Understanding AI

Exercise 1

Title: Analyzing Second-Hand Car Sales Data with Supervised and Unsupervised Learning Models

Abstract:

This study examined machine learning models to forecast second-hand car prices in the UK. The dataset was a mock representation of the same comprising 50,000 samples and 7 features. Both supervised and unsupervised learning models were applied to find the best predictive model for second-hand car price prediction. These included Single Linear Regression, Polynomial Linear Regression, Multiple Linear Regression, Random Forest Regression, Artificial Neural Network, K-means clustering, and DBSCAN Clustering. The models were evaluated based on their respective R2 scores and Mean Absolute Errors (MSEs). In some cases, Silhouette Scores were also utilized to determine the performances of unsupervised models. Among all models, the ANN model exhibited the highest R^2 value i.e. 0.9999, and the lowest MSE value i.e. 1.849325e+04.

Introduction:

With the advent of online portals like AutoTrader, Cazoo, Lookers, The AA, and many more, it has become easier for both buyers and sellers to understand the trends and the patterns that affect the value of used cars on the market. That indicates that the monetary value of used cars is increasing at a rapid pace. For this purpose, Machine Learning comes into play. Forecasting a Second-hand car's price might be dependent on a lot of aspects. The mock data under consideration included Manufacturer, Model, Engine Size, Fuel Type, Year of manufacturer, Mileage and Price to be predicted. Before diving into modeling, it's essential to preprocess the data, find correlations, and drop any unnecessary material to make more sense of the data. From previous studies, random forest has also proved to be a good fit to predict second-hand car price. “From experimental results, the training accuracy was found out to be 95.82%, and the testing accuracy was 83.63%.” (Pal et al., 2019:1).

Materials and Methodology:

The latest Python Version (3.12.4) including the libraries Numpy, Pandas, Sk-learn, Tensoflow, Keras, Matplotlib, and Seaborn in a Jupyter Notebook environment was involved in the study. The entire study could be broken down into 2 halves of experimentation; supervised and unsupervised learning. While regression models are the optimal choice as we encountered a numerical target variable, unsupervised learning models were also discussed so they could be compared to similar previous studies.

Data Preprocessing:

The first step after necessary preprocessing was to find the correlation between numerical features. The findings from the correlation matrix are interpreted on a scale from -1 to +1. The results suggested Year of manufacture and Mileage will be convenient while modeling afterward.

