



Summer Training Project Car Price Prediction model

Presentation by:
Mohan dev Vashisht
AIML (T-10)
01315611622

Introduction

Objective :

To develop a Machine Learning model which accurately predicts the price of a car based on several features like Company, Model type, Manufacturing year, Engine size, Fuel type etc using algorithms like Linear Regression and KNN.

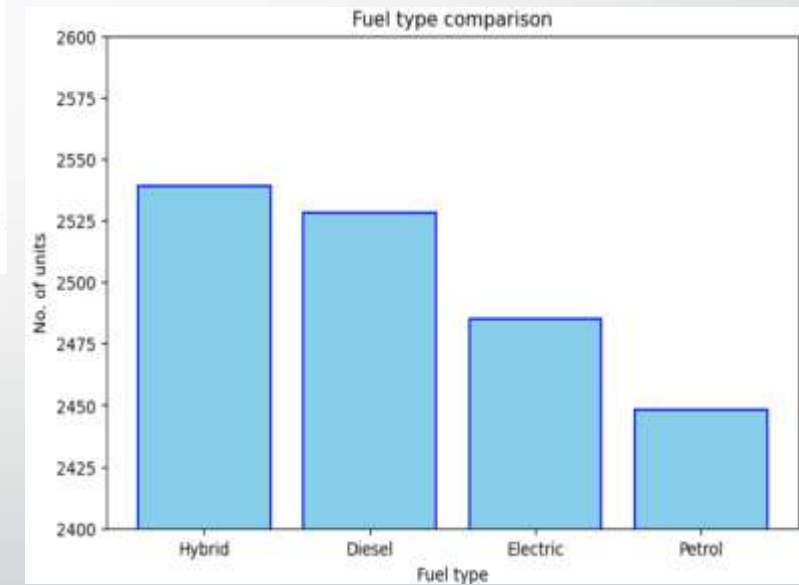
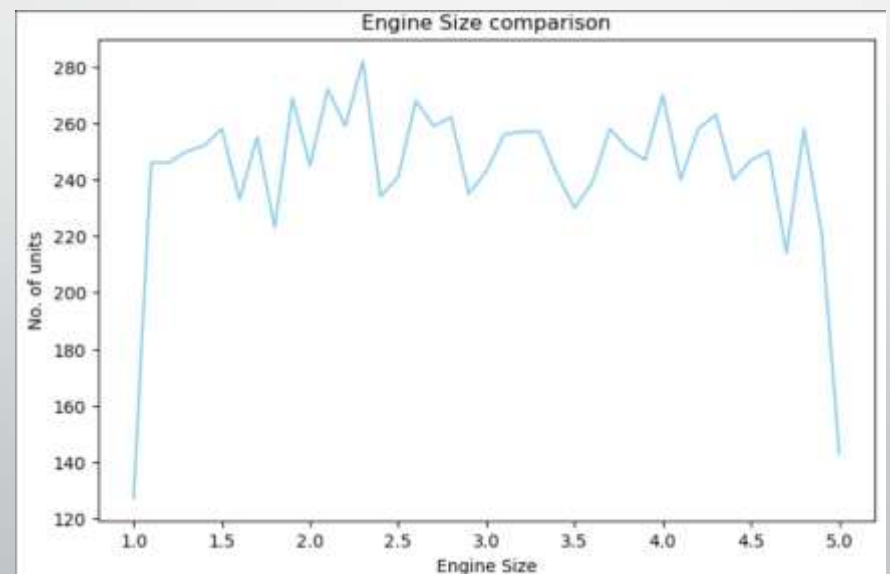
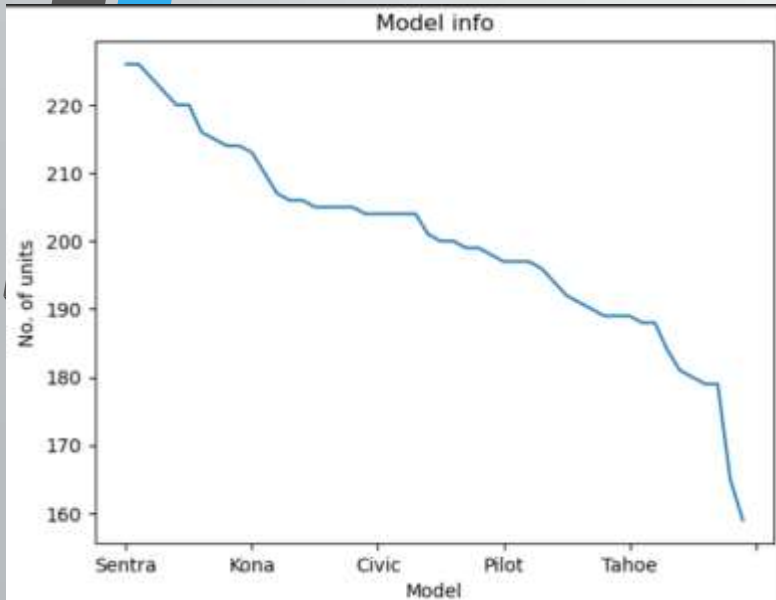
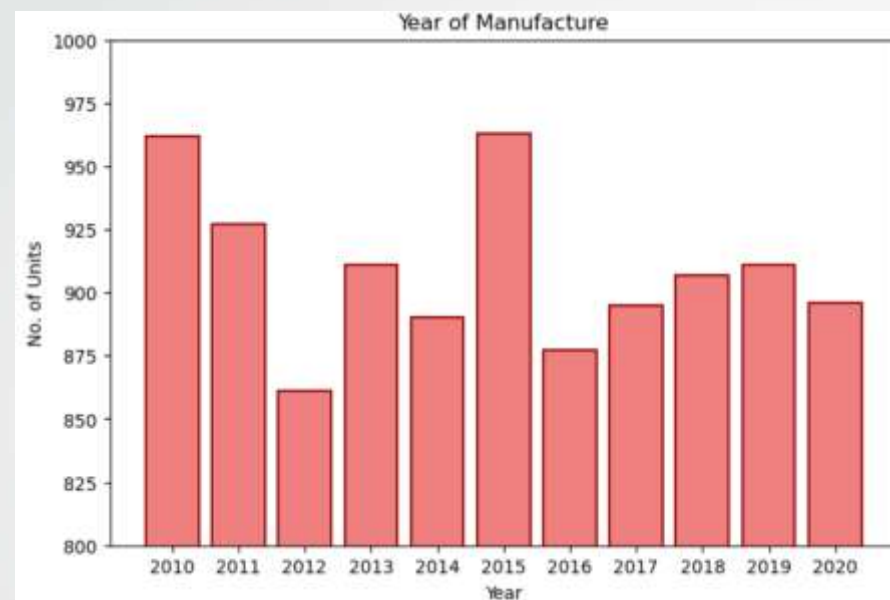
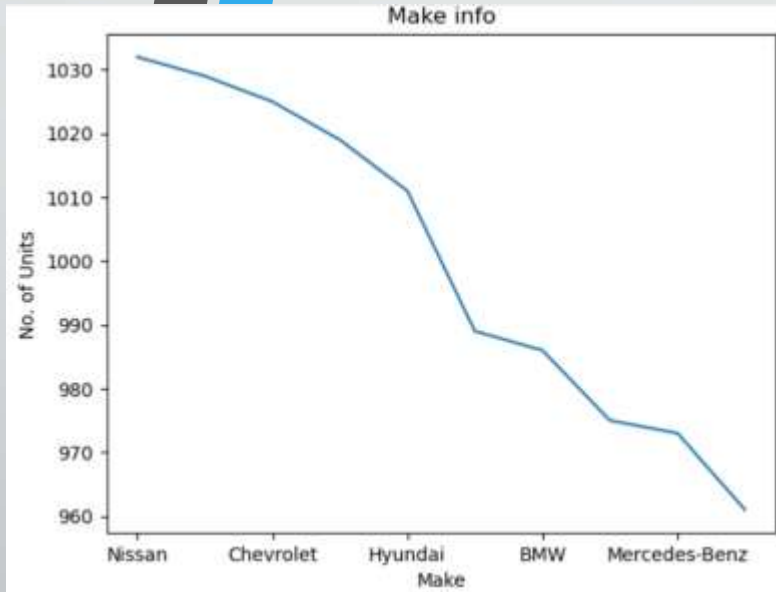
Exploratory Data Analysis

