CW1 Regression Challenge Report

K23154082

1 Introduction

This coursework is a supervised regression task: given a tabular dataset, we predict the target variable `outcome` from the remaining features. Performance is evaluated using out-of-sample R^2 on a held-out test set sampled from the same distribution. Our goal is to build a robust pipeline that maximizes generalization via systematic model comparison and cross-validation.

2 Exploratory Data Analysis and Preprocessing

We first inspected dataset dimensions, data types, and missingness. Since the dataset is tabular and includes categorical variables, we used one-hot encoding (`pandas.get_dummies`) to convert categorical features into numeric indicator variables. Missing values were handled using median imputation, which is robust to outliers and simple to implement.

To avoid information leakage, all preprocessing steps (imputation and model fitting) were wrapped in a `scikit-learn Pipeline` and evaluated using 5-fold cross-validation (CV) with shuffling and a fixed random seed. For test-time prediction, we aligned train/test one-hot encoded columns by reindexing the test matrix to match the training feature set (filling absent categories with zeros).

3 Model Selection

We compared several regression models under the same 5-fold CV protocol using R^2 as the scoring metric. Table 1 shows the mean and standard deviation across folds. Linear models (Linear Regression, Ridge) perform substantially worse than non-linear ensemble methods, indicating that the data-generating process likely contains non-linearities and/or feature interactions. Among the tested methods, gradient boosting achieved the strongest generalization.

4 Model Training and Hyperparameter Tuning

We selected `HistGradientBoostingRegressor` as the final model due to its superior CV performance. Hyperparameters were optimized using `RandomizedSearchCV` with 60 candidates and 5-fold CV. The best configuration achieved a mean CV R^2 of 0.4748.

The tuned parameters were:

- `learning_rate = 0.01`, `max_iter = 2000`
- `max_depth = 2`, `max_leaf_nodes = 63`
- `min_samples_leaf = 120`, `max_bins = 128`
- `l2_regularization = 0.0`

This configuration favors smoother fits (small learning rate with many iterations and shallow trees) and larger leaf sizes, which reduces variance and improves generalization on this simulated dataset.

Table 1: 5-fold cross-validation performance (R^2).

Model	Mean R^2	Std.
Linear Regression	0.2711	0.0159
Ridge ($\alpha = 100$)	0.2855	0.0160
Random Forest	0.4417	0.0132
HistGradientBoosting (baseline params)	0.4632	0.0211
HistGradientBoosting (tuned)	0.4748	–

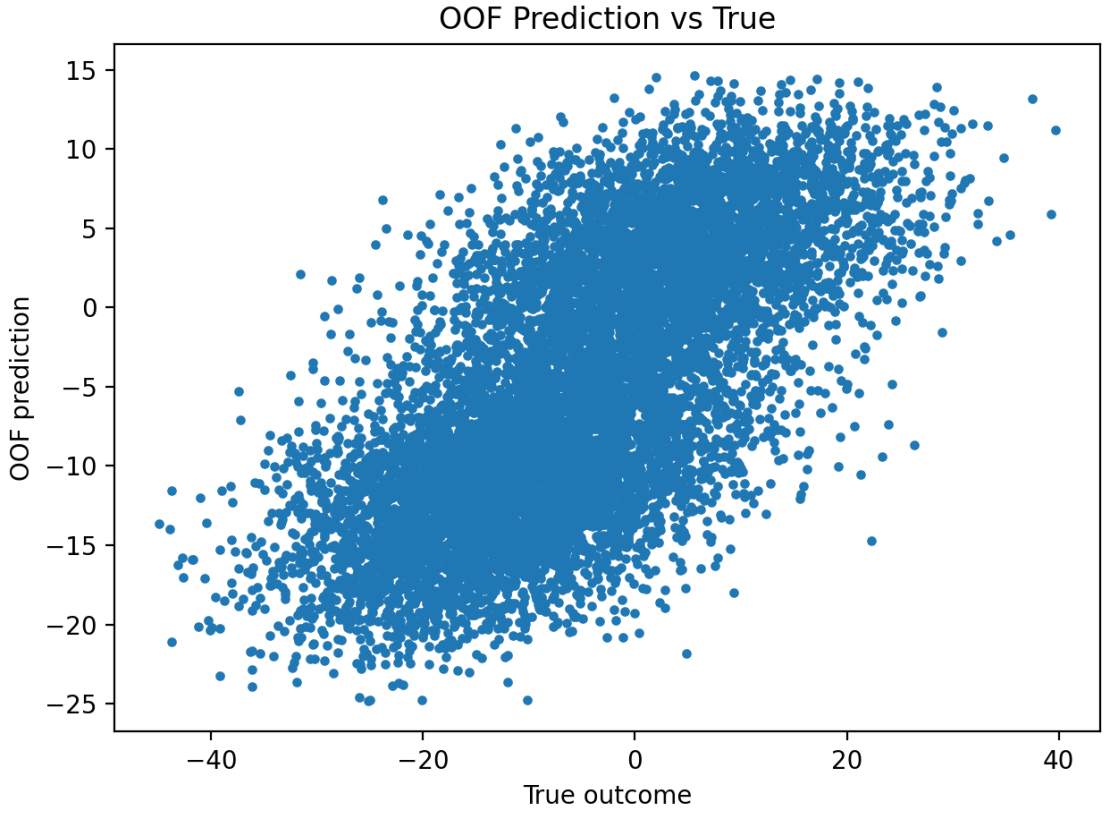

Figure 1: Out-of-fold predictions versus true outcomes for the tuned HistGradientBoostingRegressor.

5 Evaluation and Diagnostics

To assess generalization, we computed out-of-fold (OOF) predictions under 5-fold cross-validation. Figure 1 shows a clear positive association between predictions and true targets, indicating that the model captures substantial signal. The predicted values have a smaller range than the targets, implying underestimation for extreme outcomes. Residuals (not shown) are approximately centered around zero, with occasional large errors, consistent with noisy targets and/or unmodelled interactions.

6 Code Supplement and Reproducibility

All experiments were run with a fixed random seed (42) and evaluated using 5-fold cross-validation to ensure reproducibility. The implementation includes scripts for model training, cross-validation, hyperparameter tuning, and generation of the submission file (`CW1_submission_K23154082.csv`).

Code repository / appendix: `https://github.com/Irohas0416/MLF-CW1`.