**Evaluation Report: Comparison of Decision Tree and Bagging Classifiers**

Introduction

The aim of this report is to compare the predictive performance of different implementations of decision tree classifiers and bagging classifiers using **repeated** **K-fold cross-validation** on five classification datasets.   
The evaluation metrics used were **accuracy, precision, recall, F1-score, and ROC-AUC score**. All evaluation metrics presented in the results relate to a test-sets in the cross-validation.  
The implementations compared included the **DecisionTreeClassifier** and **BaggingClassifier** from the sklearn package, as well as **MyID3** and **MyBaggingID3** implemented in python.

Datasets

The five datasets selected for this study were the **Divorce Predictors Dataset, Titanic Dataset, Heart Disease Cleveland Dataset, Breast Cancer Wisconsin Dataset, and Qualitative Bankruptcy Dataset.** These datasets were chosen due to their varying characteristics and complexity.

Results -

| **Dataset** | **Method** | **Accuracy** | **F1-score** | **Precision** | **Recall** | **ROC AUC** | **fit time (ms)** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Breast Cancer Wisconsin | MyID3 | 0.752 | 0.592 | 0.790 | 0.494 | 0.699 | 1712 |
| Breast Cancer Wisconsin | DecisionTreeClassifier | 0.740 | 0.560 | 0.780 | 0.451 | 0.683 | 60 |
| Breast Cancer Wisconsin | MyBaggingID3 | 0.373 | 0.542 | 0.373 | 1.000 | 0.500 | 237168 |
| Breast Cancer Wisconsin | BaggingClassifier | 0.750 | 0.589 | 0.777 | 0.491 | 0.700 | 2347 |
| Divorce Predictors | MyID3 | 0.969 | 0.968 | 0.976 | 0.961 | 0.969 | 277 |
| Divorce Predictors | DecisionTreeClassifier | 0.971 | 0.968 | 0.971 | 0.966 | 0.970 | 33 |
| Divorce Predictors | MyBaggingID3 | 0.975 | 0.973 | 0.996 | 0.952 | 0.974 | 19085 |
| Divorce Predictors | BaggingClassifier | 0.975 | 0.971 | 1.000 | 0.945 | 0.973 | 1161 |
| Heart Disease Cleveland | MyID3 | 0.818 | 0.795 | 0.833 | 0.768 | 0.816 | 1216 |
| Heart Disease Cleveland | DecisionTreeClassifier | 0.830 | 0.805 | 0.845 | 0.775 | 0.826 | 73 |
| Heart Disease Cleveland | MyBaggingID3 | 0.470 | 0.575 | 0.467 | 0.839 | 0.516 | 152932 |
| Heart Disease Cleveland | BaggingClassifier | 0.815 | 0.793 | 0.824 | 0.771 | 0.813 | 2331 |
| Qualitative Bankruptcy | MyID3 | 0.988 | 0.985 | 0.984 | 0.988 | 0.987 | 152 |
| Qualitative Bankruptcy | DecisionTreeClassifier | 0.995 | 0.993 | 0.986 | 1.000 | 0.996 | 214 |
| Qualitative Bankruptcy | MyBaggingID3 | 0.939 | 0.930 | 0.917 | 0.954 | 0.944 | 16563 |
| Qualitative Bankruptcy | BaggingClassifier | 0.988 | 0.985 | 0.991 | 0.980 | 0.987 | 3601 |
| Titanic Survival Predictors | MyID3 | 0.770 | 0.670 | 0.803 | 0.589 | 0.741 | 1766 |
| Titanic Survival Predictors | DecisionTreeClassifier | 0.765 | 0.673 | 0.783 | 0.601 | 0.740 | 97 |
| Titanic Survival Predictors | MyBaggingID3 | 0.565 | 0.157 | 0.417 | 0.139 | 0.487 | 307920 |
| Titanic Survival Predictors | BaggingClassifier | 0.779 | 0.693 | 0.778 | 0.629 | 0.755 | 2873 |

Weights & Biases link -   
https://wandb.ai/machinelearning\_37225214/MachineLearning\_Assignment1

Discussion

Overall, we found that the Bagging Classifier consistently performed well on most of the datasets, achieving high accuracy, F1-score, precision, and ROC\_AUC scores, and low fit time. This suggests that using an ensemble of multiple weak classifiers can improve the overall performance of a model.

Interestingly, we observed that the performance of some models was lower on certain datasets. For example, the MyBaggingID3 model had low accuracy and F1-score on the Breast Cancer Wisconsin, the Heart Disease Cleveland and the Titanic Survival Predictors datasets compared with the BaggingClassifier model despite their base classifiers (MyID3 and DecisionTreeClassifier respectively), achieving comparable metrics.   
This can be attributed to the default split criterion of the sklearn model being ‘gini’ whereas MyBaggingID3 uses entropy as the splitting method. This change leads to differences in performance, specifically in the case of imbalanced datasets where entropy can help improve recall at the cost of accuracy and precision.

Another potential factor that may have affected the performance of the models is the preprocessing of the data. The three datasets mentioned above have multiple continuous features which have been binned and oneHot encoded in preprocessing (creating multiple low-value features) when switching to binary features may explain the poor performance observed in some cases.

Regarding fit time, it is apparent that, for most datasets, the DecisionTreeClassifier is more efficient than MyID3 by an order of magnitude or even two. This may be due to the implementation language of the models with MyID3 being implemented in Python and the sklearn model being implemented in C. This carries over to the ensemble models resulting in the BaggingClassifier being much more efficient than the MyBaggingID3 model which, including the hyperparameter tuning and the cross-validation, can take up to a couple of minutes to fit.

In summary, the BaggingClassifier was the best-performing model in most metrics, however for the simpler datasets with straightforward categorical features the base decisionTrees perform just as well and more efficiently. The ensemble models struggle with datasets with imbalanced classes and with continuous features which were aggressively discretized and oneHot encoded to match the binary feature requirement, leading to loss of data and numerous low-information features.