Week 5: Summary of Classification Predictors on the Titanic Survivor Dataset

Background and Motivation

This analysis was performed on the Titanic survival dataset, made available via the "Titanic - Machine Learning from Disaster" Kaggle competition (Kaggle 2022). The dataset contains records of over 1300 passengers of the "unsinkable" Titanic, categorizing the "training" dataset into those that survived and those that perished. The dataset contains details of each passenger's socioeconomic status, age, gender, and familial relationships, allowing analysts to identify the characteristics most strongly correlated with surviving disaster.

Approaches to Pre-Processing the Data

The original dataset provided by Kaggle contained 11 independent variables: two numeric (fare paid and passenger age) and nine categorical variables. Data was prepared in six tranches (with each step building on the last):

- Step 0: Basic preprocessing, including filling missing values and converting data that was captured as numeric into categorical
- Step 1: Feature engineering of numerics, namely by binning age and fare into discrete categories
- Step 2: Additional feature engineering, including the creation of a "Title" variable based on passenger name, a "Deck" feature based on ticket details, and indicators about family makeup
- Step 3: Create appropriate interaction effects.

Modeling Approach

Once the dataset was processed, models were developed using different tree-based classifiers: Random Forest, Gradient Boosting (XGBoost), and Extra Trees. Each modeling approach was run through a two-stage hyperparameter tuning: the first was a random grid search on a wide array of possible values; the second was a refinement to test local deviations from the best outcome of the random search.

Each model was evaluated using a cross-validation design, with model selection based on average accuracy. Once all models were built, a final stacked regression was created using the best-performing models from last week's assignment (Logistic Regression, LDA, and KNN) and this week's assignment (Random Forest and Extra Trees). Results:

Model	Cross-Val Accuracy	Train ROC AUC	Test Accuracy
LDA (last week's best)	83.05%	0.897	77.03%
Random Forest	82.61%	0.931	77.27%
Gradient Boost (1st)	82.27%	0.927	75.12%
Gradient Boost (2nd)	82.16%	0.956	75.36%
Extra Trees	81.93%	0.92	76.79%
Stacked Classifier			76.79%

Implications

- The first Gradient Boost model performed surprisingly poorly compared to the other models, and compared to the best-performing notebooks on Kaggle. Three potential reasons are that the feature engineering was insufficient, I focused my hyperparameter tuning efforts in the wrong direction, or the "best" models were over-tuned and therefore overfit. Given that the transformations I performed were relatively standard, my hypothesis is that overfitting was the main source of underperformance
- Random Grid Search is a useful way to quickly narrow down the possibility set when hyperparameter tuning. That said, repeated attempts at building the models sometimes revealed widely divergent best outcomes (e.g., number of estimators near the highest setting and model depth near the lowest setting in one run, and then switching in the next). This approach forces one to recognize that it is likely impossible to find the absolute best set of parameters and hyperparameters.

References

"Titanic - Machine Learning from Disaster." Kaggle. Accessed January 24, 2022. https://www.kaggle.com/c/titanic/overview.