Report

Kaggle Competition: "House Prices - Advanced Regression Techniques"

Objective

The objective of the competition was to predict house prices based on a set of features. Various machine learning models were employed to achieve accurate predictions.

Libraries used

The following Python libraries were used throughout the project:

- **pandas**: For data manipulation.
- **numpy**: For numerical operations.
- **matplotlib & seaborn**: For data visualization.
- **scikit-learn**: For model building and evaluation.

Data Preprocessing

Loading Data

Both train and test datasets were loaded.

Info Of data

Using pandas function info() and describe() to get knowlegde of train and test data

Handling Missing Values

Calculates missing value in each column of train and test using pandas isna() function.
Numerical columns: Filled with median values.
Categorical columns: Filled with the most frequent values.

Mutual Information

Calculated to understand the relationship between features and the target (house price). Using scikit learn mutual info regression function to get mutual information between features and target.Features with low mutual information were dropped.

Correlation Matrix

A correlation matrix was created to identify highly correlated features, which were then removed to avoid multicollinearity. Using seaborn lib to plot the heatmap.

Encoding

To convert categorical features into numerical ,i used one hot encoding function of scikit learn Library.

Scaling

Standardscalar function of scikit learn is used to scale the numerical features.

Model Selection

Several machine learning models were trained and evaluated:

- **RandomForestRegressor**
- **GradientBoostingRegressor**
- **AdaBoostRegressor**
- **LinearRegression**
- **Ridge**
- **Lasso**
- **ElasticNet**
- **SGDRegressor**
- **DecisionTreeRegressor**
- **KNeighborsRegressor**
- **RadiusNeighborsRegressor**
- **ExtraTreeRegressor**
- **XGBRegressor**
- **SVR**
- **MLPRegressor**
- **GaussianProcessRegressor**
- **LGBMRegressor**
- **CatBoostRegressor**

Evaluation Metrics

Different metrics were used to evaluate the performance of the models:

- **Mean Squared Error (MSE)**
- **Mean Absolute Error (MAE)**
- **R2 Score**

Best Performing Model

Out of all the models, **Gradient Boosting Regressor** performed the best.

Hyperparameter Tuning

I performed hyperparameter tuning on the Gradient Boosting model using **GridSearchCV** to find the optimal combination of parameters, which further improved the model’s accuracy.

Conclusion

Through a systematic approach of data preprocessing, feature engineering, model selection, and hyperparameter tuning, I was able to develop a highly accurate model for predicting house prices.

Link

https://github.com/rajiv3011/Machine_learning/blob/main/Kaggle/House_Pred.ipynb