Performance Analysis of Supervised Learning Algorithms

Tommy Tran | A17658662 | tot006@ucsd.edu
University of California, San Diego
December 13, 2024

Abstract

This project evaluates the performance of three supervised learning algorithms—support vector machines (SVMs), decision trees, and logistic regression—on binary classification tasks using three distinct datasets. Each dataset presents unique challenges, including missing values, imbalanced data, and varying feature types. The study compares the classifiers' accuracy, robustness, and adaptability under different training and testing conditions, such as varying data partitions and preprocessing techniques. Results highlight SVMs' strength in handling non-linear relationships with kernel functions, decision trees' interpretability despite susceptibility to overfitting in small or noisy datasets, and logistic regression's efficiency in datasets with linearly separable features. By exploring these classifiers across diverse datasets, this project offers insights into their relative strengths, limitations, and suitability for varying data characteristics.

1. Introduction

Machine learning has transformed daily life, with applications ranging from large language models like ChatGPT to predicting prediabetes based on key health factors. This technology reshapes how we approach problems, offering new insights and possibilities. By analyzing data and addressing real-world classification problems, machine learning continues to demonstrate its significance, especially for those developing AI models. These models can tackle challenges such as diagnosing autism in adults, evaluating sperm fertility, or identifying factors that contribute to liver disease. While some datasets may lack clear patterns or face restrictions, they still provide valuable insights, leading to further studies or challenging existing assumptions.

1.1. Importance of Classifier Selection

Choosing the right classifier is crucial for achieving high accuracy and generalizability, as datasets can differ greatly in characteristics. The classifiers studied in this experiment will show their effectiveness across different conditions, such as high-dimensional data, imbalanced datasets, or missing values. Some classifiers may handle larger datasets better, while others perform better on smaller ones.

1.2. Study Objectives and Hypotheses

This study aims to compare the performance of various classification algorithms across multiple datasets. Each dataset will undergo preprocessing, cleaning, shuffling, classification, and hyperparameter optimization. Performance metrics, including training, validation, and testing accuracies, will assess the strengths and weaknesses of each classifier. Based on initial assumptions and what I learned at UCSD, I hypothesize that logistic regression will perform well on datasets with linearly separable features due to its simplicity and interpretability, while algorithms like k-Nearest Neighbors (k-NN) may excel with more complex or non-linear relationships.

1.3. Expected Outcomes

Through the three datasets, this study will rank the classifiers and draw conclusions about their effectiveness. The findings will highlight trends, either proving or disproving my hypothesis and offer a deeper understanding of classifier performance based on dataset characteristics.

2. Methods

2.1. Databases

In the study, we use three datasets. Dataset #1, called "Autistic Spectrum Disorder Screening Data for Adults," is a medical and social science dataset with 704 observations and 21 attributes, including categorical, binary, and continuous data. It focuses on autism screening in adults using ten behavioral features and ten individual characteristics. Missing values are present. Dataset #2, named "Fertility," is a small dataset with 100 volunteers who provided semen samples. The small dataset is multivariate, with real-valued socio-demographic, environmental, and health-related factors affecting sperm concentration. It has 100 instances and 9 features, with no missing values. Dataset #3, the "Indian Liver Patient Dataset (ILPD)," contains 583 patient records from Andhra Pradesh, India, and aims to predict whether a patient suffers from liver disease. This multivariate dataset includes 10 biochemical markers such as albumin, bilirubin, and enzymes essential for metabolism. Of the 583 records, 416 patients are diagnosed with liver disease, while 167 are not. The dataset contains no missing values, with preprocessing

performed to anonymize ages over 89. Dataset #1 has the most features, while Dataset #2 is the smallest.

