From Branches To Forests : A Comparative Analysis of Decision Tree & Random Forest Models for Used Car Price Prediction

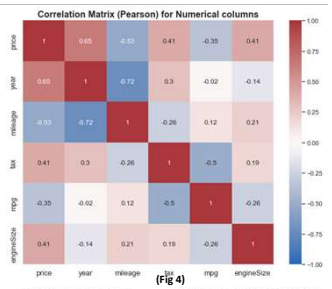
Name : **Fatema Fidvi**

Description and motivation

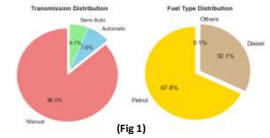
The poster delves into comparing the performance of two supervised algorithms namely Random Forest and Decision Tree for the Regression Problem.
The goal is to compare results to those obtained by N. Pal *et al.* [1] using a similar dataset for Random Forest and then apply Decision Tree algorithm and use performance metrics to assess both models' performances.
The intention is to accurately predict the prices of used cars and compare the efficiency of these two algorithms in doing so.

Exploratory Analysis

- Dataset : Utilizing a subset of 100,000 UK used car data set from Kaggle, focusing specifically only on the Ford car dataset to perform better analysis and optimize computational efficiency.
- The dataset includes key attributes such as model, year, price, transmission, mileage, fuel type, tax, mpg (miles per gallon) and engine size. It originally comprised of 17,965 entries across 9 columns, where it displays diverse characteristics of pre-owned vehicles. From these, the 'price' is the target column while the others are the predictor columns.
- While the dataset exhibited no missing values, it did contain duplicates and a noticeable outlier which were promptly addressed following the conduction of visual exploration of the dataset and examining the distributions of all variables and their correlations.
- Figure 1 displays the distribution of categorical columns, 'transmission' and 'Fuel type'.
- Various label encoding techniques were applied as part of feature engineering, including ordinal encoding for the 'year' column, integer encoding based on frequency for the 'model' column, and one-hot encoding for 'fuelType' and 'engineSize' columns.
- Figure 2 provides a comprehensive statistical summary of the numerical columns, illustrating key metrics such as minimum, maximum, mean, and standard deviation for each variable.
- Figure 3 reveals that the numerical columns lack a normal distribution, displaying various degrees of skewness and uneven patterns. While scaling these columns was a potential option, it was opted to perform predictive modeling using unscaled data as Random Forest and Decision Tree algorithms are both robust to unscaled data. But, for the purpose of checking the precision of the results, scaling is done using MinMax method at the end to compare prediction results on scaled and unscaled data for both the algorithms.
- A Pearson Correlation matrix (fig(4)) depicts a strong correlation between price column with that of the year column. This correlation stems from the common observation that newer cars tend to be less used and fetch higher prices. Additionally, a significant negative correlation exists between 'price' and 'mileage,' indicating that higher mileage contributes to lower prices. Other variables also exhibit notable correlations with the target column.



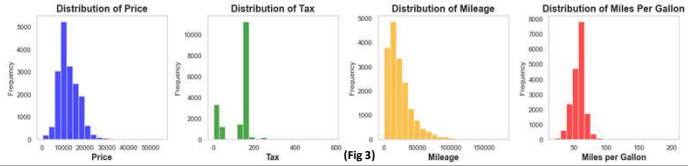
(Fig 4)



(Fig 1)

	year	price	Mileage	tax	Mpg	Engine Size
mean	2016.86	12270.10	23380.41	113.31	57.91	1.35
std	2.02	4736.26	19418.18	62.03	10.13	0.43
min	1996.00	495.00	1.00	0.00	20.80	0.00
max	2020.00	54995.00	177644.00	580.00	201.80	5.00

(Fig 2)



(Fig 3)

Decision Tree

i. Decision Tree algorithm is a supervised machine learning method used for both classification as well as regression tasks.
ii. "The result, such as a class label for classification or a numerical value for regression, is represented by each leaf node in the tree-like structure that is constructed, with each internal node representing a judgment or test on a feature." [2]
iii. Decision Tree is used to make a non-linear model quickly and to interpret how the model is making the decisions. [3]

Advantages

- Easy to understand, interpret and visualize.
- Robust to scaling or normalization.
- Does not need a lot of data preprocessing.
- Can deal with categorical features and does not necessarily need label encoding. [6]
- Particularly suitable for large datasets because it splits the data into small packages, enhancing efficiency and accuracy. [5]
- It is quick to build.

