**MANCHESTER METROPOLITAN UNIVERSITY**

**Course Title:**

Machine Learning Concepts

**Assignment Title**:

Machine Learning Project

**Full Name**:

MMU ID:

# Introduction This is a Coursework project for Machine Learning Concepts. The dataset is a Car Sale Adverts dataset provided by Auto Trader, one of Manchester Metropolitan University-industry partners. An overview of the business Auto Trader is the UK and Ireland's largest digital automotive marketplace. It’s the go-to destination for car buyers and has been for the past 40 years. Auto Trader exists to Drive change together. Responsibly, Its aim is to grow both car-buying and selling audiences. It also aims to change how the UK shops for cars by providing the best online car-buying experience and enabling all retailers to sell online. It aims to build stronger partnerships with its customers, use its voice and influence to drive more environmentally friendly vehicle choices, and create a diverse and inclusive culture. Auto Trader was listed on the London Stock Exchange in March 2015 and is now a member of the FTSE 100 Index Auto Trader's Website.

# Data/Domain Understanding and Exploration The dataset contains an anonymised collection of adverts with information on vehicles such as brand, type, colour, mileage, as well as the selling price. 2.1 Meaning and Type of Features; Analysis of Distributions Features Definitions public\_reference: The unique ID for each car advert. mileage: The total distance covered by the car to date. reg\_code: This is the unique code for each year of registration. standard\_color: The colour of the car. standard\_make: The brand of the car. standard\_model: The brand model of the car. vehicle\_condition: The vehicle condition; either NEW or USED. year\_of\_registration: The first registration year of the car. price: The price of the car in pounds(£). body\_type: The body type of the car i.e SUV, Saloon, Minibus, and so on. crossover\_car\_and\_van: A crossover is a type of automobile with an increased ride height that is built on unibody chassis construction shared with passenger cars, as opposed to traditional sport utility vehicles (SUV) which are built on a body-on-frame chassis construction similar to pickup trucks. fuel\_type: The type of fuel the car runs on.

# General information about features, non\_missing values count, data types, and also the shape of the dataset.

RangeIndex: 402005 entries, 0 to 402004

Data columns (total 12 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 public\_reference 402005 non-null int64

1 mileage 401878 non-null float64

2 reg\_code 370148 non-null object

3 standard\_colour 396627 non-null object

4 standard\_make 402005 non-null object

5 standard\_model 402005 non-null object

6 vehicle\_condition 402005 non-null object

7 year\_of\_registration 368694 non-null float64

8 price 402005 non-null int64

9 body\_type 401168 non-null object

10 crossover\_car\_and\_van 402005 non-null bool

11 fuel\_type 401404 non-null object

dtypes: bool(1), float64(2), int64(2), object(7)

