All Models United: An AutoML Collective

Ziad Helal, Zihan Xiu, Amulya Nataraja
Team-Betelgeuse

Modality 1

Motivation

- **Problem Statement:** The challenge is to build an efficient and robust AutoML solution for tabular data that can generalize well to unseen test sets.
- **Objective:** Develop an automated pipeline capable of exploring and tuning a range of models to maximize predictive performance, leveraging the strengths of ensemble learning and transfer learning.
- **Significance:** Achieving superior model performance in AutoML can reduce manual effort and improve the applicability of machine learning to diverse real-world datasets.

Contributions

- **Comprehensive Pipeline:** Implemented a pipeline that automates feature engineering, hyperparameter optimization, and model ensembling using cutting-edge libraries and techniques.
- **Transfer Learning:**everaged previous optimization results to initialize subsequent dataset training, reducing search time and improving model efficiency.
- **Ensemble Approach:** Created a voting ensemble integrating XGBoost, LightGBM, and CatBoost, enhancing generalization on the exam dataset.

- Week 1 (checked)
- Week 2
- Week 3 (checked)
- Week 4 (checked)
- Week 5
- Week 6
- Week 7
- Week 8
- Week 9
- Week 10
- Bonus
- Literature

Our Approach

Workflow

- **Data Acquisition and Preprocessing:** Download datasets from OpenML. Preprocess, clean, and scale data using standard practices.
- **Separate Objective Functions:** Developed distinct objective functions for XGBoost, LightGBM, and CatBoost, each tailored to optimize hyperparameters for maximizing R^2.
- **Iterative Optimization:**
  - Used Optuna to independently optimize each model type across practice datasets, with cross-validation for robust evaluation.
  - Enqueued best parameters from previous datasets to warm-start new searches.
- **Model Ensembling:** Formed a Voting Regressor ensemble with the best configurations to combine model strengths effectively.

Benefits

- **Robust Evaluation**: The use of cross-validation ensures that the models generalize well and are not overfitting to a specific train-test split.
- **Efficient Hyperparameter Tuning with hyperband**: By optimizing each model independently, you can exploit each algorithm's strengths and find the most suitable hyperparameters.
- **Transfer Learning**: Using best-performing configurations from one dataset as a starting point for another can reduce search time and leverage commonalities between datasets.
- **Ensemble Learning**: Combining the models helps capture different aspects of the data, often leading to improved predictive performance compared to individual models.

Resources Used

For development:
- 1 T4 Nvidia GPU
- Quad-core i7 Intel CPUs
- Total compute estimate: 20 CPU-h

For AutoML:
- Google Colab
- 1 T4 Nvidia GPU
- 2h 30min

Workforce:
- 2 full week on average

Number of queries for test score generation: **1**

Results

Learning Curve:

- **Performance over Time:** Early on in the optimisation process, all three models show strong R2 scores; however, there are variations, particularly with CatBoost, which gradually displays significant drops and recoveries.
- **Model Behavior: CatBoost:** Displays significant fluctuations, with notable drops and recoveries.
  - **LightGBM:** Exhibits more stable performance with less abrupt changes in R^2 scores.
  - **XGBoost:** Shows relatively consistent performance.
- **Speed of Convergence:** LightGBM and XGBoost typically converge to high R^2 scores more rapidly, while CatBoost tends to take longer and perform more inconsistently.
- **Model Efficiency:** When considering time efficiency, LightGBM and XGBoost appear to be more reliable options for this dataset, based on their ability to achieve high R^2 values consistently.

Performance Heatmap:

- LightGBM has high R2 values in the majority of trials, including many flawless R2 values of 0.99. Its lower performance peaks, such as trials 4, 5, and 6, are present.
- CatBoost performs admirably as well; several trials have an R2 value of 0.99. There are a few notable performance drops (trials 1, 6, 7, 8, and 10 are among them).
- XGBoost often exhibits good performance, with R2 values frequently approaching or equal to 0.99. It receives a few lower trial ratings, including 6, 10, and 17.

