Playground Series – Season 3, Episode 9

Tabular Regression with Concrete Strength Dataset

**Problems Encountered**:

1. There are too many valid duplicates meaning in both train and test case there exists different target values for same feature values(the great duplicate saga)
2. Any feature engineering mostly reduces the leaderboard and cv score even though new features has high correlation
3. Leaderboard score does truly represent the actual score as test size is very big but public leaderboard sample size is small
4. Hyperparameter tuning using optuna or any other method is not reliable as the leaderboard score decreases because of them
5. Hyperparameters fails to generalize on test

**Solutions:**

|  |  |
| --- | --- |
| Solutions | Links |
| #1 Solution: Cross-validation and diversity win | **https://www.kaggle.com/competitions/playground-series-s3e9/discussion/394592** |
| #12th place solutions: My 6 step process for any competition | **https://www.kaggle.com/competitions/playground-series-s3e9/discussion/394600** |
| #10 - Nothing fancy, tuning and stacking with tidymodels | **https://www.kaggle.com/competitions/playground-series-s3e9/discussion/394654** |
| #44 place solution using features based on LinearRegression | **https://www.kaggle.com/competitions/playground-series-s3e9/discussion/394641** |

**What I Have Learned:**

* **It is better to do competition in six steps/notebook like mentioned by 12th place solution(all of them need further learning by me)**
  + EDA
  + Feature Creation
  + Feature Selection (RFECV and manual selection)
  + Single Model Tuning
  + Multi Model CV / Multi Model : best fold
  + Ensembling
* **If number discrete feature values are low you can try target encoding(like AgeInDays)**
* **If public leaderboard uses very small amount of test data compared to all of the test data then it is better to not trust leaderboard score as they may not represent full test set. In general it is better to not look into public leaderboard**
* **Optuna hyperparameters may not survive the change of seed in kfold so Optuna may not always bet the best way to do hyperparameter tuning so it may be better to just manually do hyperparameter tuning**
* **If optuna finds you some good hyper parameters then test how effective they are by changing their values very little if the score changes a lot that mean the hyper parameters are probably not that reliable**
* **Random forests are simple to optimize: The most important hyperparameter is “min\_samples\_leaf.” Which has too low value of default just 1**
* **SHAP can be used to learn feature relations as they are very reliable compared to randomfeatures**