Project Report

21/11/2023

Course: Artificial Intelligence and Machine Learning

Faculty: Dr. Sitara

Title: House Value Prediction Using Machine Learning

Members:

- Rohith Kumar V (106121107)
- Anshul Prabhakar (106121015)
- B Anantha krishnan (106121025)
- MD Shahbaz Hussain (106121075)
- Devadev Sujith (106121035)

Introduction

The real estate market is a complex system influenced by various factors such as location, size, amenities, and economic conditions. Predicting house values accurately is essential for both buyers and sellers. In this project, we aim to create a machine learning model for predicting house values based on the California housing dataset from sci-kit learn.

Literature Survey

Several studies have been conducted on house value prediction using machine learning techniques. Notable research papers include:

1. Advanced Machine Learning Algorithms for House Price Prediction: Case Study in Kuala Lumpur

This paper evaluates the performance of various machine learning algorithms for house price prediction, providing insights into the strengths and weaknesses of different models like *multiple regression analysis, ridge regression, lightGBM,* and *XGBoost*.

2. Machine Learning based Predicting House Prices using Regression Techniques

This research tackles the challenging task of predicting house sale prices in cities like Bengaluru, considering factors such as property size, location, and amenities. Utilizing a dataset from a machine hackathon platform, the study employs multiple regression techniques, including multiple linear regression, Lasso, Ridge regression, support vector regression, and boosting algorithms such as XG Boost. The objective is to construct a predictive model that accurately evaluates house prices based on these influential factors. Through a comparative analysis of predictive errors, the research aims to identify the most effective model for precise price estimation. This work contributes to the advancement of predictive modeling in the dynamic real estate landscape of urban centers like Bengaluru.

3. House Price Prediction using Random Forest Machine Learning Technique

This paper focuses on the pragmatic approach of predicting price variances, treating price prediction as a classification issue. While the House Price Index (HPI) is a common tool for assessing house price inconsistencies, it lacks precision for individual house predictions due to its aggregate nature. The research employs the Random Forest machine learning technique to enhance house price prediction accuracy. Utilizing the UCI Machine Learning repository's Boston housing dataset, the model demonstrates acceptable predicted values with an error margin of ±5 when compared to actual prices, showcasing its effectiveness in capturing price variations.

Design and Implementation

We use the California housing dataset from sci-kit learn, which contains features like median income, housing median age, average rooms, etc.

Machine Learning Algorithms used

1. **Linear Regression:** A simple and interpretable model that assumes a linear relationship between features and house prices.
2. **Decision Tree:** A non-linear model that can capture complex relationships in the data.
3. **Random Forest:** An ensemble model that combines multiple decision trees for improved prediction accuracy.

The dataset is split into training and testing sets. Each algorithm is trained on the training set, and hyperparameters are tuned using cross-validation.

The values for evaluation metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), R2 score are calculated for all the three algorithms mentioned above and their values are compared in all the three cases to decide the most efficient algorithm out of all three.

Evaluation Metrics

Evaluation metrics are quantitative measures used to assess the performance and effectiveness of a statistical or machine learning model. These metrics provide insights into how well the model is performing and help in comparing different models or algorithms. Here, we have used:

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

Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is a metric used in machine learning to measure the average magnitude of errors between predicted and actual values. Mathematically, the Mean Absolute Error is calculated by taking the average of the absolute differences between the predicted values (represented as ^y) and the actual values (represented as y) in a dataset. It can be expressed as:

- MAE = $(1/n) * \Sigma|\hat{}y - y|$

It is not suitable for large errors penalizes large errors. It also ignores outliers making it more suitable for regression analysis.

Mean Squared Error (MSE)

The Mean Squared Error (MSE) or Mean Squared Deviation (MSD) of an estimator measures the average of error squares i.e. the average squared difference between the estimated values and true value. Mathematically, the Mean Absolute Error is calculated by taking the average of the square of the absolute differences between the predicted values (represented as ^y) and the actual values (represented as y) in a dataset. It can be expressed as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2$$

It is suitable for higher values as it doesn't penalize large errors.

