ICS 435/635

Machine Learning

Spring 2025

Assignment Report

Section 1: Task Description

In this project, I created three types of models to classify the breast cancer data. Results from each model showed advantages and disadvantages for each model. For example, a decision tree model returned poorer accuracy scores than a random forest model, however, it is more comprehensive and easier to explain to someone in a practical situation. On the other hand, a random forest model recorded pretty high scores even though the hyperparameters are not changed. Even though it explained the data well, due to the ensemble learning model, it is more difficult to explain in the real world. Considering the processes and codes below, I go over the simplicity of each model and performance of them.

Section 2: Model Description

KNN model: I firstly started working on the K-nearest neighbors model with n_neighbors of 5. Gradually, I adjusted the parameter to analyze its impact on classification performance.

Decision Tree model: Used with default parameters. Gradually, adjusted max_depth, which made the model more complex and higher performance, however, it also increased the risk of overfitting.

Random forest model: An ensemble learning model, returned stable results regardless of parameters tuning. Started with 100 trees and adjusted max_depth as parameters.

Section 3: Experiment Settings

3.1 Dataset Description

The breast cancer dataset consists of 8 classes: data, target, frame, target_names, DESCR, feature_names, filename, and data_module. I only use the data as explaining features(called X), and target as explained value(called y). The dataset contains 30 columns with 569 entries. I split the data X and y into X_train, X_test, y_train, and y_test with a test ratio of 20% and shuffle the dataset. For preservation of the result, I used random_seed = 42. Only for the KNN model, I used a standard scaler to adjust the scale of train data and test data(both are called scaled_X_train and scaled_X_test).

3.2 Detailed Experimental Setups

The first KNN model is set up with n_neighbors = 5. In the updating parameter step, pre-defined parameter values are used. The list of n_neighbors [1, 3, 5, 7, 9, 11, 13, 15, 17, 19] are used to find the optimal KNN model for prediction.

The decision tree model used defaul;t settings initially, but updating max_depth parameter in = [1, 3, 5, 7, 9, 11, 13, 15].

The random forest model starts with n_estimartors = 100 and changes the max_depth eventually. However, at the optimization step, I only used max_depth to compare the impact of its changes with the decision tree since a random forest model is made of some decision trees.

3.3 Evaluation Metrics

Accuracy score measures how many samples are classified correctly. This matrix is used when a developer takes the number successful classification is more important than mis classifications. However, If you only use the accuracy matrix, it does not tell if the model works well or not.

Precision score measures true positive rate among predicted positives.

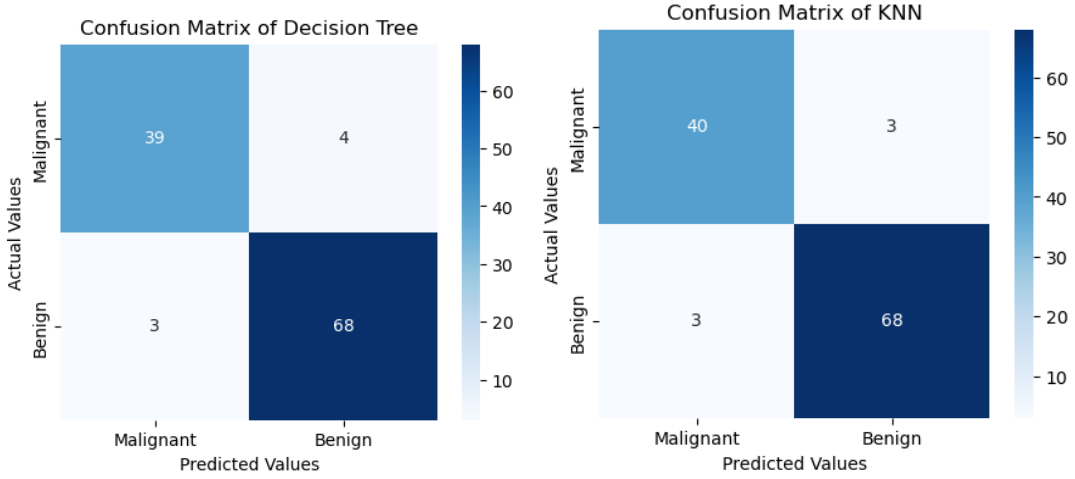
Recall measures how well the model identifies actual positives.

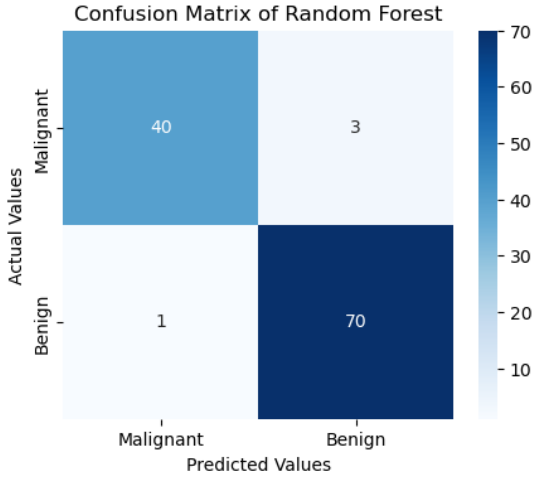
F1 score is a harmonic mean of precision and recall, balancing false positives and false negatives. Due to its features, it depends on the precision score and recall score overall.

