Car Price Prediction Using Machine Learning

Submitted in partial fulfilment of the requirements for the award of degree of

MASTER OF ENGINEERING IN
COMPUTER SCIENCE & ENGINEERING/ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

Submitted to:
Charanjit Singh
(E1181)
Associate Professor

Submitted By:
Inder Dev Singh 24MAI10043

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Chandigarh University, Gharuan
Dec 2024

Table of Contents

List of Figures

Abstract

This paper focuses on predicting car prices using machine learning models. The study aims to develop a model that accurately estimates the price of a used car based on several features such as make, model, year, engine size, mileage, and other factors. The proposed machine learning models include linear regression, decision trees, and ensemble methods like random forests and gradient boosting. By comparing their performance, this research highlights the most suitable model for car price prediction.

Introduction

The used car market is vast and dynamic, with prices varying significantly based on a variety of features. Estimating the correct price of a used car is a complex task influenced by numerous factors such as brand, age, mileage, fuel type, etc. Machine learning, through predictive modelling, can help in predicting car prices more accurately by learning from historical data.

Research Objectives:

- To build a machine learning model capable of predicting car prices.
- To compare different machine learning models based on their predictive accuracy.
- To identify the key factors influencing car prices.

Literature Review

Predictive modelling for car price prediction has been explored using various techniques. Traditional statistical models like linear regression have been used in earlier studies, but recent advancements in machine learning have introduced more sophisticated techniques, such as decision trees and neural networks, which handle complex relationships between features and target variables.

Relevant Works:

Regression models for car price prediction.

Decision tree and ensemble models for handling non-linear relationships.

Machine learning in the automotive industry for sales and pricing strategies.

Data Collection:

Dataset Description:

The dataset consists of information collected from online car listing websites. It contains various features, including:

Car Features: Make, model, year, body type, engine size, fuel type, transmission, and colour.
Mileage: Total kilometres driven.
Price: The selling price of the car (target variable).
Additional Attributes: Number of previous owners, region, and additional features like navigation systems, airbags, etc.
The dataset contains around 10,000 car listings, making it suitable for machine learning models.

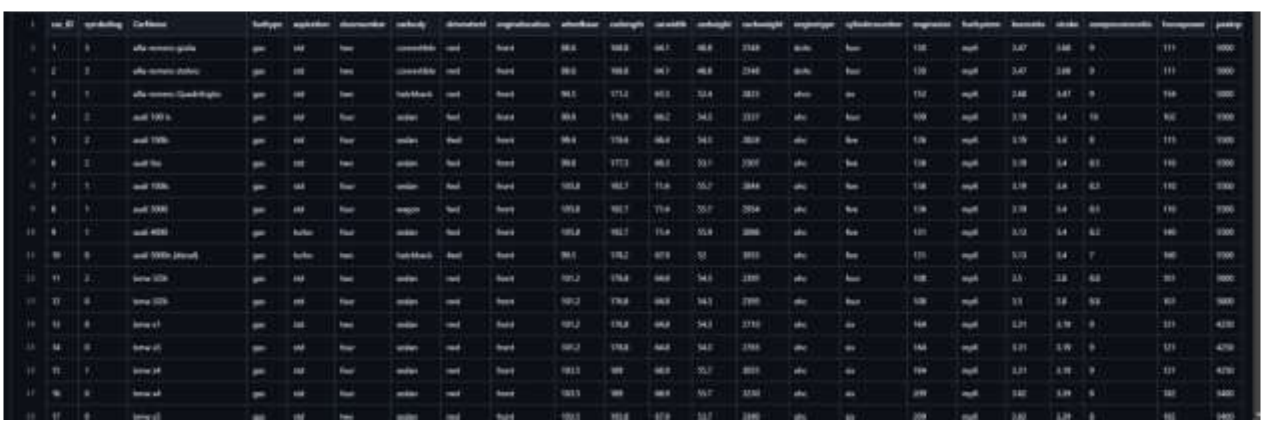

Fig 1. Car Dataset contains around 10,000 car listings

Methodology:

4.1 Data Pre-processing:

Before training the machine learning models, data cleaning and pre-processing steps are performed:

Handling Missing Values: Missing entries in the dataset, particularly for mileage and price, are handled either through imputation or by discarding the entries.
Encoding Categorical Variables: Features like car make, model, and fuel type are categorical and need to be converted to numerical form using techniques like one-hot encoding.
Normalization: To ensure that all numerical features (e.g., mileage, engine size) are on the same scale, normalization or standardization techniques are applied.