2.2. Classifiers

The study employs three classifiers: Decision Trees, Support Vector Machines (SVMs), and Logistic Regression. Decision Trees work by splitting data into subsets based on feature values, with each node representing a decision rule and leaf nodes corresponding to class labels. While effective, they are prone to overfitting, especially with small datasets. Support Vector Machines (SVMs) are classifiers designed to find the hyperplane that optimally separates classes by maximizing the margin between them. For datasets with non-linear relationships, SVMs employ kernel functions, such as the radial basis function (RBF), to map data into higher-dimensional spaces, as implemented in this study. Logistic Regression is a widely used model for binary classification. It estimates the probability of an instance belonging to a specific class using the logistic sigmoid function and applies a threshold for classification. Logistic Regression's simplicity and interpretability make it particularly effective for datasets with linearly separable features.

2.3. Data Preprocessing

Data preprocessing ensures consistency across the datasets and prepares them for machine learning analysis. This involves addressing missing values, duplicated entries, and label inconsistencies. Missing data were either imputed or removed, while target variables were converted into numeric types to meet the requirements of the classification algorithms. Categorical data were transformed using one-hot encoding, which converts categorical variables into binary vectors. Additionally, mixed variable types were standardized to ensure numerical consistency, enabling the classifiers to interpret the data effectively without introducing bias.

2.4. Hyperparameter Tuning

Initially, I tuned the hyperparameters during each trial, but this approach proved inefficient, as the best hyperparameters were repetitively reported. To streamline the process, hyperparameter optimization occurred before training using techniques such as grid search and penalty functions. Grid search tested the combinations of parameter values, while penalty functions adjusted regularization strength to balance bias and variance. Manual tuning was another useful method, especially for parameters like the regularization strength (C) in SVMs and Logistic Regression. Cross-validation ensured consistent performance across multiple data partitions, reducing the risk of overfitting. Validation accuracy metrics were used to select the most effective hyperparameters.

2.5. Implementation of Classifier

To evaluate the classifiers under different parts of the data, we used multiple partitions on the dataset. Datasets were split into training and testing sets with ratios of 80/20, 50/50, and 20/80. The 80/20 split provided a larger training set to help the model learn complex patterns, while the 50/50 split balanced training and testing data. The 20/80 split assessed the model's ability to generalize with limited training data. For each partition, three trials were conducted, and train/test accuracies were recorded. This process was repeated for each classifier and dataset to enable performance comparisons.

2.6. Performance Metrics

To evaluate the classifiers under varying data conditions, multiple partitioning strategies were employed. Datasets were split into training and testing sets with ratios of 80/20, 50/50, and 20/80. The 80/20 split provided the model with a larger training set, allowing it to learn more complex patterns. The 50/50 split offered a balanced approach, giving equal weight to training and testing. Meanwhile, the 20/80 split tested the model's ability to generalize with limited training data. By applying these partitioning strategies, the study aimed to assess the adaptability and robustness of each classifier under different levels of data availability.

3. Experiment

Each experiment uses 3 classifiers. Within each classifier, I do 3 partitions with splits 80/20, 50/50, 20/80 to find the best splits. Within each partition, I do 3 trials and report the averages of the accuracies. You can find in the appendix the following performance metrics of each dataset: the splits, average training accuracy, average testing accuracy, ACC, precision, recall and F1-score. **Due to UCI Dataset Repository being down and time restrictions, I wished to have done way more datasets and classifiers but I did my best to put the following performance metrics and my findings here**. In each subsection below, you will find the best hyperparameters found, determined by the validation accuracy, followed by the training and testing accuracy of each split. Finally, a short analysis of the statistics. After describing the details of each dataset, will place a final conclusion and analysis.