Disadvantages

- Highly sensitive to small changes in the data.
- Sensitive to outliers.
- Highly prone to overfitting.
- Exhibits a stepwise predictive nature, introducing a distinctive, non-smooth pattern in the predictions for regression. (as proved in model below)
- In classification, it is biased towards dominant class. [6]
- A small error in the top leaves can lead to severe impacts on the final model. [5]

Random Forest

i. Random Forest is a supervised ensemble method which combines the output of multiple decision trees to reach a single result.
ii. These trees are trained independently on random subsets of the data. The final prediction is then determined by taking the mean (in case of regression) or mode (in case of classification) of the predictions of individual trees, providing improved accuracy, robustness, and generalization compared to a single Decision Tree. [3]
iii. Random Forest is used if we have plenty of computational ability and we want to build a model that is likely to be highly accurate without worrying about how to interpret the model. [3]

Advantages

- Robust to outliers or noise in the data.
- Robust to scaling or normalization of the features.
- Does not need a lot of data preprocessing.
- Shows high accuracy even on complex data. [6]
- Although it is a complex algorithm, it is fast and can handle large datasets.
- Random Forest provides a measure of feature importance, which can help in feature selection and data understanding. [4]

Disadvantages

- Even though Random Forest is not as prone to overfitting as decision tree algorithm, there is a slight potential of overfitting that needs to be considered. [6]
- It is less interpretable than decision tree.
- The time taken by random forest to train a model is higher when compared to other algorithms .
- Random Forest requires more memory than other algorithms because it stores multiple trees. This can be a problem if the dataset is large.[4]

Hypothesis Statement

- Random Forest and Decision Tree algorithms are both good for regression, but it is anticipated to get a moderately higher R-squared value from the Random Forest model, as decision tree is more prone to overfitting and does not perform as well for complex data as the one being used.
- Drawing inspiration from C. Jin's [5] research, where the R-square values for Random Forest and Decision Tree models were found to be 0.904 and 0.851 respectively, these figures are anticipated to serve as baseline benchmarks in this study, acknowledging the slight difference in the datasets used.
- Additionally, it is assumed that the results will be comparable for scaled and unscaled data, considering the robustness of Random Forest and Decision Tree models to unscaled input.

Choice of Hyperparameters and Experimental Results

I. Random Forest

Choice of Hyperparameters :

Following Grid Search, the two best hyperparameters selected for training this model were Max Number of Trees = 60 and Max Number of Splits = 150. The number of trees determines the model's complexity and ability to generalize, while the number of splits controls the depth of individual trees, influencing their decision-making process.

Experimental Results :

- Upon training the Random Forest model using 'TreeBagger' method with the selected hyperparameters, the model exhibited robust performance on the training set, achieving a high R-squared value of 0.9215 indicating that 92.15% of the variance in the predicted values is explained by the model. However, the Mean Squared Error (MSE) value of 1751473.18 suggests some degree of error between predicted and actual prices.
- Subsequently, feature importance analysis was conducted using the Out-of-Bag (OOB) Predictor Importance feature of Tree Bagging. After comparing the model performance before and after excluding non-important features, the results demonstrated negligible differences. Hence, retaining the original model without removing any features was preferred.
-A comparison was also conducted to check the robust nature of the model on scaled and unscaled data and the results show very minimum variation as assumed in the hypothesis.

