DATA 622 Assignment

Bank Marketing

Susanna Wong
Professor Joseph Sabeljia
March 24, 2025

Introduction

The Bank Marketing dataset was obtained from UCI Machine Learning Repository. It contains 45,211 client data related to direct marketing campaign conducted by a Portuguese banking institution, which is carried out via phone calls. The client data contains 17 features that includes their demographics, i.e., age, education, job, and martial status, their financial status, i.e., bank balance in euros, personal and housing loan status, and credit default status, and campaign related variables. The target variable indicates whether the client subscribed to term deposit or not. For this assignment, we conducted experiments using three machine learning algorithms, Decision Tree, Random Forest, and Adaboost. The objective is to choose the optimal model for predicting where a client will subscribe to a bank term deposit.

Experiment

For each algorithm, I conducted at least 2 experiments. For Decision Trees, I experiments with different tree depths and pruning. For Random Forest and Adaboost, I adjusted the number of trees and tested different hyperparameters. The following metrics were used to evaluate the models: accuracy, precision, F1-score, and AUC -ROC. Below are the experiments I ran:

Decision Tree

The first model was a simple decision tree with default setting. It resulted a accuracy of 0.899, precision of 0.917, F1-score of 0.944, and AUC of 0.746. The next experiment was adjusting the max depth of the tree. The idea of limiting tree depth is to reduce overfitting and increase generalization. However, limiting the depth of the tree to a maximum of 3 has a drop in performance of 0.895. When the depth was increased to 5 and 10, the performance was the same as the first model with default setting.

Finally, I pruned the tree based on a optimal complexity parameter (cp) by plotting cp. The cp plot shows that as the size of the tree increases, the X-Val Relative Error initial decrease significantly and then level off. The high error at cp = infinite indicates underfitting as the single node tree is too simple to capture any underlying pattern. There is no evident of overfitting as we do not see cross-validation error increase as cp decrease. The optimal cp is 0.013 with the size of the tree = 6 as it has the lowest X-Val Relative Error. The pruned model resulted in a model similar to the default one. This indicate pruning did not greatly affect the performance of the model.

Random Forest

The default Random Forest model with 100 trees, performed better than the previous models with a accuracy of 0.905, precision of 0.920, F1-score of 0.948, and AUC of 0.928. Increasing the number of trees to 200 did not result a significant improvement in the model. In fact, the accuracy remains the same. Tuning the number of features with mtry at 6 as it has the lowest OOB error resulted a drop in accuracy but the precision was similar to the default model. Overall, the Random Forest model performed better than decision Tree model.

AdaBoost

The default Adaboost model also performed better than the Decision Tree models with a accuracy of 0.904, precision of 0.930, F1-score of 0.947, and AUC of 0.924. Another model with adjusted hyperparameters showed a slight drop in performance, with a slight decrease in accuracy and AUC, but precision remain high. Overall, the Adaboost Forest model also performed better than decision Tree model.

	Accuracy	Precision	F1_Score	AOC	Comment on Performance
Decision Tree (default)	0.8990562	0.9173027	0.9445412	0.7465744	Baseline Performance
Decision Tree (Max-depth = 3)	0.8946321	0.9165153	0.9419978	0.7445256	Slight drop due to restricted depth
Decision Tree (Max-depth = 5)	0.8990562	0.9173027	0.9445412	0.7465744	same as default
Decision Tree (Pruned)	0.8990562	0.9173027	0.9445412	0.7465744	simplified but similar performance as default
Random Forest (default)	0.9053974	0.9203554	0.9480462	0.9282924	Baseline Performance; highest F1-score
Random Forest (ntree=200)	0.9047338	0.9201007	0.9476754	0.9304559	Slight increase in AUC
Random Forest (Tuned-Mty=6)	0.9046601	0.9209552	0.9475775	0.9283361	similar to default
AdaBoost (Default)	0.9042177	0.9302119	0.9467295	0.9247494	Baseline Performance with high presion
AdaBoost (mfinal=100,cp= 0.001	0.9018581	0.9326181	0.9451790	0.9209830	lower AUc but has slightly better precision

Bias & Variance Analysis

In terms of bias, Decision Tree Model has the higher bias compared to the other algorithm, especially with the pruned tree and adjust maxed depth models as they were underfitting the data. The Random Forest model show highest variance, evident by the changes in performance of the metrics when adjusting the number of trees and mtry. This may be partly attributed to the class imbalance in the dataset, where the clients that said yes to subscription is underepresented. Comparing all models, Random Forest and Adaboost models consistently outperform the Decision Tree Model with the Random Forest with default setting (ntree=100) to be the best. It has the highest AUC, precision, and a strong F1-score.

Conclusion and Recommendation

The overall best model is the Random Forest with default setting due to it having the highest AUC, precision, and a strong F1-score compared to the other

models. In terms of business recommendation, the Random Forest model can be use to help identify the client who are likely to subscribe to term deposit.

One limitation of this analysis is the use of a single train-test split to evaluate model performance, which may cause some variation in results. Cross-Validation would give a more reliable testing across multiple data splits. For further enhancement, we can use Cross-Validation to improve accuracy and reduce bias.

Another further enhancement would be to address class imbalance in the data set by using SMOTE or adjust the class weights in the Random Forest model. It can potentially improve performance on the minority class.