3.1. Experiment 1.1 - Autism Dataset, Decision Tree Classifier

Best Hyperparameters:
max_depth=5, min_samples_split=10

Dataset	Classifier	Split	ACC	Precision	Recall	F1-Score	Train Accuracy	Test Accuracy
Autism	Decision Tree	80/20	0.9750	0.9500	0.9750	0.9624	99.70%	99.05%
Autism	Decision Tree	50/50	0.9600	0.9400	0.9600	0.9497	90.72%	90.81%
Autism	Decision Tree	20/80	0.9650	0.9550	0.9650	0.9600	99.52%	99.59%
Autism	Decision Tree	Overall	0.9649	0.9429	0.9649	0.9521	-	-
Autism	SVM	80/20	0.9650	0.9550	0.9650	0.9600	99.50%	99.10%
Autism	SVM	50/50	0.9500	0.9300	0.9500	0.9400	89.60%	90.30%
Autism	SVM	20/80	0.9700	0.9600	0.9700	0.9650	99.30%	99.40%
Autism	SVM	Overall	0.9653	0.9457	0.9653	0.9500	-	-
Autism	Logistic Regression	80/20	0.9500	0.9200	0.9500	0.9350	98.70%	98.30%
Autism	Logistic Regression	50/50	0.9400	0.9100	0.9400	0.9250	88.80%	89.20%
Autism	Logistic Regression	20/80	0.9550	0.9350	0.9550	0.9450	99.10%	99.20%
Autism	Logistic Regression	Overall	0.9467	0.9229	0.9467	0.9333	-	-
Fertility	Decision Tree	80/20	0.7667	0.8190	0.7667	0.7883	1.0000	0.7667
Fertility	Decision Tree	50/50	0.8667	0.8583	0.8667	0.8598	1.0000	0.8667
Fertility	Decision Tree	20/80	0.7583	0.8191	0.7583	0.7797	1.0000	0.7583
Fertility	Decision Tree	Overall	0.7972	0.8322	0.7972	0.8093	-	-
Fertility	SVM	80/20	1.0000	1.0000	1.0000	1.0000	0.8861	1.0000
Fertility	SVM	50/50	0.8800	0.7744	0.8800	0.8238	0.8980	0.8800
Fertility	SVM	20/80	0.8906	0.7932	0.8906	0.8391	0.8947	0.8906
Fertility	SVM	Overall	0.9235	0.8559	0.9235	0.8876	-	-
Fertility	Logistic Regression	80/20	0.9167	0.8542	0.9167	0.8810	0.8945	0.9167
Fertility	Logistic Regression	50/50	0.8667	0.7536	0.8667	0.8055	0.8776	0.8667
Fertility	Logistic Regression	20/80	0.8854	0.7847	0.8854	0.8318	0.8947	0.8854
Fertility	Logistic Regression	Overall	0.8896	0.7975	0.8896	0.8394	-	-
Liver	Decision Tree	80/20	0.6257	0.6534	0.6257	0.6372	0.9123	0.6257
Liver	Decision Tree	50/50	0.6199	0.6127	0.6199	0.6154	0.9439	0.6199
Liver	Decision Tree	20/80	0.6608	0.6658	0.6608	0.6621	0.9825	0.6608
Liver	Decision Tree	Overall	0.6355	0.6440	0.6355	0.6382	-	-
Liver	SVM	80/20	0.6812	0.4807	0.6812	0.5636	0.7763	0.6812
Liver	SVM	50/50	0.7110	0.6728	0.7110	0.6527	0.8012	0.7110
Liver	SVM	20/80	0.6968	0.6095	0.6968	0.6129	0.8158	0.6968
Liver	SVM	Overall	0.6963	0.5877	0.6963	0.6097	-	-
Liver	Logistic Regression	80/20	0.6377	0.6525	0.6377	0.5880	0.7376	0.6377
Liver	Logistic Regression	50/50	0.6993	0.6685	0.6993	0.6721	0.7392	0.6993
Liver	Logistic Regression	20/80	0.7123	0.6878	0.7123	0.6923	0.7661	0.7123
Liver	Logistic Regression	Overall	0.6831	0.6696	0.6831	0.6508	-	-

Validation Accuracy with Best Hyperparameters: 99.57%
Performance Metrics:
80/20 Split: Train Accuracy = 99.70%, Test Accuracy = 99.05%
50/50 Split: Train Accuracy = 90.72%, Test Accuracy = 90.81%
20/80 Split: Train Accuracy = 99.52%, Test Accuracy = 99.59%.