R2 Score

R2, the coefficient of determination or coefficient of multiple determination measures a model's predictive accuracy. Derived from total (SST) and residual (SSR) sum of squares.

- SST = Σ(yi – ȳ)^2
- SSR = Σ(yi – ^y)^2

The R2 score is a normalized measure, ranging from 0 to 1, with higher values indicating better model fit:

- R2 = 1 - (SSR / SST).

But in some applications like psychology, even models with low R2 scores are considered to be good models. If we have many statistically significant predictors, then we can draw conclusions from the modal even if the R2 score is low.

Models

Linear Regression

Linear regression is a supervised learning algorithm used for predicting a continuous outcome variable (also called the dependent variable) based on one or more predictor variables (independent variables). The primary goal of linear regression is to find the linear relationship between the input variables and the target variable. The linear relationship is represented by a linear equation:

- y= mx + c

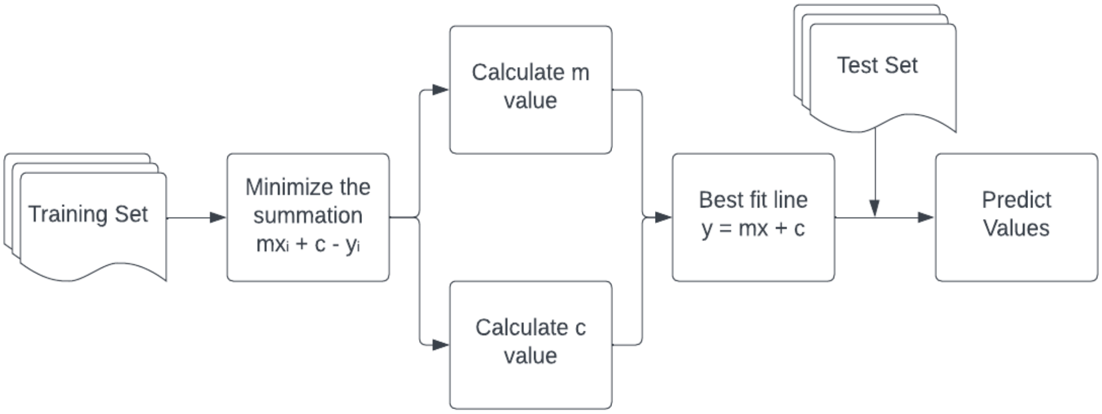
y is the dependent variable (the variable we are trying to predict).

x is the independent variable (the input feature).

m is the slope of the line, representing the change in y for a one-unit change in x.

c is the y-intercept, representing the value of y when x is 0.

