CW1 – Machine Learning Pipeline Report

5CCSAMLF – Machine Learning

K23114605

February 2026

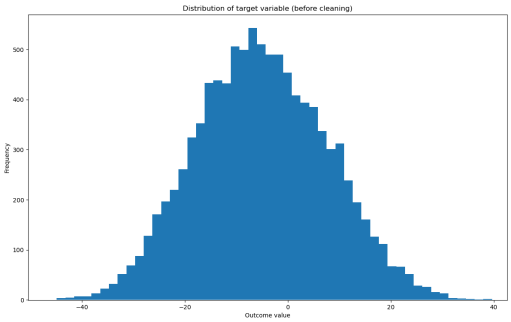
1 Exploratory Data Analysis

The training set contains 10,000 diamonds with 30 features and a continuous target variable (`outcome`). Features include standard diamond attributes and 20 additional numeric columns (`a1`–`b10`). No missing values were observed.

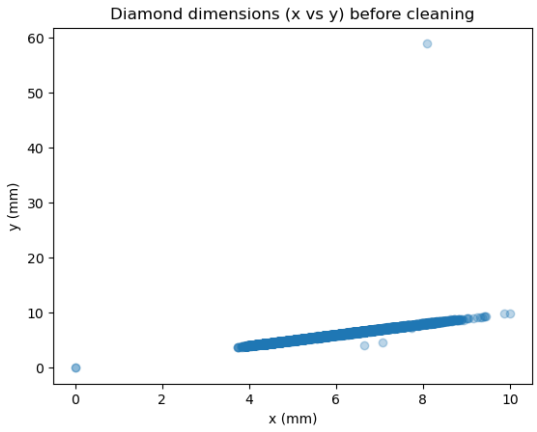
Target distribution. The outcome variable is approximately normally distributed (mean ≈ -5.0, standard deviation ≈ 12.7; Fig. 1a), so no transformation was applied.

Correlation analysis. Pearson correlation was used to examine linear relationships with the target. The strongest correlation was observed for `depth` ($r = -0.41$). Some unnamed features showed moderate correlations, including `b3` ($r = 0.23$), `b1` ($r = 0.17$), and `a1` ($r = 0.15$). In contrast, core attributes such as carat ($r \approx 0.00$) and price ($r = 0.02$) had little linear relationship with `outcome`. The dimensions x, y, and z were highly correlated with carat ($r > 0.97$); this was retained as tree-based models are robust to multicollinearity.

Data cleaning. Rows with impossible measurements (e.g., $x = 0$, $y = 0$, or $z = 0$) were removed. Conservative bounds were also applied to `depth`, `table`, and physical dimensions to remove extreme outliers. After cleaning, 9,995 samples remained. Figure 1b illustrates the dimension outliers before cleaning.



(a) Distribution of the target (`outcome`).



(b) Dimensions before cleaning.

Figure 1: Main EDA plots.

2 Model Selection

Four regression models with increasing complexity were compared:

1. **Ridge Regression** ($\alpha = 5.0$)
2. **Random Forest** (400 trees)
3. **XGBoost** (500 trees, learning rate = 0.05, depth = 5)
4. **Gradient Boosting Regressor (GBR)** (500 trees, learning rate = 0.03, depth = 3)

Tree-based ensemble models were prioritised due to non-linear relationships, correlated inputs, and mixed feature types.

3 Model Training and Evaluation

Preprocessing. Categorical features were one-hot encoded. Numerical features were median-imputed and standardised. All steps were implemented within an sklearn `Pipeline` to prevent data leakage.

Cross-validation. Models were evaluated using 5-fold cross-validation with the R^2 metric (Table 1).

Table 1: 5-fold cross-validation results.

Model	**Mean R^2**	**Std**
Ridge Regression	0.2853	0.0181
Random Forest	0.4557	0.0095
XGBoost	0.4631	0.0142
Gradient Boosting	**0.4719**	**0.0098**

GBR achieved the highest average R^2 with low variance, indicating stable performance. Ridge Regression underperformed, suggesting that linear assumptions were insufficient. XGBoost was competitive but slightly inferior, possibly due to higher model complexity.

Hyperparameter tuning. GBR was further optimised using `RandomizedSearchCV` (20 iterations, 5-fold CV) over the following space:

Table 2: GBR hyperparameter search space.

Parameter	**Values**
`n_estimators`	{100, 200, 300, 500}
`learning_rate`	{0.01, 0.05, 0.1}
`max_depth`	{2, 3, 4}
`subsample`	{0.6, 0.8, 1.0}

The best configuration was refit on the full cleaned training set and used to generate predictions for the test data.

Code Supplement

The full implementation is provided in `train_and_submit.py`. GitHub: `https://github.com/aytenmarab/ml_cw1`

Bibliography

@misckapoor2023diamond, author = Kapoor, K., title = Diamond Price Prediction, howpublished = Kaggle Notebook, year = 2023, url = https://www.kaggle.com/code/karnikakapoor/diamond-price-prediction, note = Accessed: 19 February 2026