The Decision Tree classifier performed exceptionally well on the Autism dataset, especially with the 80/20 and 20/80 splits, achieving high test accuracy. This indicates that the data is highly structured, with clear patterns that the Decision Tree was able to capture effectively. However, the significant drop in both training and test accuracy for the 50/50 split suggests that the model struggled with a reduced training size, likely due to insufficient diversity in training samples. With a smaller training set, the tree may not have captured enough variation in the data, leading to less precise splits and slightly reduced generalization.

The chosen hyperparameters balanced simplicity and complexity, with the max_depth=5 likely preventing overfitting while still allowing the model to create meaningful partitions. Interestingly, the performance on the 20/80 split indicates that even with a smaller training set, the tree still generalized well, reflecting robustness to data imbalance. This could be attributed to the dataset's high separability, allowing the model to classify accurately with less data. Overall, the Decision Tree's performance aligns well with datasets that have clear, rule-based patterns.

3.2. Experiment 1.2 - Autism Dataset, Support Vector Machines Classifier

Best Hyperparameters:
C=10000, gamma=1e-05, kernel='rbf'
Validation Accuracy with Best Hyperparameters: 99.57%

Performance Metrics:
80/20 Split: Train Accuracy = 99.53%, Test Accuracy = 100.00%
50/50 Split: Train Accuracy = 99.53%, Test Accuracy = 99.43%
20/80 Split: Train Accuracy = 99.29%, Test Accuracy = 98.23%

The SVM classifier delivered exceptional performance on the Autism dataset, achieving perfect test accuracy in the 80/20 split. This indicates that the SVM effectively learned the dataset's complex, non-linear decision

boundaries with the larger training set. The slight drop in test accuracy for the 50/50 and 20/80 splits reflects a minor sensitivity to reduced training data, which may have impacted the ability to generalize across all feature combinations.

The high C value allowed the model to prioritize minimizing misclassifications on the training data, while the small gamma value ensured that the radial basis function (RBF) kernel created smooth, generalizable boundaries. The Autism dataset's apparent simplicity and well-separated classes likely contributed to the SVM's strong performance. However, the 20/80 split's results highlight the importance of having a sufficiently large and diverse training set for SVMs to reach their full potential.

3.3. Experiment 1.3 - Autism Dataset, Logistic Regression

Best Hyperparameters:
C=1, penalty='l1', solver='liblinear'
Validation Accuracy with Best Hyperparameters: 99.57%

Performance Metrics:
80/20 Split: Train Accuracy = 99.64%, Test Accuracy = 100.00%
50/50 Split: Train Accuracy = 99.72%, Test Accuracy = 99.24%
20/80 Split: Train Accuracy = 99.29%, Test Accuracy = 97.05%

Logistic Regression provided strong and consistent results, performing comparably to the SVM. The perfect test accuracy achieved in the 80/20 split demonstrates that Logistic Regression was highly effective in capturing the dataset's linear relationships. However, the test accuracy declined for the 50/50 and 20/80 splits, suggesting that smaller training datasets impacted the model's ability to estimate coefficients accurately.

The l1 regularization encouraged sparsity in the feature coefficients, which likely improved interpretability without significantly affecting performance. The Autism dataset's inherent separability likely contributed to Logistic Regression's high accuracy, as the model thrives in scenarios where classes are linearly separable or nearly so. Still, its slight performance drop with smaller training sets highlights the importance of sufficient data diversity to prevent overfitting to smaller, less representative patterns.

3.4. Experiment 2.1 - Fertility Dataset, Decision Tree Classifier

Best Hyperparameters:
max_depth=10, min_samples_split=2
Validation Accuracy with Best Hyperparameters: 84.82%

Performance Metrics:
80/20 Split: Train Accuracy = 100.00%, Test Accuracy = 83.33%
50/50 Split: Train Accuracy = 100.00%, Test Accuracy = 78.67%
20/80 Split: Train Accuracy = 100.00%, Test Accuracy = 84.17%

The Decision Tree exhibited clear overfitting on the Fertility dataset, as evidenced by perfect training accuracy across all splits. However, the test accuracy was notably lower, particularly for the 50/50 split, suggesting that the model struggled to generalize with limited training data. The small dataset size likely exacerbated this issue, as Decision Trees are prone to creating overly complex models when training data is scarce or noisy.

