A Comparative Study of Classical and Modern Boosting Algorithms for Supervised Learning

Tony Luo, Will Kim, Sichen Li, Shang Peng, Jiaheng Guo

Problem Description

We aim to conduct a comparative analysis of *boosting algorithms* for supervised learning on tabular data. Building upon the algorithms introduced in class, our study will (i) summarize and implement two classical boosting methods—**AdaBoost** and the **Gradient Boosting Machine (GBM)**—and (ii) evaluate to what extent more recent implementations—**XGBoost** and **LightGBM**—achieve practical improvements in predictive performance under consistent experimental settings.

The central research question is: *How do modern gradient-boosting frameworks improve upon classical boosting methods in design and empirical performance?* Specifically, we will examine which design innovations—such as regularization, histogram-based optimization, and leaf-wise tree growth—contribute to observed gains in scalability and accuracy.

Experiments will be conducted on several publicly available Kaggle datasets spanning both classification and regression tasks to ensure that findings are representative rather than dataset-specific.

Related Works

Classical boosting methods, such as **AdaBoost** and the **Gradient Boosting Machine (GBM)**, established the theoretical foundations of ensemble-based learning. AdaBoost constructs a strong classifier through sequential reweighting of misclassified samples [2], whereas GBM generalizes this principle by fitting weak learners to the negative gradient of a differentiable loss function [3]. Both can be viewed as stagewise additive models that iteratively minimize a loss, offering robustness and competitive accuracy across diverse domains.

Modern implementations—**XGBoost** [1] and **LightGBM** [4]—build directly upon Friedman's framework while introducing system-level and algorithmic optimizations to address scalability limitations. XGBoost incorporates explicit regularization, sparsity-aware split finding, and parallelized tree construction, while LightGBM accelerates training via histogram-based gradient estimation and leaf-wise tree growth with depth constraints. These refinements have made both models standard benchmarks for tabular machine-learning tasks.

By comparing classical and modern boosting methods within a unified experimental setup, we aim to assess whether these algorithmic innovations yield measurable gains in practice.

Proposed Work

Algorithms. The project will compare two classical and two modern boosting methods:

- **Classical:** AdaBoost [2], Gradient Boosting Machine (GBM) [3]
- **Modern:** XGBoost [1], LightGBM [4]

Datasets. Several publicly available Kaggle datasets will be used to cover both regression and classification tasks, including but not limited to:

- *Classification:* Adult Income, Heart Disease
- *Regression:* House Prices, California Housing

Including both regression and classification tasks allows us to evaluate algorithm robustness across distinct loss functions.

Preprocessing. Data cleaning will include missing-value imputation and appropriate encoding of categorical variables. All datasets will be randomly partitioned into training and testing subsets using a fixed random seed for reproducibility.

Training and Validation. Each model will be trained on identical data splits with comparable hyperparameter tuning budgets. Randomized or grid search will be applied for parameters such as number of estimators, learning rate, and maximum tree depth. Results will be averaged across multiple random seeds.

Analysis. We will compare predictive metrics and examine feature importances to interpret learned relationships. Additionally, we will analyze the trade-offs between model complexity and interpretability, highlighting how structural and regularization choices influence generalization.

Evaluation Metric

Performance Metrics.

- *Classification:* Accuracy, Precision, Recall, F1-score, and ROC–AUC
- *Regression:* Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2

Model performance will be estimated via k-fold cross-validation ($k = 10$) using identical folds across algorithms. Average scores and standard deviations will be reported to quantify stability. Hyperparameter tuning will be performed within each training fold (nested CV) to prevent information leakage between training and testing data.

References

[1] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. Association for Computing Machinery.

[2] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[3] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[4] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pages 3146–3154, Long Beach, CA, USA, 2017. Curran Associates, Inc.