaller datasets exhibited a preference for simpler models, while larger datasets highlighted the powers of ensemble methods and complex models in capturing intricate patterns. The computational complexity varied among the algorithms, resulting in diverse impacts on wall clock time and resource requirements.

Each algorithm revealed its unique characteristics and suitability for different scenarios. Decision trees excelled in interpretability, while ensemble methods like AdaBoost offered adaptability and robustness. Neural networks showcased flexibility and complexity, while SVM and k-NN addressed dataset details with kernel choices and neighbourhood dependencies, respectively.

In conclusion, the selection of an algorithm should align with the dataset's characteristics and the goals of the analysis. The findings from this study contribute valuable insights into the specific behaviors of popular machine learning algorithms, aiding practitioners in making informed choices for diverse applications. Future work may explore additional algorithms and datasets to further enrich our understanding of machine learning model dynamics.