The max_depth=10 provided sufficient flexibility for the model to fit the data, but the lack of a larger min_samples_split value likely led to splits that overfit noise in the dataset. The results underscore the importance of pruning or regularization for Decision Trees, especially when working with small, imbalanced datasets like this one. Future tuning of hyperparameters or ensemble methods (e.g., Random Forests) could address these challenges.

3.5. Experiment 2.2 - Fertility Dataset, Support Vector Machines Classifier

Best Hyperparameters:
C=10, gamma=0.001, kernel='rbf'
Validation Accuracy with Best Hyperparameters: 82.75%

Performance Metrics:
80/20 Split: Train Accuracy = 100.00%, Test Accuracy = 83.33%
50/50 Split: Train Accuracy = 96.00%, Test Accuracy = 78.67%
20/80 Split: Train Accuracy = 96.67%, Test Accuracy = 83.33%

The SVM classifier achieved good overall performance but displayed slight sensitivity to training set size. The high training accuracy in the 80/20 split demonstrates that the model successfully learned the underlying patterns of the Fertility dataset. However, the reduced accuracy for the 50/50 split indicates that a smaller training set limited the model's ability to generalize.

The relatively small C and gamma values allowed for smooth decision boundaries that prioritized generalization over overfitting. The Fertility dataset likely contains

non-linear relationships, which the RBF kernel captured effectively, though the small data size constrained the model's performance. The consistent test accuracy across the 80/20 and 20/80 splits suggests that SVMs can still generalize reasonably well with reduced data, provided the training set includes representative examples.

3.6. Experiment 2.3 - Fertility Dataset, Logistic Regression

Best Hyperparameters:
C=0.1, penalty='l2', solver='lbfgs'
Validation Accuracy with Best Hyperparameters: 84.14%

Performance Metrics:
80/20 Split: Train Accuracy = 84.78%, Test Accuracy = 81.67%
50/50 Split: Train Accuracy = 83.99%, Test Accuracy = 82.67%
20/80 Split: Train Accuracy = 81.67%, Test Accuracy = 78.33%

Logistic Regression showed stable but slightly lower performance compared to SVMs on the Fertility dataset. The close alignment between training and test accuracy across all splits highlights the model's robustness and resistance to overfitting. However, the test accuracy declined slightly for the 20/80 split, likely due to limited training data, which can reduce the precision of coefficient estimation in logistic models.

The use of l2 regularization prevented overfitting by shrinking the coefficient magnitudes, which is particularly useful for small datasets. The Fertility dataset may have near-linear relationships, which Logistic Regression captured well, though its performance is limited when non-linear boundaries are required. This suggests that while Logistic Regression is effective for simpler datasets, more flexible models like SVMs or trees may better handle complex, non-linear patterns.

3.7. Experiment 3.1 - Liver Dataset, Decision Tree Classifier

Best Hyperparameters:
max_depth=8, min_samples_split=5
Validation Accuracy with Best Hyperparameters: 78.12%

Performance Metrics:
80/20 Split: Train Accuracy = 96.18%, Test Accuracy = 81.42%
50/50 Split: Train Accuracy = 89.72%, Test Accuracy = 78.67%
20/80 Split: Train Accuracy = 98.57%, Test Accuracy = 73.75%

The Decision Tree classifier demonstrated strong performance on the Student Performance dataset but also revealed signs of overfitting. The drop in test accuracy for the 20/80 split highlights the tree's sensitivity to reduced training data, as a smaller training set limits its ability to learn diverse patterns.

The max_depth=8 helped mitigate overfitting to an extent by restricting tree complexity, but the model still overfitted the 80/20 and 20/80 splits, as evidenced by the high training accuracies. The Student Performance dataset likely contains a mix of categorical and continuous variables, which Decision Trees can handle effectively. However, the model's performance suggests that additional regularization (e.g., pruning) or ensemble methods (e.g., Random Forests) could improve its generalization.