Training set	Random Forest		Decision Tree	
	with unscaled data	with scaled data	with unscaled data	with scaled data
R2	0.9215	0.9224	0.89414	0.89657
(MSE)	1751473.18	1869048.12	2362771.18	2311829.18

Table showing results on scaled and unscaled data

II. Decision Tree

Choice of Hyperparameters :

Following Grid Search, the two best hyperparameters selected for training this model were Min Leaf Size = 20 and Max Number of Splits = 100. The "Min Leaf Size" influences model complexity by setting the minimum observations needed for a leaf node, impacting overfitting. Meanwhile, "Max Number of Splits" controls the tree's depth, affecting its decision-making process. These hyperparameters were chosen to balance complexity and generalization.[7]

Experimental Results :

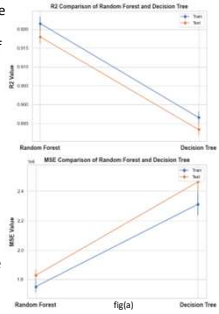
- Upon training the Decision Tree model using 'fitrtree' method with the selected hyperparameters, the model performed well on training set, achieving R-squared value of 0.8965 indicating that 89.65% of the variance in the predicted values is explained by the model. However, the MSE value came out to be 2311829.18 which is higher than expected and shows some extent of error between the predicted and actual prices.
- Even for the Decision Tree model, assessment was done to check its robustness to data scaling by comparing performance between scaled and unscaled data, with results presented in the table.

Description of the choice of Training and Evaluation Methodology

i. Following the steps of N. Pal *et al.* [1], the pre-processed data was split into training, testing and validation sets with a 70:20:10 split ratio.
ii. Subsequently, a Grid Search is conducted on validation set to optimize the hyperparameters for the random forest and Decision Tree models determined by evaluating the model performance using R-squared and MSE values.
iii. Evaluated the results on training sets using training and validation metrics after getting the best hyperparameters.
iv. Performed K-fold cross validation for evaluating the generalization ability of both the models. This approach provided a more comprehensive understanding of how well the model can perform on unseen data, reducing the risk of overfitting and providing a more accurate estimate of its overall predictive capability.
v. Implemented measures to prevent data leakage by maintaining a strict separation between training , validation and testing sets throughout the entire modeling process.

Analysis and Critical Evaluation of Results

- The R-squared values (R2) for training the random forest and decision tree models on the training set are 0.92153 and 0.89414 respectively. The better performance of random forest model is owing to its ensemble of trees that contributes to its performance boost. Notably, the decision tree model outperforms the baseline model as established by C. Jin [5] by approximately 4%. The increased number of splits in the decision tree model compared to C. Jin's allows it to capture more intricate correlations, enhancing performance on the training set. To avoid overfitting risks, the number of splits are capped at a maximum of 100, ensuring generalization to the test set without compromising the results.
- On the training set, random forest achieved an MSE of 1751473.18, outperforming the decision tree with an MSE of 2362771.18. The lower MSE of the random forest indicates superior model performance, suggesting that its predictions closely align with the actual values, emphasizing its effectiveness over the decision tree.
- During the evaluation on the test set, both models exhibited outstanding performance. The random forest model achieved an R2 value of 0.91799 with an MSE of 1828169.73, while the decision tree model yielded an R2 of 0.8805 with an MSE of 2495465.56. These results, closely aligned with the training set performance, thus proving that the models perform really well on seen as well as unseen data. The performance of both models is visually represented using two line plots with error bars in fig(a). Overall, the figure shows that Random Forest outperforms Decision Tree in terms of both R2 values and MSE across both training and test datasets.



fig(a)

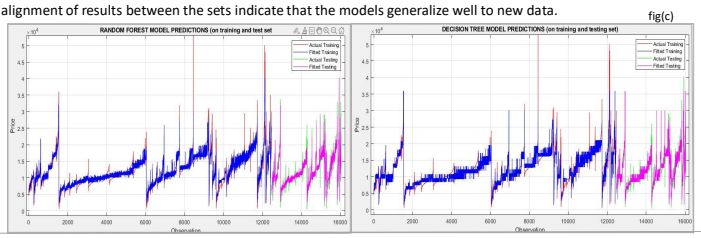
- After evaluating the models on the test set, K-fold cross-validation is conducted on the combined dataset on both the models, presenting the results in table (fig(b)). This approach allowed assessment of both the models' performance across diverse subsets of the data. The outcome provided a more reliable estimate of their overall effectiveness.