Architectural Diagram

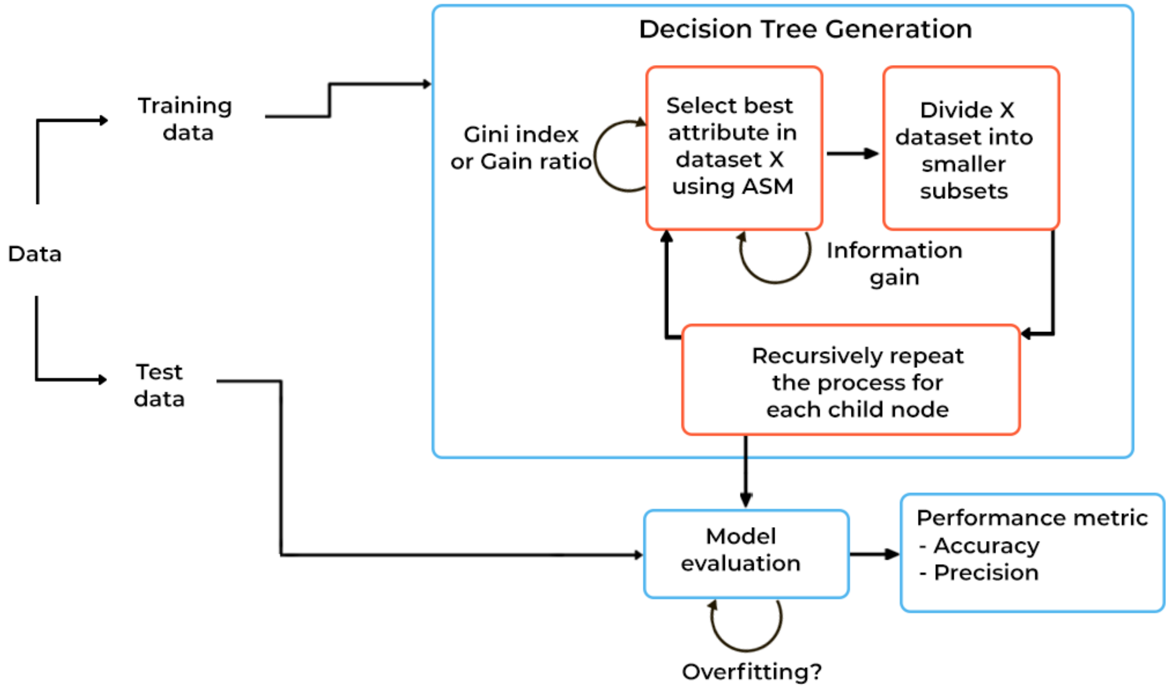


Decision Tree

A decision tree is a predictive algorithm used in machine learning for both classification and regression tasks. It's a visual representation resembling a tree, with nodes representing decisions or tests based on input features. Branches emanate from nodes, leading to possible outcomes, and leaf nodes signify final predictions. The tree is constructed by recursively selecting the most informative features to split the data, optimizing for decision-making. This hierarchical structure provides a transparent and interpretable way to understand and analyze the decision-making process, making it valuable in various applications such as finance, healthcare, and marketing.

In decision trees, entropy measures dataset disorder, and information gain quantifies the effectiveness of a feature for splitting. High entropy indicates label uncertainty. Information gain is the difference between parent node entropy and the weighted average of child node entropies after a split. Decision trees use features with maximum information gain to split nodes, reducing uncertainty. This iterative process builds a tree for making decisions based on input features, enhancing predictive accuracy by minimizing disorder in the data.

Architectural Diagram



Random Forest

Random Forest is an ensemble learning algorithm used for both classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or the mean prediction (regression) of the individual trees.

Here's a brief description of how Random Forest works:

Decision Trees: Simple models making decisions based on a series of questions, with each node representing a decision.

Ensemble Learning: Collection of decision trees, with "random" from random data and feature selection to enhance robustness.

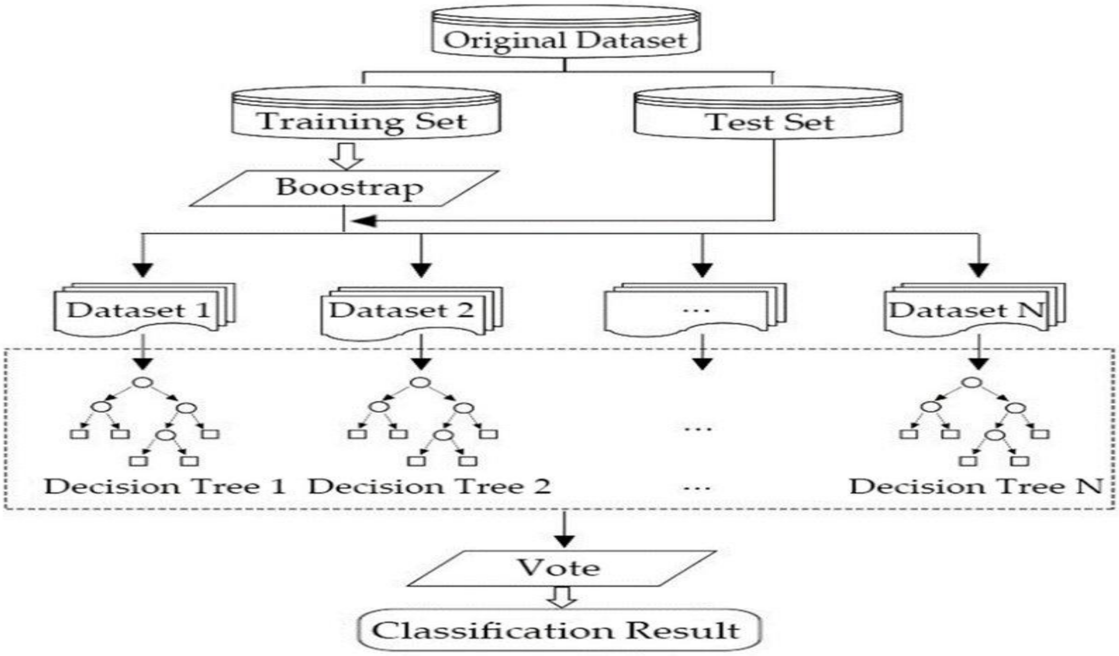
Bootstrapped Sampling: Technique using random subsets of the training data with replacement for training individual trees.