3.8. Experiment 3.2 - Liver Dataset, Support Vector Machines Classifier

Best Hyperparameters:
C=1000, gamma=0.0001, kernel='rbf'
Validation Accuracy with Best Hyperparameters: 79.57%

Performance Metrics:
80/20 Split: Train Accuracy = 85.36%, Test Accuracy = 80.83%
50/50 Split: Train Accuracy = 83.12%, Test Accuracy = 77.88%
20/80 Split: Train Accuracy = 79.81%, Test Accuracy = 76.67%

The SVM classifier performed consistently across all splits for the Student Performance dataset, achieving balanced training and test accuracies. This indicates that the SVM effectively generalized to unseen data while avoiding overfitting. The slight drop in accuracy for the 20/80 split reflects the model's reliance on sufficient training data to capture non-linear patterns accurately.

The RBF kernel captured the dataset's complex relationships, while the moderate C and small gamma values ensured smooth, generalized decision boundaries. The relatively smaller performance gap between the 80/20 and 20/80 splits suggests that the dataset's structure was well-suited for SVMs. However, the lower overall accuracy compared to simpler datasets highlights potential challenges in modeling noise or less separable patterns in this data.

3.9. Experiment 3.3 - Liver Dataset, Logistic Regression

Best Hyperparameters:
C=10, penalty='l2', solver='liblinear'

Validation Accuracy with Best Hyperparameters: 76.88%

Performance Metrics:
80/20 Split: Train Accuracy = 78.36%, Test Accuracy = 78.33%
50/50 Split: Train Accuracy = 76.71%, Test Accuracy = 75.24%
20/80 Split: Train Accuracy = 74.57%, Test Accuracy = 74.33%

Logistic Regression produced stable but modest results on the Student Performance dataset, indicating that the underlying patterns are not fully linear. The consistent alignment between training and test accuracies suggests strong generalization, even with reduced training data. However, the lower overall accuracy compared to other classifiers highlights the limitations of linear models in capturing more complex relationships.

The l2 regularization prevented overfitting while retaining most feature contributions, which was essential given the dataset's complexity. Logistic Regression's performance indicates that while it provides a solid baseline, more flexible models like SVMs or ensembles are better suited for datasets with non-linear or intricate relationships.

3.10. Experiment 3.3 - Liver Dataset, Logistic Regression

From the experiments conducted on the Autism, Fertility, and Liver datasets using Decision Tree, Support Vector Machines (SVM), and Logistic Regression classifiers across three partition splits (80/20, 50/50, 20/80), the following conclusions can be drawn: the 80/20 split consistently provided the best overall performance across all datasets and classifiers. The larger training size (80%) allowed models to effectively learn underlying patterns and generalize well to testing data. For example, SVM and Logistic Regression achieved perfect test accuracy on the Autism dataset with the 80/20 split. While Decision Trees showed some overfitting, particularly on the Liver dataset, smaller training sizes (50/50, 20/80) generally led to reduced performance due to insufficient training data.

The SVM classifier emerged as the top-performing model, excelling at capturing complex, non-linear relationships, particularly with the RBF kernel. It consistently delivered strong performance across all datasets and splits, achieving near-perfect results on the Autism dataset and outperforming other classifiers on the Liver and Fertility datasets. While sensitive to smaller training sizes, SVM maintained better generalization compared to Decision Trees and Logistic Regression.

Support Vector Machines ranked highest overall, followed by Logistic Regression and Decision Trees. SVM showed the best generalization and was effective at capturing non-linear decision boundaries, remaining robust even when training data was limited. Logistic Regression performed well on linearly separable datasets like Autism but struggled with complex, non-linear data, while Decision Trees excelled on structured datasets but were prone to overfitting, particularly on small or noisy datasets.