During EDA, we adopt a visual approach to analyze a dataset using statistical and graphical techniques, which ultimately enables us to identify patterns and relationships in a dataset.

The dataset provided to us consisted of several features pertaining to different attributes of a car like the Company, model, fuel type etc.

Several insights were drawn from the data which are discussed in the upcoming slides.

EDA on all the features and attributes



Data Preprocessing

- We first tried to find missing values using the `isnull().sum()` function but fortunately the dataset didn't have any missing values.
- However the data points like company name, model and fuel type consisted of values with the 'Object' data type, they were encoded to 'int' type using replace function and label encoder

Model Building

- Choice of Algorithm : Linear Regression
- Reason : Linear regression is a common method for predicting car prices because it can predict values based on the correlation between variables. It is also quite simple to interpret and is very versatile in its nature of application.

Model Evaluation

- The score of our linear regression model came out to be 0.003% with the mean squared error being 344137528.60339.

```
[38]: lin_reg_model.score(X_test, Y_test)
```

```
[38]: 0.0003067525447315589
```

```
[37]: from sklearn.metrics import mean_squared_error
```

```
Y_pred = lin_reg_model.predict(X_test)  
mse = mean_squared_error(Y_test, Y_pred)
```

```
print("Mean Squared Error:", mse)
```

```
Mean Squared Error: 344137528.6033962
```

- The factors contributing to the low accuracy along with the ways to optimize it are discussed in the following slides.



Insights regarding the factors

- The dataset provided to us consisted of only 3 main features to carry out the price prediction, which are the company, model and fuel type.
- These features comparatively provide lesser number of data points for the model to be trained successfully with a high accuracy.

Insights regarding the solution

- However if we consider a different dataset which consisted of some additional features like kilometers driven, transmission type and seller type, we are able to train our model with a much higher accuracy due to the availability of more diverse data points.
- The sample dataset is as follows-

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0
...
296	city	2016	9.50	11.60	33988	Diesel	Dealer	Manual	0
297	brio	2015	4.00	5.90	60000	Petrol	Dealer	Manual	0
298	city	2009	3.35	11.00	87934	Petrol	Dealer	Manual	0
299	city	2017	11.50	12.50	9000	Diesel	Dealer	Manual	0
300	brio	2016	5.30	5.90	5464	Petrol	Dealer	Manual	0

301 rows × 9 columns

Evaluation of our new model

- The score of our new model which incorporated some more features came out to be 0.83 with the mean square error being 2.1501 which is quite satisfactory for a linear regression model

```
[76]: lin_reg_model.score(X_test,Y_test)
```

```
[76]: 0.836576671502687
```

```
[83]: from sklearn.metrics import mean_squared_error
```

```
Y_pred = lin_reg_model.predict(X_test)  
mse = mean_squared_error(Y_test, Y_pred)
```

```
print("Mean Squared Error:", mse)
```

```
Mean Squared Error: 2.150129918983868
```

Conclusion

- From these insights, we can derive that if we are working on more relevant features and are thoroughly equipped with the domain knowledge of the subject, we can successfully build models which are highly accurate and reliable.
- Thus we are able to successfully predict the price of a car based on the selected features and attributes like company, model, transmission type etc.

Scope for future work

- This project can be further amplified by incorporating the use of Neural Networks to assess other features like paint wear-off, visual conditions of the interior etc.
- More data can be used to increase the data points for more accurate training of the model.
- Lastly, the developed model can be deployed as a web application using python frameworks like Flask or Django.

Thank you!