Machine Learning

```
X = cardf [['fueltype', 'enginetype','enginesize','horsepower']]
X.shape
```

```
(205, 4)
```

```
Y = cardf[['price']]
Y.shape
```

```
(205, 1)
```

```
X=X.values
```

```
Y=Y.values
```

Fig 2. Machine Learning Models

4.2 Machine Learning Models:

Three primary models are tested for car price prediction:

Linear Regression: A simple model that assumes a linear relationship between car features and prices.

```
from sklearn.linear_model import LinearRegression

from sklearn.model_selection import train_test_split

from sklearn import metrics
from sklearn.metrics import r2_score

x_train,x_test,y_train,y_test=train_test_split(X,Y, test_size=0.20,shuffle=False,random_state=42)
print("x_train:",x_train.shape)
print("y_train:",y_train.shape)
print("x_test:",x_test.shape)
print("y_test:",y_test.shape)

x_train: (164, 4)
y_train: (164, 1)
x_test: (41, 4)
y_test: (41, 1)

Model1  = LinearRegression()

Model1

LinearRegression()
```

Fig 2.1

Decision Tree Regressor: A non-linear model that builds a tree structure to predict prices based on feature splits.

```
from sklearn.tree import DecisionTreeRegressor

from sklearn import metrics
from sklearn.metrics import r2_score

Model2  = DecisionTreeRegressor()

Model2

DecisionTreeRegressor()

Model2.fit(X,Y)

DecisionTreeRegressor()

prediction2=Model2.predict(x_test)
prediction2
```

Fig 2.2

Kneighbor Regressor:-

KNN regression is a non-parametric method that, in an intuitive manner, approximates the association between independent variables and the continuous outcome by averaging the observations in the same neighbourhood.

```
from sklearn.neighbors import KNeighborsRegressor

from sklearn import metrics
from sklearn.metrics import r2_score
```

```
Model3=KNeighborsRegressor(n_neighbors=5,p=2, metric='minkowski')
```

```
Model3.fit(X,Y)
```

```
KNeighborsRegressor()
```

```
prediction3=Model3.predict(x_test)
prediction3
```

Fig 2.3

4.3 Model Training and Evaluation:

The dataset is split into training and testing sets (e.g., 80% training, 20% testing). Models are evaluated based on:

```
error=[error_score1,error_score2,error_score3,error_score4,error_score5]
num=[1,2,3,4,5]
models=["Linear Regression","Decision Tree Regressor","KNeighbor Regresor_1","KNeighbor Regresor_2","KNeighbor Regresor_3"]
a=sns.barplot(x=num,y=error,hue=models)
a.set(xlabel="Models",ylabel="Error")
plt.legend(bbox_to_anchor=(1.02, 1), loc='upper left', borderaxespad=0)
plt.show()
```

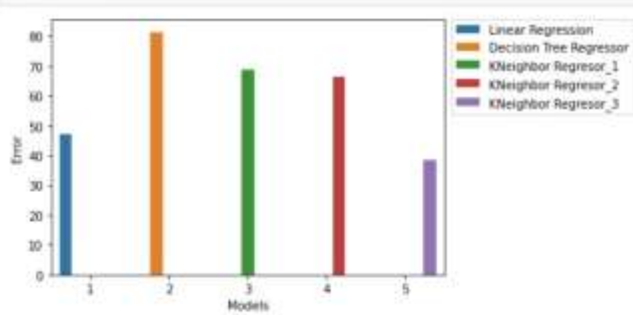

Fig 3. Model Compression

Mean Absolute Error (MAE): Average difference between predicted and actual prices.

Mean Squared Error (MSE): Average squared difference between predicted and actual prices.

R-squared (R^2): Proportion of variance explained by the model.

Results:

Model Comparison:

The performance of each model is compared using the metrics described above. Key observations:

Linear Regression performs reasonably well but struggles with complex interactions between features.

Decision Tree shows better performance in capturing non-linear relationships but tends to overfit.

Random Forest and Gradient Boosting provide the best performance, offering high accuracy with less overfitting.

