COE379L - Project 02 Report

Techniques Used to Train Models

Data was first preprocessed to find for entries that needed to be imputed. No duplicate rows or null values were found. The independent variables were all float types, so further analysis for improper values was not conducted. However, features like describe() for a pandas dataframe were used to determine if any attributes of the data had anomalies. It was noted that AveRooms and AveOccups columns had max values that differed largely from their mean. This was further emphasized by univariate analysis using KDE plots of these columns. Thus, data standardization via RobustScalers was a considered preprocessing requirement, as the dataset contained outliers that deviate significantly from the mean. Furthermore, the data set was split 70% for train data and 30% for test data. Stratification was implemented to ensure accurate representation of the price_above_median class in both train and test data sets.

Now, four models were trained to predict house prices above the median in California. These models encompass K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and AdaBoost classifiers. Each model utilized a RobustScaler for preprocessing to effectively handle outliers in the dataset. Standardization was used in conjunction with pipelines to make the same preprocessing steps to be applied to test and training data.

Techniques Used to Optimize Model Performance

To optimize model performance, cross-validation was used to ensure our models were robust against overfitting. Furthermore a parameter grid was utilized for each model, to specify the hyperparameters to search for during our subsequent call to GridSearchCV(). Lastly, after fitting the model to the train data set, hyperparameters were found by accessing best_params_ attribute of each model.

Performance Comparison

In analyzing model performance to predict the binary classification of price_above_median, we can take a look at our generated confusion matrices. The Random Forest classifier had the best accuracy with the least false negatives and the least false positives. The KNN classifier had the worst accuracy with the most false negatives and the most false positives. The models can be ranked in the same order for both false positives and false negatives, from most to least: KNN, decision tree, AdaBoost, and random forest.

Model Recommendation

The random forest classifier is the clear model to be recommended. It achieved the best results on the test data for all metrics tested on. More specifically, the random forest model achieved 89% accuracy, 88% precision, 90% recall, and 89% for F1-score. The hyperparameters that produced these results are:

1) max_depth: 19
2) min_samples_leaf: 3
3) n_estimators: 26

Metric Importance

Now on metric importance, we used a default scoring because I believe finding hyperparameters that improved accuracy the most would allow for the most applicable use cases between the general public (buyers vs sellers). This is because if we focus on bettering the recall score, that would result in decreasing the number of false negatives detected. In context of the dataset, a lower false negative would reduce the chance of undervaluing a house. Therefore, if an individual pays less than what a house is worth, this would benefit the buyer, but since we are decreasing the likelihood of false negatives, this would benefit the seller. Now if we focus on bettering the precision score, that would result in decreasing the number of false positives detected. In context of the dataset, a lower false positive would reduce the chance of overvaluing a house. Thus, if an individual pays more than what a house is inherently worth, this would benefit the seller, but since we are decreasing that likelihood, we are benefiting the buyer. Because of this dependency, the overall most important metric to improve is accuracy because false negatives and false positives should arbitrarily improve as a result of an improvement in accuracy. This would also not allow a buyer or seller to have an upper hand in the housing market.