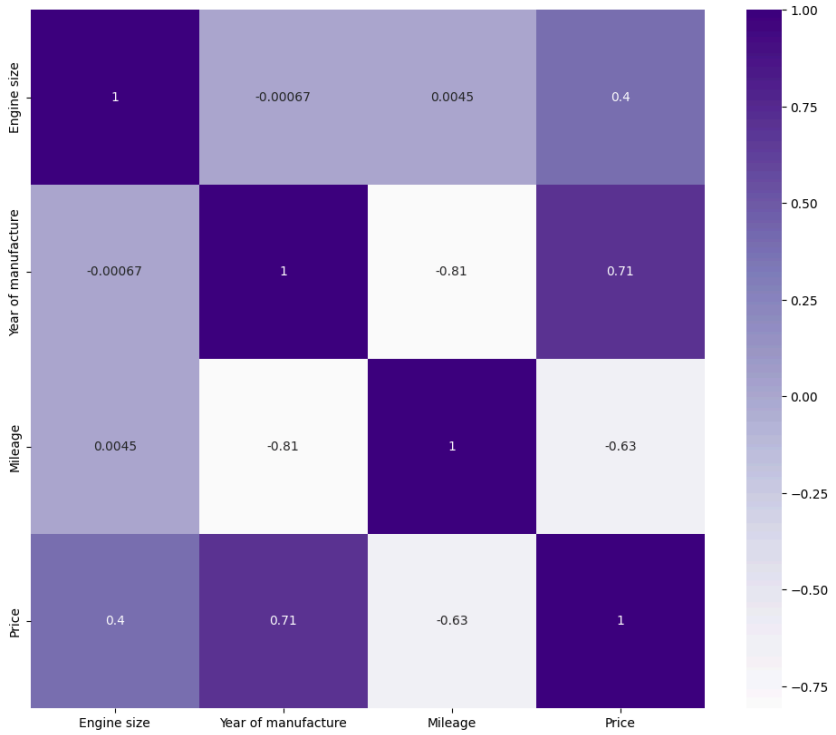


Fig 1. Correlation Matrix

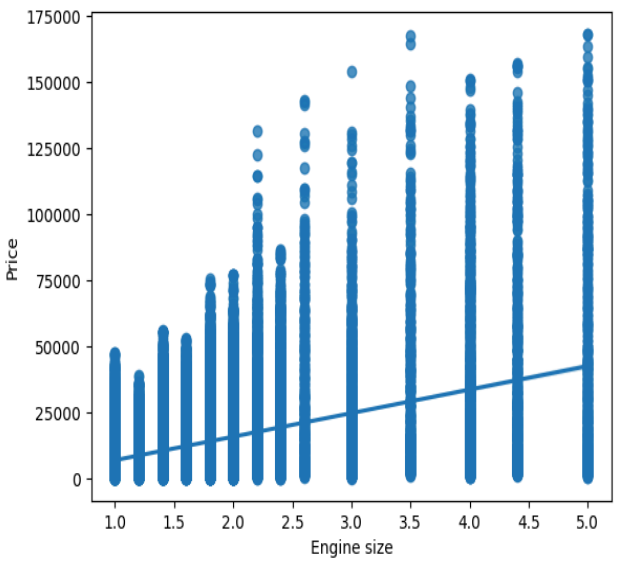
Model Building:

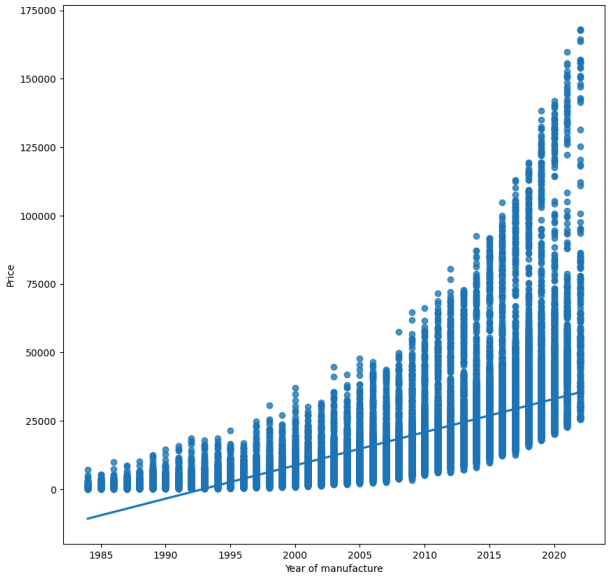
For Data Splitting, 20 percent of the data was spared for testing and the rest for training. For modeling, a single linear regression technique to train each numerical input feature at once to forecast price was applied. In Machine Learning, to evaluate the model is to assess its performance using different evaluation metrics. Here, we used R-squared and MSE. The R-squared measure indicates how strongly your model is related to the dependent variable, expressed on a scale from 0 to 100% (Jim Frost, 2018). Whereas, a lower MSE indicates a good-performing model. It calculates the mean of the squared differences between forecasted values by the model and real target values. Squaring these differences prioritizes larger errors, increasing sensitivity to outliers (Encord Computer Vision Glossary, 2013).

Single Linear Regression:

Linear regression is characterized as a regression algorithm i.e. that predicts a numerical output and works on a linear relationship between a dependent or target variable and one or more independent or predictor variables (Kim et al., 2022:4-5). The linear equation describes the relationship between both the dependent and target variables.

$$y = w1x1 + b$$





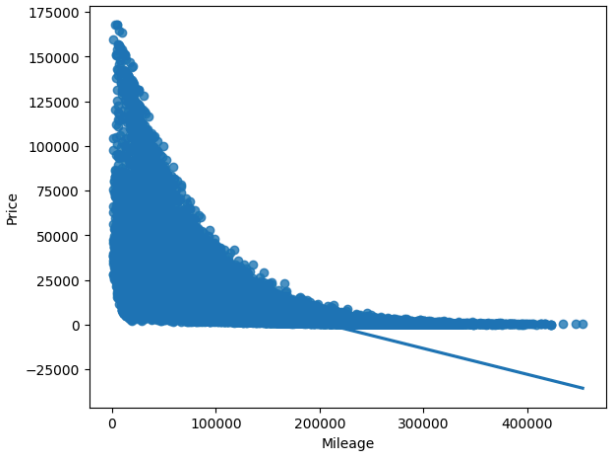


Fig 2. Regplots between every numerical feature and price

Polynomial Regression:

Polynomial regression involves fitting a polynomial function of degree n to model the relationship between the dependent variable Y and independent variable(s) X. This method seeks to optimize the curve fitting process using observed data points. (Simplilearn, 2023).