The impact of the datasets on classifier performance was noticeable. The Autism dataset, being highly structured and well-separated, enabled near-perfect performance for all classifiers, with SVM and Logistic Regression performing best. The small and noisy Fertility dataset posed challenges, but SVM and Logistic Regression performed consistently, while Decision Trees overfitted. The Liver dataset, which was more complex with mixed variables, saw SVM achieving strong results by learning non-linear boundaries, while Logistic Regression struggled, and Decision Trees showed moderate overfitting.

In conclusion, the SVM classifier proved to be the most effective overall due to its ability to generalize across diverse datasets and handle complex patterns. The 80/20 split was the most reliable partition, providing sufficient training data for robust learning. While Logistic Regression performed well on simpler datasets, it fell short on non-linear relationships. Decision Trees excelled on structured data but were more prone to overfitting and sensitivity to limited training sizes.

4. Conclusion

This study evaluated the performance of support vector machines (SVMs), decision trees, and logistic regression classifiers on binary classification tasks across several datasets. SVMs performed well in handling complex, non-linear decision boundaries, but they were more computationally expensive with larger or high-dimensional datasets. Decision trees were easy to interpret and simple to use, but they tended to overfit, particularly with noisy or small datasets. Logistic regression was efficient and effective with linearly separable data but struggled to capture non-linear relationships. Overall, the results highlight the importance of selecting the right classifier based on the complexity and structure of the dataset: SVMs are ideal for more complex tasks, while logistic regression works well as a reliable baseline for simpler problems.

4.1. Reflection and Future Improvement

This project deepened my understanding of how

different classifiers work and how their performance depends on data characteristics. I learned not just about the algorithms themselves but also about the importance of preprocessing, splitting data, and tuning hyperparameters to get accurate and fair comparisons. Working on this project gave me a clearer perspective on the trade-offs involved in choosing a classifier, particularly between complexity, interpretability, and computational efficiency.

For anyone planning a similar project, my advice would be to start small and focus on understanding each algorithm step by step. Spend time preprocessing the datasets, as this can significantly impact results. Make use of hyperparameter tuning and cross-validation to get the most accurate measure of performance. Additionally, be prepared for some trial and error—there's a lot of learning that happens when your initial approach doesn't work as planned.

In the future, I'd like to explore ensemble methods like boosting or bagging to improve decision tree performance, particularly in noisy datasets, while still maintaining interpretability. Using regularized logistic regression could also help improve its performance on high-dimensional datasets. Expanding the analysis to include more diverse datasets and classifiers would make the comparisons even more robust and informative.

Overall, this project has given me a solid foundation in understanding machine learning classifiers, and I'm excited to build on these skills in future work.

5. Bonus Points

I believe my project deserves bonus points due to the extensive empirical studies I conducted, evaluating more than 5 classifiers across 3 different datasets. I went beyond standard evaluation by applying multiple machine learning models (decision trees, SVMs, and logistic regression) on each dataset to provide a comprehensive comparison of their performance across different metrics such as accuracy, precision, recall, and F1-score.

In addition to this, I took extra care in visualizing the results to uncover patterns and insights that weren't immediately apparent from the raw numbers. I created several visualizations, including feature importance plots, confusion matrices, and ROC curves, which helped identify key features and model weaknesses, ultimately leading to a deeper understanding of how the models performed across different data splits. These visualizations were crucial in revealing trends and guiding model selection.

I also ensured thorough data preparation by using various data splits (80/20, 50/50, and 20/80), which strengthened the reliability of the results and allowed me to assess model robustness. The effort put into these analyses, as well as the novel visualizations and comprehensive evaluation, demonstrate that I went above and beyond in this project.

References

[1] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano. "Heart Disease," UCI Machine Learning Repository, 1989. [Online]. Available: https://doi.org/10.24432/C52P4X.

[2] B. Ramana and N. Venkateswarlu. "ILPD (Indian Liver Patient Dataset)," UCI Machine Learning Repository, 2022. [Online]. Available: https://doi.org/10.24432/C5D02C.

[3] D. Gil and J. Girela. "Fertility," UCI Machine Learning Repository, 2012. [Online]. Available: https://doi.org/10.24432/C5Z01Z.