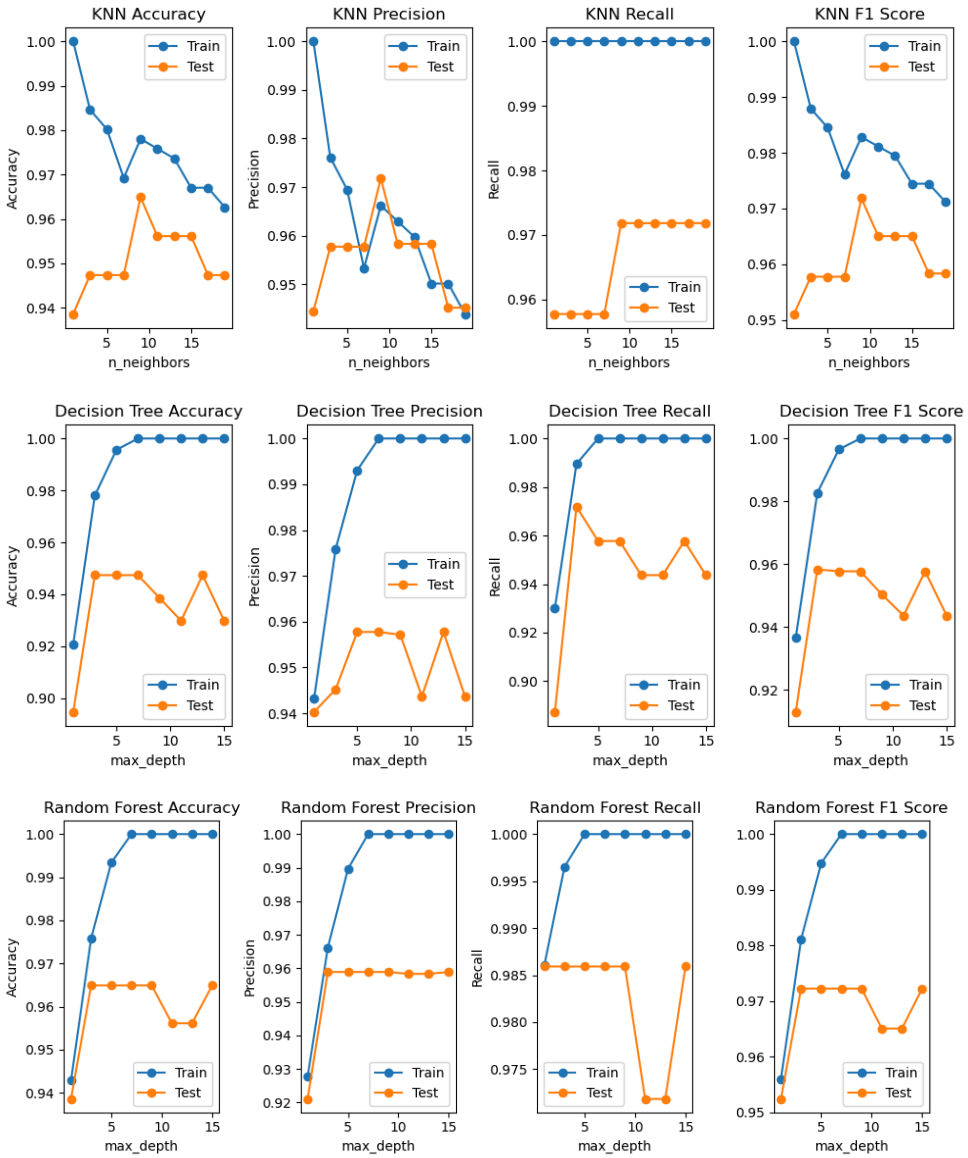
3.4 Source Code

https://github.com/000Aym612/ICS435_hws/blob/main/hw1/hw1.ipynb

3.5 Model Performance







3.6 Ablation Studies

As for the KNN model, increasing n_neighbors performs better until a certain point. After the point, roughly around 8~10, the model performance starts decreasing with test data, but keeps increasing in training data. That indicates the model starts overfitting.

Meanwhile, the decision tree has really unstable prediction results. Generally, as the number of max_depth increases, the model performance increases as well, however, the model works sometimes good but sometimes bad regardless of the max_depth value. Thus, I conclude the decision tree is simple and interpretable but really unstable.

Compared with the first two models, the random forest model performed stable. Even though the scores against training data tended to be overfitting, that did not affect test data evaluation. Considering the gap between training data and test data, the sweet spot would be the minimum point that satisfies the best model performance.

Section 4: Conclusion

In this study, three machine learning models: KNN, Decision Tree, and Random Forest, were evaluated for classifying breast cancer data. Each model showed unique strengths and weaknesses. The KNN model demonstrated improved performance with increasing neighbors but high risk of overfitting beyond a certain threshold. The Decision Tree model was interpretable, however, produced highly unstable results, making it less reliable for consistent classification tasks. In contrast, the Random Forest model, which is the only ensemble learning model among three, delivered stable and high quality of results, even without extensive hyperparameter tuning. However, its complexity makes it harder to interpret in real-world applications. Overall, model selection should be balanced performance, interpretability, and stability based on the specific tasks and purposes.