memory usage: 34.1+ MB

# Descriptive Statistics of Numerical Features

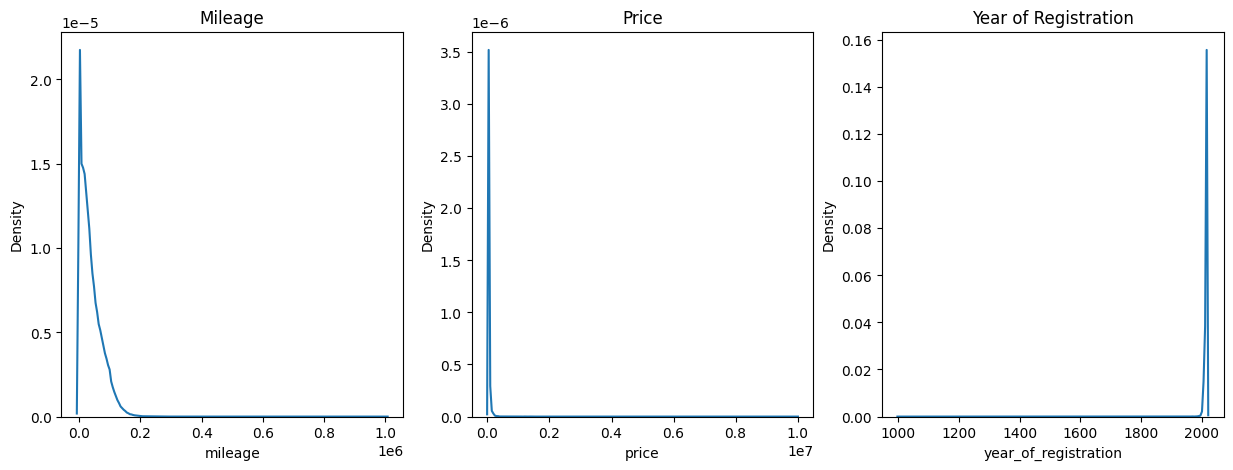
[

|  | **public\_reference** | **mileage** | **year\_of\_registration** | **price** |
| --- | --- | --- | --- | --- |
| count | 4.020050e+05 | 401878.000000 | 368694.000000 | 4.020050e+05 |
| mean | 2.020071e+14 | 37743.595656 | 2015.006206 | 1.734197e+04 |
| std | 1.691662e+10 | 34831.724018 | 7.962667 | 4.643746e+04 |
| min | 2.013072e+14 | 0.000000 | 999.000000 | 1.200000e+02 |
| 25% | 2.020090e+14 | 10481.000000 | 2013.000000 | 7.495000e+03 |
| 50% | 2.020093e+14 | 28629.500000 | 2016.000000 | 1.260000e+04 |
| 75% | 2.020102e+14 | 56875.750000 | 2018.000000 | 2.000000e+04 |
| max | 2.020110e+14 | 999999.000000 | 2020.000000 | 9.999999e+06 |

**Descriptive Statistics of Categorical Features**

| **reg\_code** | **standard\_colour** | **standard\_make** | **standard\_model** | **vehicle\_condition** | **body\_type** | **crossover\_car\_and\_van** | **fuel\_type** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| count | 370148 | 396627 | 402005 | 402005 | 402005 | 401168 | 402005 | 401404 |
| unique | 72 | 22 | 110 | 1168 | 2 | 16 | 2 | 9 |
| top | 17 | Black | BMW | Golf | USED | Hatchback | False | Petrol |
| freq | 36738 | 86287 | 37376 | 11583 | 370756 | 167315 | 400210 | 216929 |

**Analysis of Univariate Distribution**

****

Both the Mileage and Price histogram plots are rightly skewed. This implies that the mean of these features is greater than its median. Also, for the Mileage, 77% of cars have mileages 0km/h and 60,000km/h, 10% of cars have between 60,000km/h and 80,000km/h, and 13% have mileages greater than 80,000km/h. For Price, 94% of the car prices are between £120 and £40,000. 6% of cars have prices above £40,000. For the Year\_of\_registration, 91% of cars were registered between the years 2000 and 2020.

# Data Processing for Machine Learning

This report covers the processes required to prepare a Car Sales Adverts dataset for machine learning analysis provided by Auto Trader, one of Manchester Metropolitan University's industry partners. The dataset contains an anonymized collection of advertisements containing vehicle information such as brand, type, colour, mileage, and selling price. This preparation cleans the dataset, handles missing values, encodes categorical variables, removes outliers, and features engineers the data for modelling.

**Steps in Data Preprocessing**

1. **Dropping extraneous Columns:** The columns "public\_reference", "reg\_code", and "crossover\_car\_and\_van" were removed since they were deemed extraneous to the analysis.
2. **Encoding Categorical Variables:** The LabelEncoder function was used to encode the categorical columns ("standard\_colour", "standard\_make", "standard\_model", "vehicle\_condition", "body\_type", and "fuel\_type").
3. **Missing Values:** KNNImputer was used to handle missing values in the dataset's first seven columns.
4. **Outliers** were removed by calculating the interquartile range (IQR) for each column and identifying outliers using the IQR. The dataset was cleaned up by removing the outliers.
5. **Feature Engineering:** By computing the age of the car and the odometer reading every year, two new features, "age" and "odometer\_reading," were generated. The columns "mileage" and "year\_of\_registration" were removed.
6. **Data Scaling:** MinMaxScaler was used to scale the data.

