Model Validation and Hyperparameter Optimization in Machine Learning

1. Introduction

Machine learning models often perform well on training data but fail on unseen data due to overfitting. This project focuses on understanding model validation techniques and improving model performance using cross-validation and hyperparameter tuning. The goal is to build a model that generalizes well rather than memorizing the training data.

2. Objective

The objectives of this project are:

- To understand overfitting and underfitting
- To evaluate models using proper validation techniques
- To apply cross-validation for stable performance measurement
- To tune hyperparameters using GridSearchCV
- To compare baseline and optimized models

3. Dataset Description

The California Housing dataset from scikit-learn is used in this project. It contains multiple numerical features related to housing conditions, and the target variable represents house prices. The dataset is suitable for regression-based machine learning tasks.

4. Methodology

4.1 Train-Test Split

The dataset is divided into training and testing sets. The training set is used to train the models, while the test set is used to evaluate their performance on unseen data.

4.2 Baseline Model

A Linear Regression model is used as the baseline model. It provides a reference point to compare the performance of more complex models.

4.3 Overfitting Observation

A Decision Tree Regressor is trained without any restrictions. This model shows very low training error but high testing error, indicating overfitting.

4.4 Cross-Validation

Cross-validation is applied to measure model performance more reliably. It helps reduce bias caused by a single train-test split and provides a better estimate of real-world performance.

4.5 Hyperparameter Tuning

GridSearchCV is used to tune hyperparameters such as maximum depth and minimum samples per split. This helps control model complexity and reduce overfitting.

5. Evaluation Metrics

- **RMSE (Root Mean Squared Error): 0.6390654005312799**

Measures the average magnitude of prediction errors.

- **R^2 Score: 0.6883380738855668**

Indicates how well the model explains the variance in the target variable.

6. Results and Comparison

- Linear Regression provided a baseline performance.
- Untuned Decision Tree showed overfitting.
- Tuned Decision Tree achieved better generalization with lower RMSE and improved R^2 score.

The tuned model performed better on unseen data compared to the untuned model.

7. Conclusion

This project demonstrates the importance of model validation and hyperparameter tuning in machine learning. Cross-validation and GridSearchCV significantly improve model reliability and prevent overfitting. Proper evaluation ensures better model performance in real-world applications.

8. Future Scope

- Try advanced models such as Random Forest or Gradient Boosting
- Use larger real-world datasets
- Deploy the tuned model using a web interface