Machine Learning Project

Predicting Fish Weight Using Machine Learning

---

1. Introduction

Accurately predicting the weight of a fish based on physical attributes is a valuable task in fields such as aquaculture, fisheries management, and food logistics. This report presents an end-to-end machine learning pipeline to estimate fish weight from morphological features. It evaluates multiple regression models and identifies the most effective one based on a comprehensive set of performance metrics.

---

2. Problem Statement

Given the physical measurements of a fish, including lengths, height, width, and species, the goal is to predict its weight. The prediction is treated as a supervised regression problem.

---

3. Dataset Description

The dataset used comes from the Fish Market dataset on Kaggle, which contains the following features:

- Species (categorical)
- Weight (target)
- Length1, Length2, Length3
- Height
- Width

A new feature, `Volume_Estimate`, was engineered using the formula:

```
Volume_Estimate = Length1 * Height * Width
```

This feature captures a crude volumetric approximation and significantly improves predictive power.

---

4. Exploratory Data Analysis (EDA)

Prior EDA highlighted relationships between the target (Weight) and numerical features. The feature `Height` and the new `Volume_Estimate` showed strong correlation with `Weight`. A few outliers with `Weight = 0` and apparent annotation errors were removed.

---

5. Preprocessing

The preprocessing steps included:

- Dropping invalid and outlier entries
- Creating `Volume_Estimate`
- Standard scaling for numeric features
- One-hot encoding for the `Species` categorical feature
- Train-test split (80-20 ratio)

---

6. Model Training and Evaluation

Four models were trained and evaluated:

- Linear Regression
- Decision Tree Regressor
- Random Forest Regressor
- Gradient Boosting Regressor

Each model was assessed on the following metrics:

- Train and Test RMSE

- Test MAE
- Test R^2 Score
- Cross-validated R^2 Mean and Std

Results:

Model	Train RMSE	Test RMSE	Test MAE	Test R^2	CV R^2 Mean	CV R^2 Std
Linear Regression	46.08	40.89	28.64	**0.9850**	0.9666	0.0119
Gradient Boosting	6.66	48.65	34.19	0.9788	0.9668	0.0185
Random Forest	19.59	52.37	32.98	0.9755	0.9658	0.0232
Decision Tree	0.00	68.37	42.45	0.9582	0.9126	0.0434

---

7. Best Performing Model

The best performing model was **Linear Regression**, which achieved the highest R^2 score of **0.9850** on the test set. Despite its simplicity, it outperformed the tree-based ensemble models, likely due to the strong linear relationship between the input features and the target variable. The model was less prone to overfitting and generalized well on unseen data, as shown by the close values of Train and Test RMSE.

---

8. Visualizations

Multiple plots were generated to analyze model behavior:

- Actual vs. Predicted scatter plots
- Residual plots
- Error distribution histograms
- Learning curves

- Feature importance bar plots

These visualizations confirmed that Linear Regression had the most stable residuals and lowest variance between training and test scores.

---

9. Hyperparameter Tuning

Although tree-based models like Random Forest and Gradient Boosting were tuned using GridSearchCV, their performance did not exceed that of Linear Regression. Hyperparameter tuning did improve R^2 scores for these models but was still marginally lower than the Linear model.

---

10. Conclusion

This study demonstrates that a well-preprocessed Linear Regression model can outperform complex ensemble models when data has a strong linear structure. The inclusion of engineered features like `Volume_Estimate` proved to be a key enhancement. Future work can explore more advanced feature extraction and potential non-linear modeling if the dataset becomes more complex.
