Auto-sklearn Fail

Farshad Ghorbanishovaneh

University of Wyoming

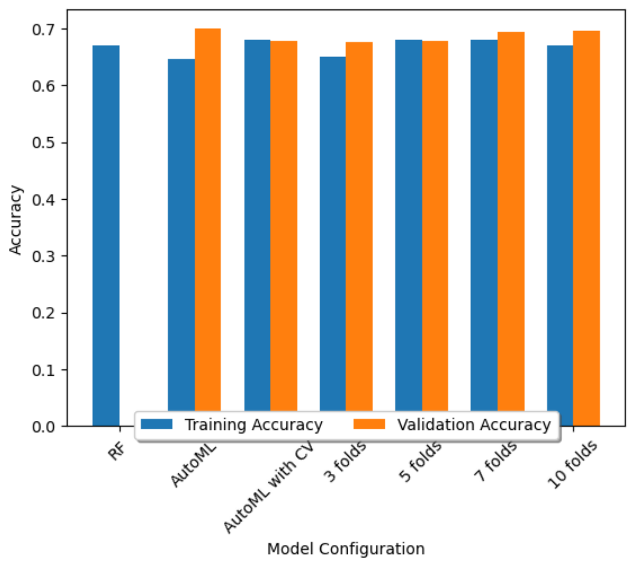

Figure 1: *Auto-sklearn results with different cross-validation folds.*

Auto-sklearn Fail

The auto-sklearn is an automated machine learning (AutoML) tool that automatically searches for the best machine learning model and hyperparameters for a given dataset. It is built on top of scikit-learn and uses Bayesian optimization to search for the best model and hyperparameters. The auto-sklearn is a powerful tool that can save a lot of time and effort when building machine learning models. In this case, I used auto-sklearn to predict wine quality from a dataset. Initially, auto-sklearn was set to run for 5 minutes, aiming to find the best hyperparameters and model. However, it could not outperform the Random Forest algorithm with default settings, which achieved an accuracy of 0.67. Auto-sklearn reached an accuracy of 0.6475, but on the validation dataset, it scored 0.699495, indicating potential overfitting.

To address this, I implemented a resampling strategy using cross-validation. When I varied the number of folds in the cross-validation, I observed different results: with 3 folds, the accuracy was 0.65 and the validation accuracy was 0.677231; with 5 folds, the accuracy improved to 0.68 and the validation accuracy was 0.678899; and with 7 folds, the accuracy remained at 0.68 with a validation accuracy of 0.694746. Further refinements with 10 folds resulted in an accuracy of 0.67 and a validation accuracy of 0.696414.

Additionally, I experimented with various preprocessing techniques, such as the Standard Scaler and PCA. However, these techniques generally led to a decrease in accuracy or had minimal impact, suggesting that the initial feature set was already well-suited for the model. The dataset features imbalanced classes, a small number of samples, and features with varying scales and distributions, all of which could significantly affect model performance. However, since Random Forest can effectively manage such datasets and the use of cross-validation helps mitigate these effects, these factors should not overly disadvantage the model.

Additionally, the hyperparameter search space for auto-sklearn may be too broad. A very broad space can render the search process inefficient, especially under time constraints, such as the 5-minute limit we faced. This could prevent auto-sklearn from identifying the most effective model configuration within the allotted time. The results illustrate in figure 1.