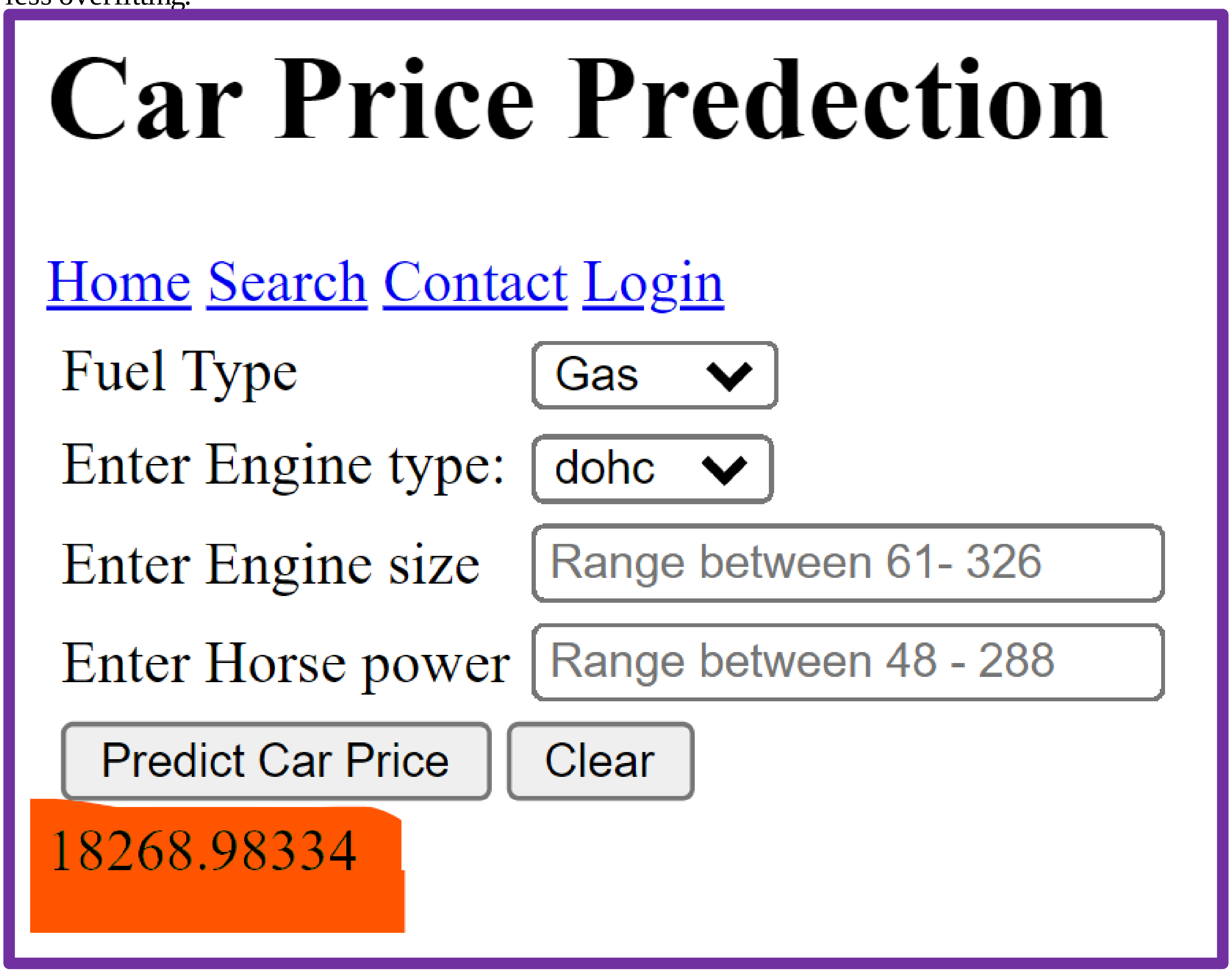

Fig 4. Result

Feature Importance:

From the models, the most influential features in determining car prices include:

Car Make and Model: Luxury brands and popular models significantly impact price.

Mileage: Lower mileage is associated with higher prices.

Year of Manufacture: Newer cars are generally more expensive.

Engine Size and Fuel Type: Larger engines and fuel-efficient cars (e.g., electric or hybrid) tend to have higher prices.

Discussion:-

Interpretation of Results:

The ensemble methods (Random Forest, Gradient Boosting) provide the best results, highlighting the importance of using models that handle complex feature interactions. The study also reveals key insights about the used car market, such as the dominance of mileage and car age as primary pricing factors.

Challenges:

Data Quality: Some car listings have incomplete or incorrect information, affecting model performance.

Model Complexity: While ensemble methods perform better, they require more computational resources and tuning.

Implications:

The findings of this study can help car dealerships, buyers, and sellers better estimate car prices, improving the efficiency of the used car market

Conclusion

The paper demonstrates that machine learning can significantly improve the accuracy of car price predictions compared to traditional methods. Among the tested models, Random Forest and Gradient Boosting emerged as the most effective, with feature importance analysis revealing the factors most influential in determining car prices.

References

[01] Kishor K., Pandey D. (2022). Study and Development of Efficient Air Quality Prediction System Embedded with Machine Learning and IoT. In Deepak Gupta et al. (Eds), Proceeding International Conference on Innovative Computing and Communications. Lect. Notes in Networks, Syst., Vol. 471, Springer, Singapore, https://doi.org/10.1007/978-981-19-2535-1_24.

[02] K.Samruddhi, & Kumar, D. R. (2020, September). Used Car Price Prediction using KNearest Neighbor Based Model. International Journal of Innovative Research in Applied Sci-ences and Engineering (IJIRASE), 4(3), 686-689.

[03] Samruddhi, K.; Kumar, R.A. Used Car Price Prediction using K-Nearest Neighbor Based Model. Int. J. Innov. Res. Appl. Sci. Eng. 2020, 4, 629–632.

[04] Kaushal Kishor " Study of quantum computing for data analytics of predictive and prescriptive analytics models", Book Chapter in "Quantum-Safe Cryptography", DE GRUYTER, PP. 121-146., 2023, ISBN 978-3-11-079800-5 e-ISBN (PDF) 978-3-11-079815-9 e-ISBN (EPUB) 978-3-11-079836-4 ISSN 2940-0112, DOI: https://doi.org/10.1515/9783110798159010.

[05] https://github.com/Inderdev07

[06] https://github.com/Inderdev07/car-price-prediction

[07] https://github.com/Inderdev07/car-price-prediction/blob/main/CarPrice_Assignment.csv

[08] https://colab.research.google.com/drive/1d83lrxcMinjhxpClfvl0jjgh75BhEIom