Model	Avg MSE	Avg R2
Random Forest	1950604.03	0.91594
Decision Tree	2521596.39	0.88777

K-Fold Cross Validation results fig(b)

- Visualizing the predicted values alongside the actual values for both the training and test data revealed distinct characteristics, as depicted in fig(c). While the random forest model exhibits smooth predictions without noticeable stepwise patterns, the decision tree model shows a more stepwise and less smooth nature. While both models accurately capture the range of actual values, the random forest model's predictions are closer to the actual values, attributed to its lower MSE compared to the decision tree. Despite the decision tree model performing better than anticipated, it lacks the smoothness observed in the random forest model.
- The consistent performance of both models on both the training and test sets justifies their robustness. The close alignment of results between the sets indicate that the models generalize well to new data.

fig(c)



Lessons Learned

- Managing the random seed (rng) for reproducibility is generally beneficial; however, it is observed that setting the seed within a specific range occasionally led to unexpected behavior in the Decision Tree model, resulting in anomalous outcomes. Therefore, the model was allowed to run without a fixed seed which enhanced stability and prevented erratic behavior.
- Outliers could've been dealt with for producing better and more accurate predictive values.

Future Work

- In grid search, considering additional hyperparameters like 'CategoricalPredictors' and 'NumPredictorsToSample' for Random Forest, and 'PredictorSelection' and 'Prune' for Decision Tree, could further refine and optimize the predictive models, potentially enhancing the accuracy and robustness of the models.
- Despite the system's success in the automotive price prediction challenge, it is desired to see how it performs on other datasets. OLX and eBay's used car data sets will improve the test data and validate the methods as suggested by M. Kathiravan *et al.* [8]
- In future, exploring the impact of scaling the data on prediction outcomes using various models, such as logistic regression or Naïve Bayes could be valuable to get more insights.
- It is possible to reach a better estimation rate with a data set with more units and different variables as suggested by O. Celik and O. Osmanoglu [9].

References :

[1] N. Pal, P. Arora, P. Kohli, D. Sundararaman, and S. S. Palakurthy, "How much is my car worth? A methodology for predicting used cars' prices using Random Forest," in *Advances in intelligent systems and computing*, 2018, pp. 413–422. doi: 10.1007/978-3-030-03402-3_28.
[2] GeeksforGeeks, "Decision Tree algorithms," *GeeksforGeeks*, Nov. 11, 2023. https://www.geeksforgeeks.org/decision-tree-algorithms/
[3] Zach, "Decision Tree vs. Random Forests: What's the Difference?," *Statology*, Aug. 09, 2021. https://www.statology.org/decision-tree-vs-random-forest/
[4] A. Fleiss, "What are the advantages and disadvantages of random forest?," *Rebellion Research*, Feb. 19, 2023. https://www.rebellionresearch.com/what-are-the-advantages-and-disadvantages-of-random-forest
[5] C. Jin, "Price Prediction of Used Cars Using Machine Learning," 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT), Chongqing, China, 2021, pp. 223-230, doi: 10.1109/ICESIT53460.2021.9696839.
[6] A. Jehad et al., "Random Forests and Decision Trees," International Journal of Computer Science Issues (IJCSI), vol. 9, 2012.
[7] "Determine the amount of splits in a decision tree of sklearn," *Stack Overflow* https://stackoverflow.com/questions/49672484/determine-the-amount-of-splits-in-a-decision-tree-of-sklearn
[8] M. Kathiravan, M. Ramya, S. Jayanthi, V. V. Reddy, L. Ponguru and N. Bharathiraja, "Predicting the Sale Price of Pre-Owned Vehicles with the Ensemble ML Model," 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2023, pp. 1793-1797, doi: 10.1109/ICESC57686.2023.10192988.
[9] O. Celik and O. Osmanoglu, "Prediction of the Prices of Second-Hand Cars," *European Journal of Science and Technology*, vol. 16, pp. 77-83, 2019.