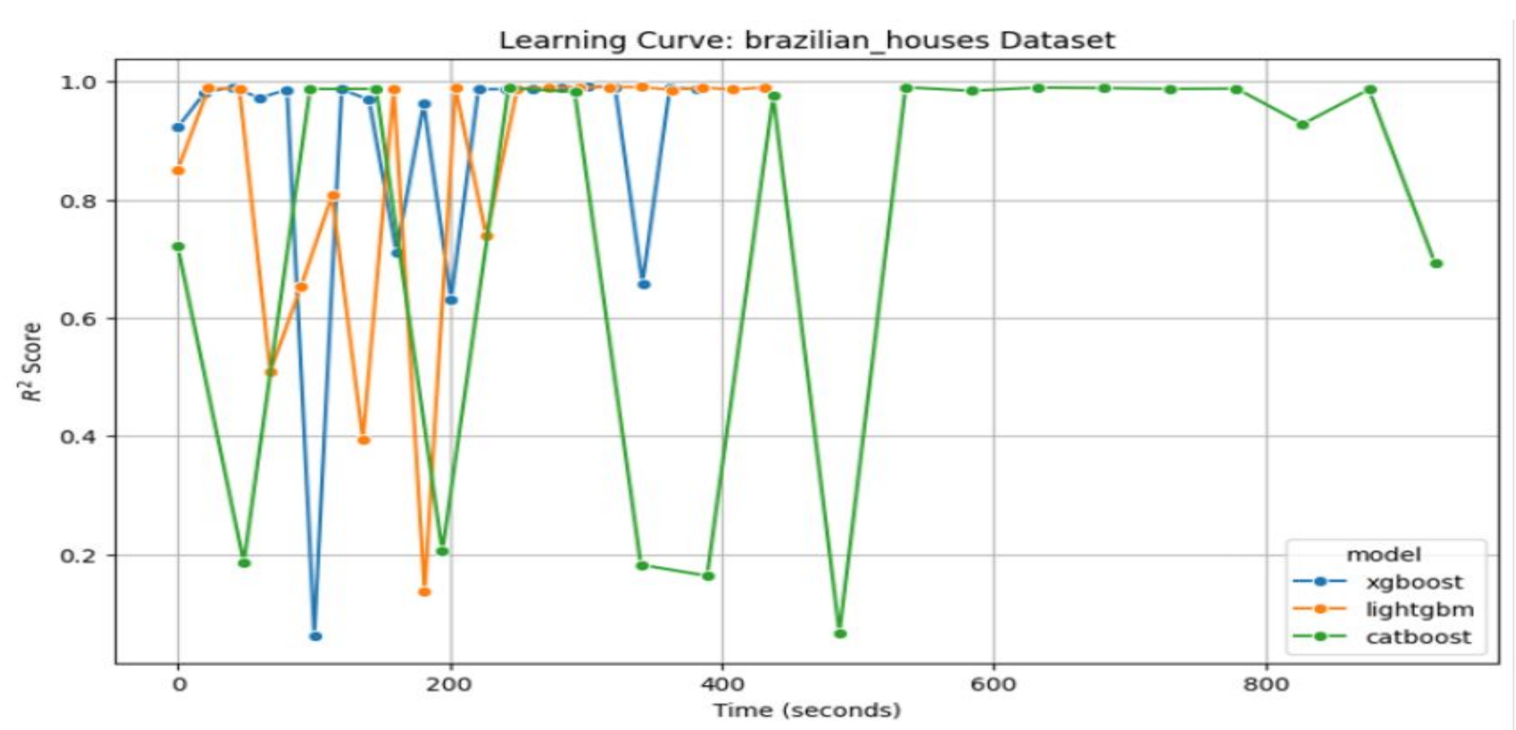

Fig 1: Learning Curve - Example Dataset - Brazilian Houses

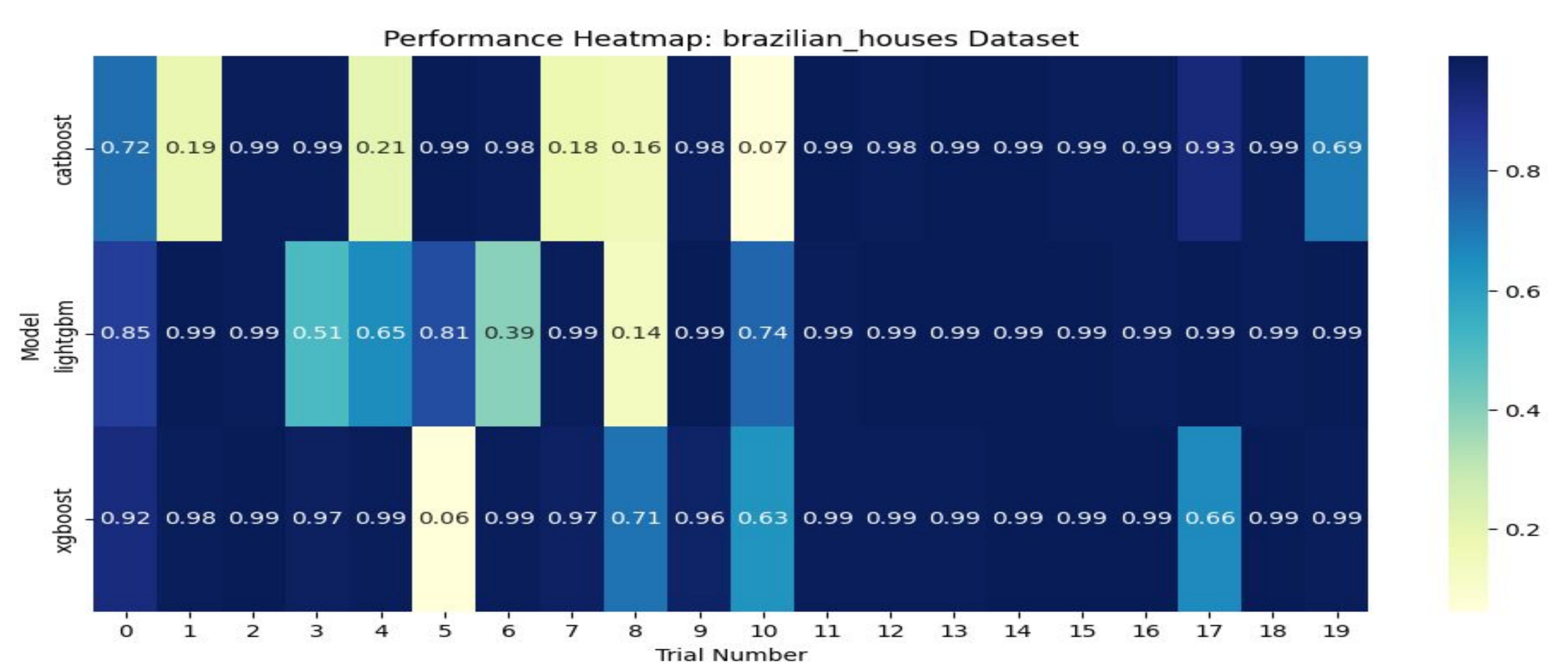

Fig 2 : Performance Heatmap - example dataset - Brazilian house