Here we kept the degree equal to 2.

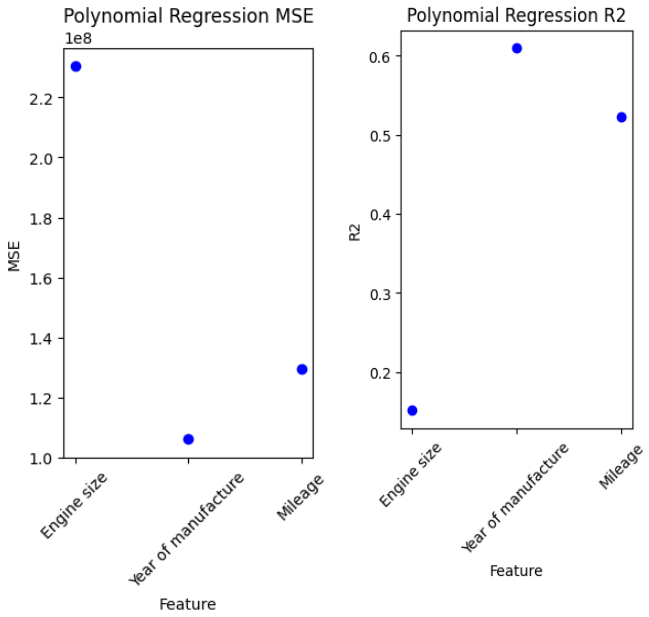


Fig 3. Scatterplots showing evaluation scores

Multivariate Linear Regression:

Multivariate or Multiple Linear Regression works in the same way as the single linear regression. The only thing differing between these two is Multivariate LR, as its name suggests, takes in multiple input features simultaneously.