Random Feature Selection: At each tree node, a random subset of features is considered for making splits, reducing correlation between trees.

Voting or Averaging: For classification, trees vote for a class; for regression, outputs are averaged for the final prediction.

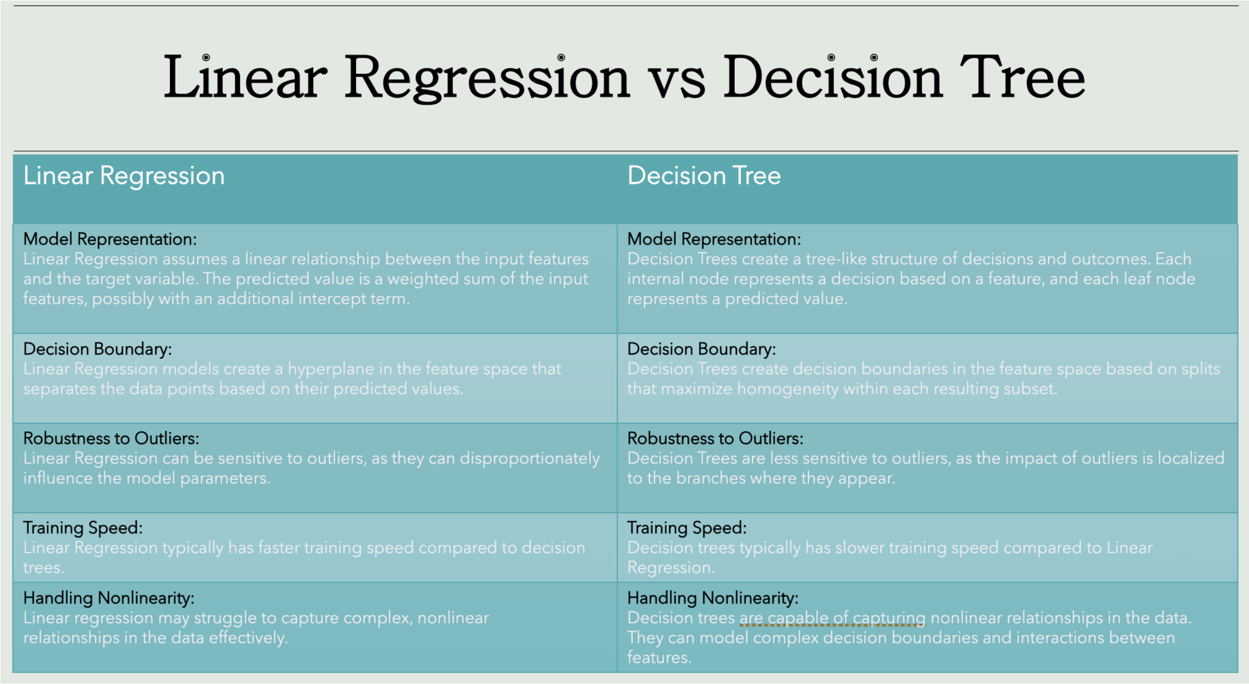
Robustness and Generalization: Combining diverse trees enhances robustness, enabling better handling of noisy data and avoiding overfitting.

