

# Testing Differences in Performance of Pricing Models

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. **Gradient Boosting Machines (GBMs)** have shown great predictive performance in this setting; however, they tend to be unstable since they have a lot of hyperparameters and are prone to overfitting without adequate tuning.

**Example.** New regulation states that insurance companies cannot use certain client characteristics in pricing models. How does the removal of these predictors affect model performance?



## OBJECTIVES

- Transform the model training pipeline so that it:
- Allows for a more **robust estimation** of model performance.
  - Implements a procedure to test if differences in model performance are **statistically significant**.
  - Effectively adapts to different **computational and time resources**.

## MODELS & METHODS

### Gradient Boosting Machines (GBMs)

- Boosting is a technique that combines several weak learners into one big model.
- In GBMs, the weak learners are usually decision trees.
- Two of the most popular implementations are XGBoost and LightGBM.
- Hyperparameter tuning is crucial to avoid overfitting.

### Repeated Cross Validation (RCV)

- Cross validation ( $k$ -fold CV): split training data into  $k$  disjoint sets and use each as a validation set once.
- Repeated cross validation: perform  $k$ -fold CV a total of  $r$  times to obtain  $r \times k$  performance observations.
- Hyperparameter optimization using all the validation sets yield over-optimistic performance estimations.

### Nested Cross Validation (NCV)

- Within a cross validation fold, split the training set again using  $k$ -fold CV to tune hyperparameters.
- Since the validation sets are not used to tune, performance observations are less biased in general.
- Depending on the number of “outer” and “inner” folds, it can be computationally expensive.

### ”Corrected” $t$ -statistic

- Statistically assess differences in performance using a paired  $t$ -test.
- Need to account for the fact that training sets in cross validation are not independent to stabilize type I error rates [1].

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

## CASE STUDY (I)

- Claim severity data from a Kaggle competition [2].
- The goal is to predict the severity associated with each claim (continuous response, MAE loss).
- 194,000 claims, 116 categorical predictors, 14 numerical predictors.

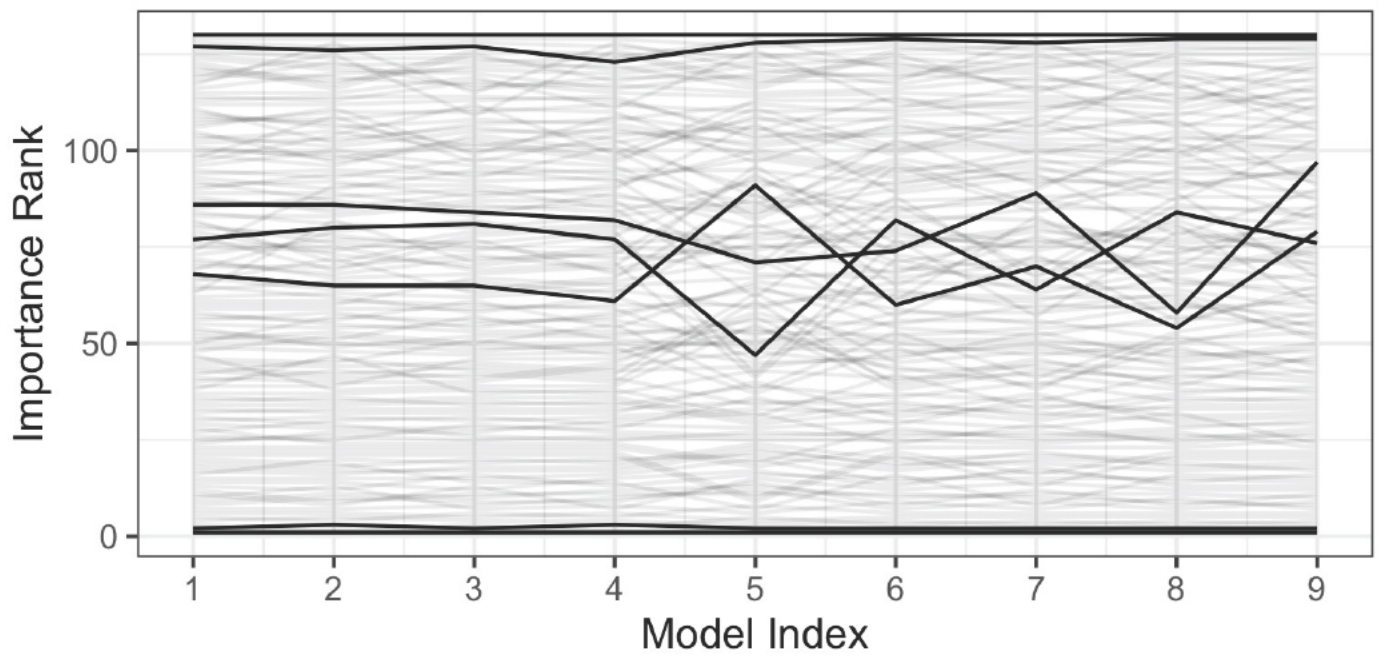
Comparing two cross validation approaches.

Cross Validation	Avg. MAE	SD MAE	Time*
Nested RCV	1149.73	10.52	2h 35m
Non-nested RCV	1148.85	9.69	1h 2m

\* Tested using 5 cores in parallel on a 2022 MacBook Air with 8 GB RAM and M2 chip (MacOS Ventura 13.1).

- Both methods yield similar results, the non-nested approach seems to be slightly optimistic.
- The nested procedure requires considerably more computation time.

### NCV estimations of variable importance



Does removing the two least important predictors affect performance?

Model	Avg. MAE	SD MAE	Avg. Diff.	$t$ -statistic
Full	1146.34	9.46	2.35	2.06
Reduced	1143.99	8.51		

$p$ -value = 0.0507, obtained with a  $t_{r \times k - 1}$  distribution.

- The reduced model shows better predictive performance!
- The difference in performance is not significant at a 5 % level.

## CASE STUDY (II)

How does the model perform on a real Kaggle submission?

Final Test Score	Public Kaggle Score	Private Kaggle Score	Kaggle Winning Score
1127.47	1122.27	1134.54	1109.71

- The test set accuracy overestimates the public Kaggle score and underestimates the private Kaggle score.
- The public score is calculated on 30 % of a holdout set, the private score is calculated on the other 70 %.

## CONCLUSIONS

- Multiple measurements obtained with repeated cross validation yield a **more robust estimation of model performance**.
- Differences can be statistically assessed** using an adjusted paired  $t$ -test.
- Repeated (nested or not) cross validation allows for **easy parallelization**. The number of repeats and folds can be adjusted to **optimally use computational and time resources**.

## 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 (2016). *Allstate claims severity*. Kaggle. Retrieved from <https://www.kaggle.com/competitions/allstate-claims-severity>

## ACKNOWLEDGEMENTS

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