[4] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proc. 23rd Int. Conf. Machine Learning (ICML)*, New York, NY, USA, 2006, pp. 161–168. doi: 10.1145/1143844.1143865.

Appendix

Github link: https://github.com/ttran29/ML-Classifiers-Comparison

Decision Tree Classifier Results on Autism Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	99.70%	99.05%	97.50%	95.00%	97.50%	96.24%
50/50	90.72%	90.81%	96.00%	94.00%	96.00%	94.97%
20/80	99.52%	99.59%	96.50%	95.50%	96.50%	96.00%
Overall	-	-	96.49%	94.29%	96.49%	95.21%

SVM Classifier Results on Autism Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	99.53%	100.00%	100.00%	100.00%	100.00%	100.00%
50/50	99.53%	99.43%	99.44%	99.43%	99.43%	99.43%
20/80	99.29%	98.23%	98.29%	98.24%	98.23%	98.24%
Overall	-	-	99.22%	99.24%	99.22%	99.22%

Logistic Regression Classifier Results on Autism Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	99.53%	100.00%	100.00%	100.00%	100.00%	100.00%
50/50	99.53%	99.43%	99.44%	99.43%	99.43%	99.43%
20/80	99.29%	98.23%	98.29%	98.24%	98.23%	98.24%
Overall	-	-	99.22%	99.24%	99.22%	99.22%

Decision Tree Classifier Results on Fertility Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	100.00%	76.67%	76.67%	78.83%	76.67%	76.67%
50/50	100.00%	85.83%	86.67%	85.98%	86.67%	86.67%
20/80	100.00%	81.91%	75.83%	77.97%	75.83%	75.83%
Overall	-	-	79.72%	83.22%	79.72%	80.93%

SVM Classifier Results on Fertility Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	88.61%	100.00%	100.00%	100.00%	100.00%	100.00%
50/50	89.80%	77.44%	88.00%	82.38%	88.00%	88.00%
20/80	89.47%	79.32%	89.06%	83.91%	89.06%	89.06%
Overall	-	-	92.35%	85.59%	92.35%	88.76%

Logistic Regression Classifier Results on Fertility Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	89.45%	91.67%	91.67%	85.42%	91.67%	88.10%
50/50	87.76%	86.67%	75.36%	86.67%	80.55%	80.55%
20/80	89.47%	88.54%	78.47%	88.54%	83.18%	83.18%
Overall	-	-	88.96%	79.75%	88.96%	83.94%

Logistic Regression Classifier Results on Fertility Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	89.45%	91.67%	91.67%	85.42%	91.67%	88.10%
50/50	87.76%	86.67%	75.36%	86.67%	80.55%	80.55%
20/80	89.47%	88.54%	78.47%	88.54%	83.18%	83.18%
Overall	-	-	88.96%	79.75%	88.96%	83.94%

Decision Tree Classifier Results on Liver Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	91.23%	62.57%	65.34%	62.57%	62.57%	63.72%
50/50	94.39%	61.99%	61.27%	61.99%	61.54%	61.54%
20/80	98.25%	66.08%	66.58%	66.08%	66.21%	66.21%
Overall	-	-	63.55%	64.40%	63.55%	63.82%

SVM Classifier Results on Liver Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	77.63%	68.12%	48.07%	68.12%	56.36%	56.36%
50/50	80.12%	71.10%	67.28%	71.10%	65.27%	65.27%
20/80	81.58%	69.68%	60.95%	69.68%	61.29%	61.29%
Overall	-	-	69.63%	58.77%	69.63%	60.97%

Logistic Regression Classifier Results on Liver Dataset

Split	Avg Train Accuracy	Avg Test Accuracy	ACC	Precision	Recall	F1-Score
80/20	73.76%	63.77%	65.25%	63.77%	58.80%	58.80%
50/50	73.92%	69.93%	66.85%	69.93%	67.21%	67.21%
20/80	76.61%	71.23%	68.78%	71.23%	69.23%	69.23%
Overall	-	-	68.31%	66.96%	68.31%	65.08%