Architectural Diagram



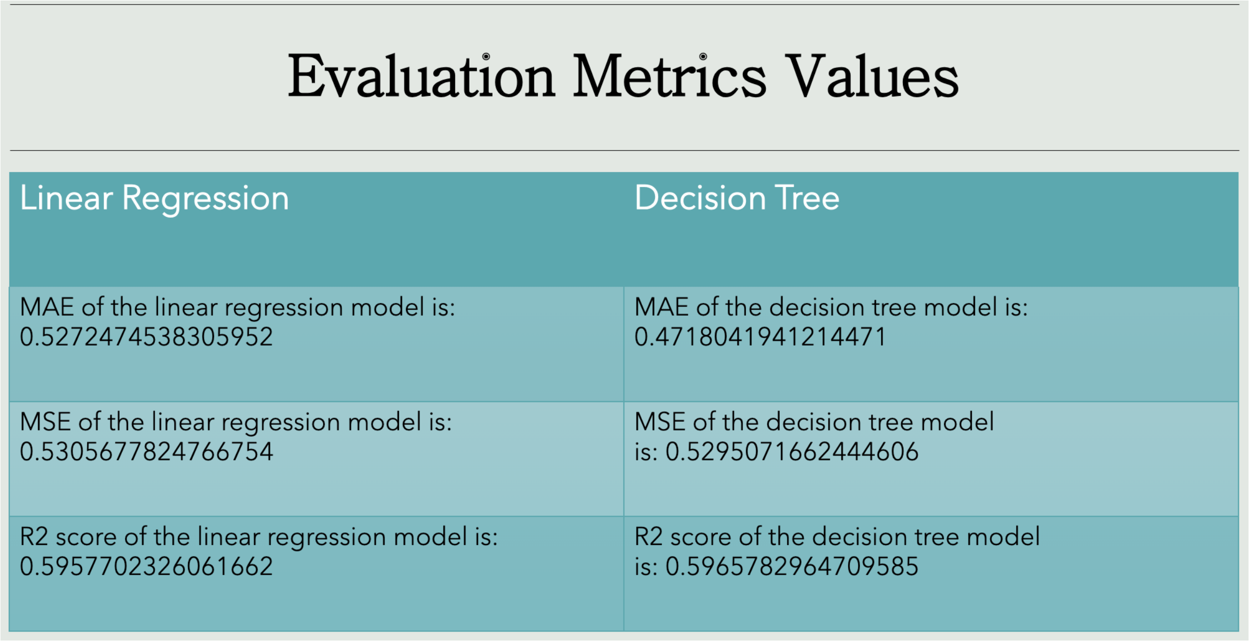
Comparison

We compare the performance of the three models using metrics such as Mean Squared Error (MSE), R-squared, and Mean Absolute Error (MAE). These metrics provide insights into the accuracy and precision of each model.