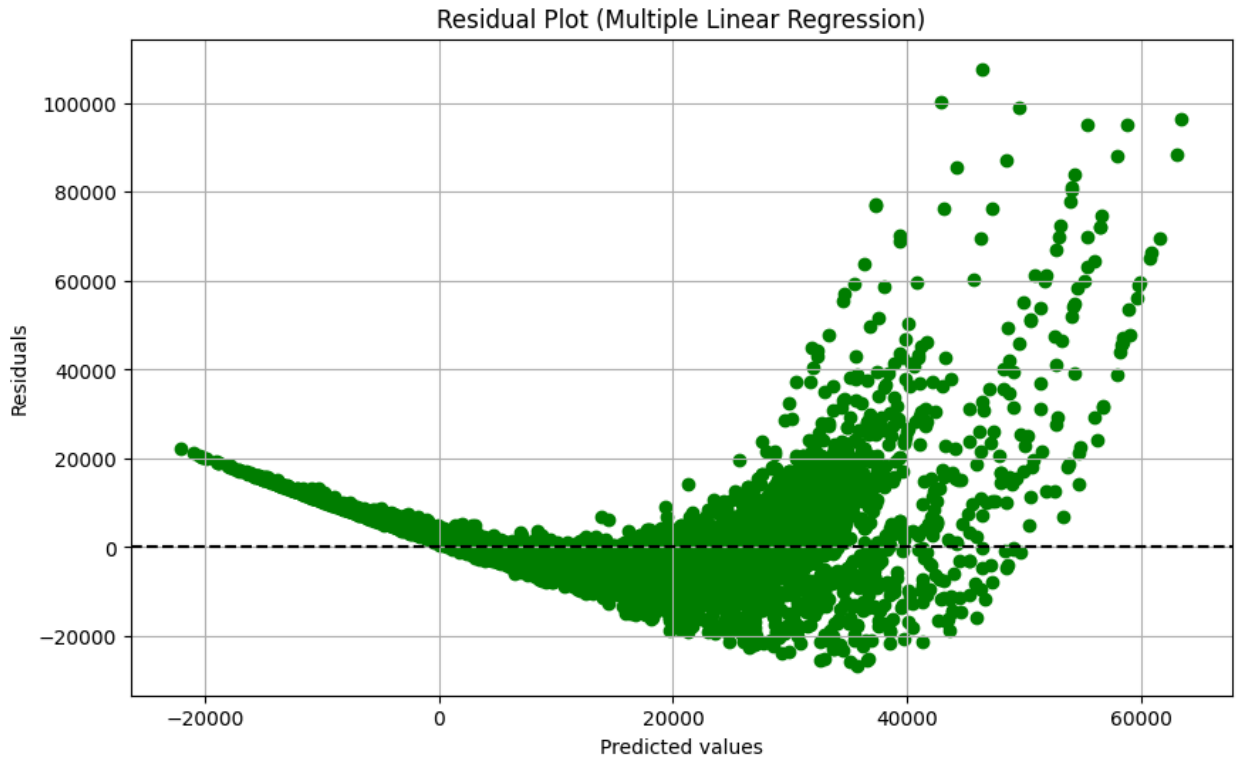


Fig 4. Residual Plot between actual and predicted values by MLR

Random Forest Regression:

A Random Forest regression model provided by sklearn, aggregates multiple decision trees into a unified model. As for the random forest regression, we require preprocessing and fitting the numerical data as well as the categorical data to capture the complex patterns of the data which could lead to a better model performance.

Artificial Neural Network:

An artificial neural network, as its name suggests, is composed of neurons and is an advanced machine-learning model designed to mimic decision-making processes akin to the human brain's cognitive functions (IBM, 2024). After a series of experiments, the ANN model exhibited better results than all the other models yet.

KMeans Clustering:

K-means clustering, derived from signal processing and applied in vector quantization, aims to group n observations into k clusters. Each observation is allocated to the cluster whose centroid (or mean) is closest, thereby acting as a representative for that cluster (Pulkit Sharma, 2024).

DBSCAN Clustering:

For this model, a grid search was utilized to fit and evaluate clusters using DBSCAN hyperparameters (eps and min_samples). The cluster counts and descriptive statistics were then printed for each combination to determine the best parameters by examining the distribution and aspects of the clusters formed.

Results and Discussions:

Model	**Features and Evaluation Metrics**					
Single Linear Regression	**Feature**	**Linear MSE**		**Linear R^2**		
	Engine size	2.304992e+08		0.150626		
	Year of manufacture	1.326790e+08		0.511087		
	Mileage	1.624686e+08		0.401314		
Polynomial Regression	**Feature**	**Linear MSE**		**Linear R^2**		
	Engine size	2.303262e+08		0.151263		
	Year of manufacture	1.059939e+08		0.609419		
	Mileage	1.296203e+0		0.522358		
Multivariate Linear Regression	**Features**	**MSE**		**R^2**		
	Engine size, Year of manufacture, Mileage	8.915862e+07		0.67145		
Random Forest Regression	**Features**	**MSE**		**R^2**		
	Manufacturer, Model, Engine size, Fuel type, Year of manufacture, Mileage	4.100258e+05		0.998489		
Artificial Neural Network	**Features**	**Loss**	**Val_loss**	**MSE**	**R^2**	
	Manufacturer, Model, Engine size, Fuel type, Year of manufacture, Mileage	13715.0996	19732.3652	1.849325e+04	0.999932	

KMeans Cluster Analysis	Features	Optimal Cluster no	Optimal Columns used	Silhouette Score
	Engine size, Year of manufacture, Mileage	2	(1, 2)	0.533

DBSCAN Cluster Analysis	Features	No of Clusters	No of noise points
	Engine size, Year of manufacture, Mileage	7	11

From the above findings, the following conclusions are made:

- Year of manufacture is the best numerical predictor when using single LR displaying the least MSE and highest R^2. Whereas, with Polynomial Regression of 2nd degree, each numerical input exhibited a negligibly better evaluation score.
- For multiple linear regression (MLR), the R^2 showed strong performance, but the higher MSE limits definitive conclusions.
- The Random Forest Regression' R^2 was 0.998489, which is an ideal value but the MSE was again too high to call it the best fit.
- An ANN model with these settings; 1 Input layer, 2 hidden layers, and an output layer. Number of neurons: 32, 64, 64, and 1 respectively. Activation functions: relu in each except the output layer which was kept to default. Kernel regularizer: 12(0.001). Epochs: 100. Batch size: 64. Validation split: 0.2, resulting in the best model yet with low MSE, high R^2, and quite consistent and low loss and val_loss values.
- Out of 27 combinations of clusters and different numerical input features for KMeans Cluster Analysis, the optimal settings with 2 clusters, Year of manufacturer, and Mileage, a silhouette score of 0.533 was achieved which is not quite convincing.
- The k-means model with a silhouette score of 0.533 performed way better in terms of clustering compared to the DBSCAN model with more noise points.