Testing Differences in Performance of Pricing Models

Jose Pliego San Martin

Duke University, Department of Statistical Science

INTRODUCTION & MOTIVATING EXAMPLE

Accurately pricing policies is key for insurance companies. Under-pricing policies results in losses for the company, while over-pricing policies may result in customers turning to competitors for a better quote or even sanctions from regulatory authorities. Tree-based models have shown great predictive performance in this setting; however, they tend to be unstable in the sense that optimal hyperparameters, the effect of different features on the response, and often predictive performance are very sensitive to changes in the training data.

Example. The goal is to fit a severity model to predict the costs associated with each insurance policy in the next period. Two different models are fitted. Model 1 uses all the 100 available predictors and has a root mean squared error (RMSE) of 0.536 in the validation set. Model 2 uses 70 of the predictors and has a RMSE of 0.527. Is model 1 significantly better than model 2, or **is it possible that under different train-validation splits the more parsimonious model exhibits better predictive performance?**

OBJECTIVES

Design a modeling pipeline that:

1. Allows for a more **robust estimation** of model performance.
2. Implements a procedure to test if differences in model performance are **statistically significant**.
3. Can be easily adapted to different amounts of **computational resources**.
4. Can be used to study various model characteristics (e.g., feature importance).

MODELS & METHODS

Gradient Boosting Machines (GBMs)

This work focuses on the specific tree-based models known as GBMs. In a traditional modeling pipeline for GBMs, there are four main sources of randomness over which statisticians have some control:

1. How the data is split into training and validation sets.
2. Hyperparameter optimization procedures (e.g., Bayesian optimization).
3. GBM bootstrap samples.
4. GBM feature samples.

After multiple experiments, the first two were identified as the main sources of variability. These two sources of randomness are not specific to GBMs, so the methods mentioned here can be useful for any Machine Learning Algorithm.

Repeated Cross Validation

Cross validation (*k*-fold CV) consists of splitting the training data into *k* disjoint sets and using each of these sets as a validation set once and the other *k-1* as a training set.

- Large *k* implies **less bias and more variance.**
- Small *k* implies **more bias and less variance.**

The idea of repeated cross validation is to repeat *k*-fold CV a total of *r* times, to obtain a total of $r \times k$ performance observations and avoid issues when choosing a "good" number of folds.

"Corrected" *t*-statistic

A paired *t*-test can be performed with the $r \times k$ performance observations to test whether one model is significantly better than another. Since observations are not independent due to cross validation, an adjustment [1] must be made to achieve desired type I error probabilities:

$$t = \frac{\sum_{i=1}^{r}\sum_{j=1}^{k} x_{ij}}{\sqrt{\left(\frac{1}{k \times r} + \frac{n_2}{n_1}\right)\hat{\sigma}^2}}$$

CASE STUDY

The models and methods introduced are applied in a real-life scenario using insurance data from a Kaggle competition [2]. The goal is to predict the severity of claims for multiple policies issued by the company.

The training data set consists of 194,000 policies. The columns are an ID column, a continuous response column (average claim severity for each policy), 116 categorical predictors, and 14 numerical predictors.

Still haven't finished this section

CONCLUSIONS

1. The multiple measurements obtained with repeated cross validation **yield a more robust estimation of model performance.**
2. **Differences in performance can be statistically assessed** using an adjusted *t*-test that accounts for the fact that cross validation folds are not independent.
3. Repeated cross validation allows for **easy parallelization** over multiple cores. Furthermore, the number of repeats and folds are not fixed, allowing for **adaptation to different amounts of computational resources**.
4. Since repeated cross validation fits multiple models, all of them can be used to **study certain characteristics like feature importance or Shapley values** (for model interpretability).

REFERENCES

[1] Nadeau, C., & Bengio, Y. (2003). Inference for the generalization error. *Machine Learning, 52(3),* 239-281. https://doi.org/10.1023/A:1024068626366

[2] Allstate Insurance. Allstate claims severity. Retrieved January 2023, from https://www.kaggle.com/competitions/allstate-claims-severity/data

ACKNOWLEDGEMENTS

This project was developed during the Summer of 2022, while the author was working as an intern Data Scientist at Liberty Mutual Insurance.