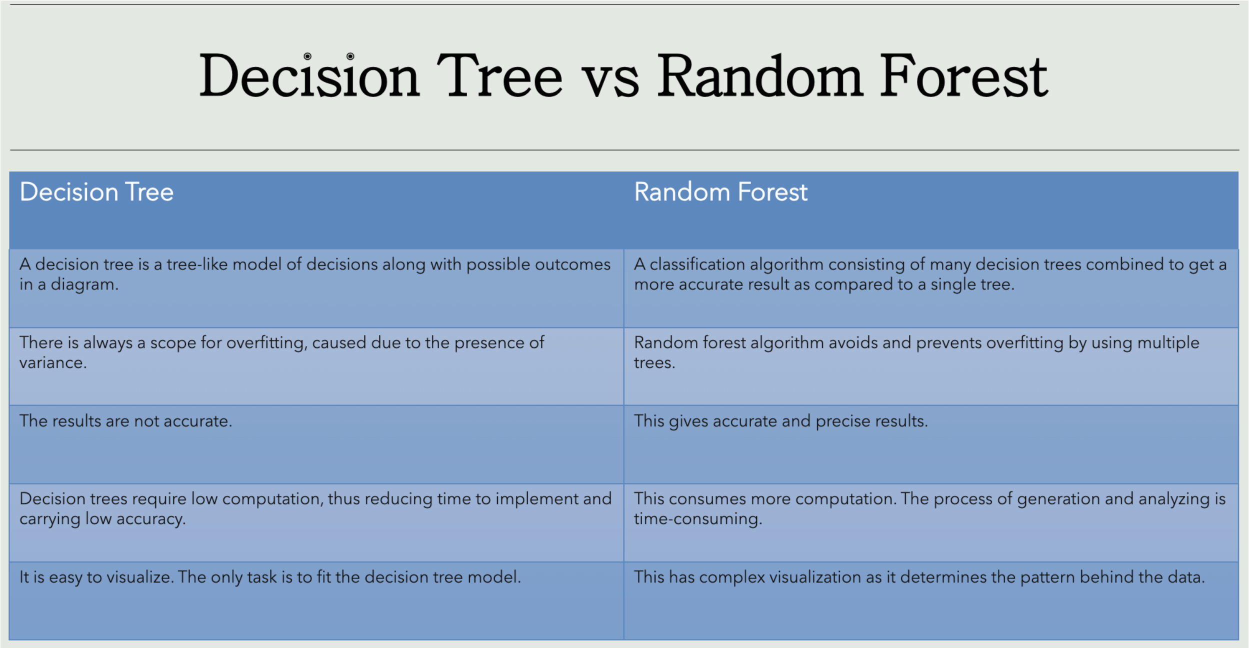
Linear Regression vs Decision Tree

Linear Regression	Decision Tree
Model Representation: Linear Regression assumes a linear relationship between the input features and the target variable. The predicted value is a weighted sum of the input features, possibly with an additional intercept term.	Model Representation: Decision Trees create a tree-like structure of decisions and outcomes. Each internal node represents a decision based on a feature, and each leaf node represents a predicted value.
Decision Boundary: Linear Regression models create a hyperplane in the feature space that separates the data points based on their predicted values.	Decision Boundary: Decision Trees create decision boundaries in the feature space based on splits that maximize homogeneity within each resulting subset.
Robustness to Outliers: Linear Regression can be sensitive to outliers, as they can disproportionately influence the model parameters.	Robustness to Outliers: Decision Trees are less sensitive to outliers, as the impact of outliers is localized to the branches where they appear.
Training Speed: Linear Regression typically has faster training speed compared to decision trees.	Training Speed: Decision trees typically has slower training speed compared to Linear Regression.
Handling Nonlinearity: Linear regression may struggle to capture complex, nonlinear relationships in the data effectively.	Handling Nonlinearity: Decision trees are capable of capturing nonlinear relationships in the data. They can model complex decision boundaries and interactions between features.



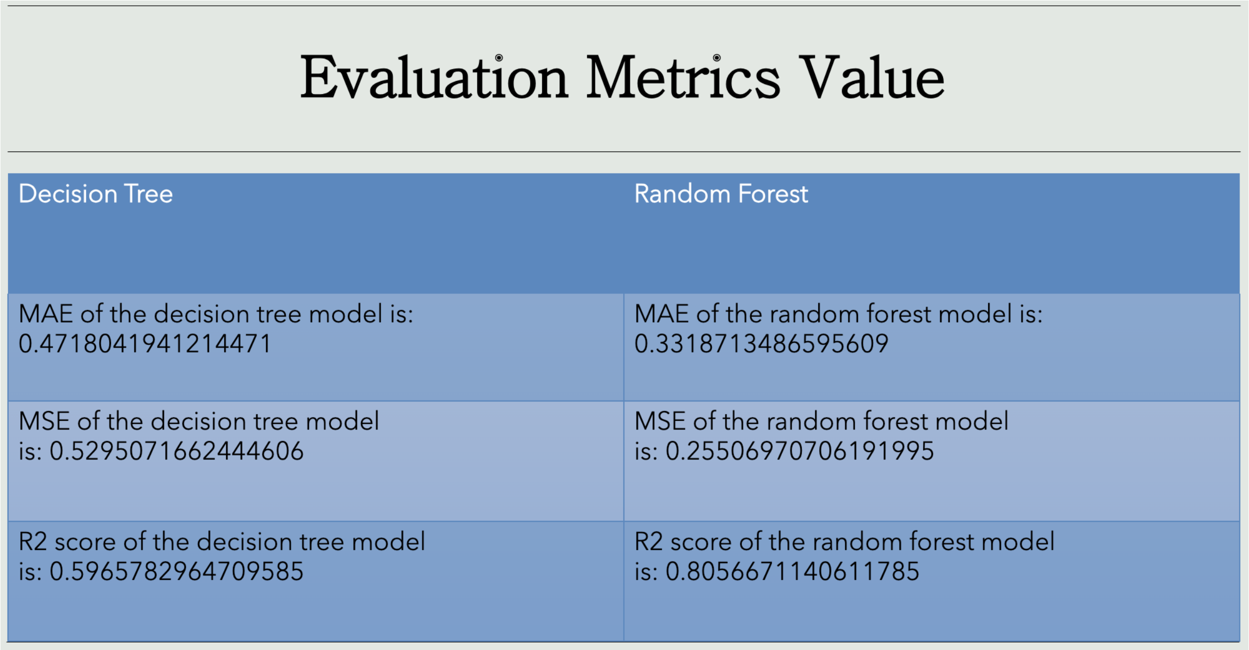
Evaluation Metrics Values

Linear Regression	Decision Tree
MAE of the linear regression model is: 0.5272474538305952	MAE of the decision tree model is: 0.4718041941214471
MSE of the linear regression model is: 0.5305677824766754	MSE of the decision tree model is: 0.5295071662444606
R2 score of the linear regression model is: 0.5957702326061662	R2 score of the decision tree model is: 0.5965782964709585



Decision Tree vs Random Forest

Decision Tree	Random Forest
A decision tree is a tree-like model of decisions along with possible outcomes in a diagram.	A classification algorithm consisting of many decision trees combined to get a more accurate result as compared to a single tree.
There is always a scope for overfitting, caused due to the presence of variance.	Random forest algorithm avoids and prevents overfitting by using multiple trees.
The results are not accurate.	This gives accurate and precise results.
Decision trees require low computation, thus reducing time to implement and carrying low accuracy.	This consumes more computation. The process of generation and analyzing is time-consuming.
It is easy to visualize. The only task is to fit the decision tree model.	This has complex visualization as it determines the pattern behind the data.



Evaluation Metrics Value

Decision Tree	Random Forest
MAE of the decision tree model is: 0.4718041941214471	MAE of the random forest model is: 0.3318713486595609
MSE of the decision tree model is: 0.5295071662444606	MSE of the random forest model is: 0.25506970706191995
R2 score of the decision tree model is: 0.5965782964709585	R2 score of the random forest model is: 0.8056671140611785

Results and Discussion

The results indicate that the random forest model outperforms linear regression and decision tree models in terms of prediction accuracy. Random forest's ability to handle non-linear relationships and reduce overfitting contributes to its superior performance.

Conclusion

In conclusion, our study demonstrates the effectiveness of machine learning in predicting house values. Among the models considered, the random forest algorithm stands out as the most promising for accurate and reliable predictions.By employing linear regression, decision tree, and random forest models on the California housing dataset, we obtained a foundation for predictive analytics. The visualization of spatial relationships and correlation analysis enhanced our understanding of feature interactions. Additionally, the project's success hinges on considerations like interpretability, hyperparameter tuning, and user-friendly interfaces.

Future Works

- Feature Engineering:

  Explore additional features or transformations of existing features that might better capture the underlying patterns in the data.Consider creating new features based on domain knowledge or by combining existing features.

- Cross-Validation:

Implement cross-validation to get a more robust estimate of the model's performance. This helps ensure that the model's performance is consistent across different subsets of the data.

- Advanced Regression Techniques:

  Explore more advanced regression techniques like Support Vector Regression, Gradient Boosting Regression, or Neural Networks, depending on the dataset size and complexity.

- Time-Series Analysis:

  If the dataset has a temporal component, consider incorporating time-series analysis techniques to capture trends and patterns over time.

References

Data set: Housing data

Research Papers:

[1] Advanced Machine Learning Algorithms for House Price Prediction: Case Study in Kuala Lumpur

[2] Machine Learning based Predicting House Prices using Regression Techniques

[3] House Price Prediction using Random Forest Machine Learning Technique