**Conclusion**

Finally, the Car Sales Adverts dataset was cleaned and encoded, missing values were handled, outliers were removed, and new features were developed. After that, the dataset was scaled and is now ready for machine learning modelling. These preprocessing processes will allow us to create a more accurate machine learning model capable of accurately predicting car prices.

**Feature Engineering**

**Polynomial feature transformation:**  is a frequent feature engineering method for creating relationships between existing features.

A degree of 2 is defined in this code to generate second-order polynomial features. The PolynomialFeatures object is then used to change the training and test data. This method can increase model performance by allowing it to capture more complex correlations between input variables and target variables, but it must be carefully tuned to avoid overfitting.

**Benefits of using Polynomial:**

* Increases the model's complexity, allowing it to capture more intricate interactions between input and target variables.
* Can increase model performance, especially when the underlying relationship is nonlinear.
* Provides a method for generating new features by combining current features, including their powers and products.
* Provides a flexible technique to model variable relationships without specifying a precise functional form.
* It is simple to build using tools such as scikit-learn.
* By identifying potential connections and non-linear correlations, it is possible to gain a deeper understanding of the underlying data.

Feature Selection and Dimensionality Reduction

The **SelectKBest** method from the sklearn library is used to choose features in this step of the project.

This approach chooses the top k features based on their scores as determined by a scoring formula. The f\_regression scoring function is utilised in this example, and the best 5 features are chosen. The fit\_transform method is used to fit the selection to the training data and transform it by removing all but the top five characteristics. Using the transform method, the same transformation is applied to the test data. The specified features' indices are obtained by calling the get\_support function with the indices=True argument. These indices are then used to extract the names of the chosen features from the polynomial feature names created by the PolynomialFeatures function.

**There’s Top 5 Features :**

* body\_type
* standard\_colour body\_type
* year\_of\_registration body\_type
* body\_type^2
* body\_type fuel\_type

# Model Building

**Models used:**

* Linear Regression
* Random Forest
* Gradient Boosting
* Ensemble Model (using voting)

**Techniques used:**

* Feature Engineering
* Grid Search (for optimizing hyperparameters)

**Regression Models:**

This research included three regression models: linear regression, random forest, and gradient boosting. The linear regression model was utilised as a baseline model due to its simplicity, while the random forest and gradient boosting models were chosen due to their higher performance when compared to linear regression. Using the three models, an ensemble model based on voting was also developed.

# Model Building

**Linear Regression:**

The scikit-learn library was used to build the linear regression model. The model was trained on the training set using the selected features obtained from feature engineering. The model's performance was tested using the mean squared error (MSE) and mean absolute error (MAE) metrics.

**The Forest at Random:**

The scikit-learn package was used to create the random forest model, and a grid search was used to optimise the model's hyperparameters. n\_estimators = 100, max\_depth = None, and min\_samples\_split = 2 were found to be the best hyperparameters. The model was trained on the training set using the features chosen through feature engineering.

**Gradient Enhancement:**

The gradient boosting model was built with the scikit-learn toolkit and n\_estimators set to 100. The model was trained on the training set using the features chosen through feature engineering. The MSE and MAE metrics were used to evaluate the model's performance.

**Ensemble Model:**

An ensemble model was also trained via voting, which incorporates the predictions of the linear regression, random forest, and gradient boosting models. The ensemble model was trained on the training set using the selected features obtained from feature engineering. The model's performance was tested using the MSE and MAE metrics.

**Conclusion:**

Finally, this experiment demonstrates the effectiveness of employing several regression models to estimate housing prices. The results show that the gradient boosting model fared the best, followed by the ensemble model, random forest, and linear regression. However, it is crucial to note that the models' performance could be enhanced by further optimising the hyperparameters or by utilising more powerful machine learning techniques. Further more, the research might benefit from increased data collection in order to improve the accuracy and resilience of the models.

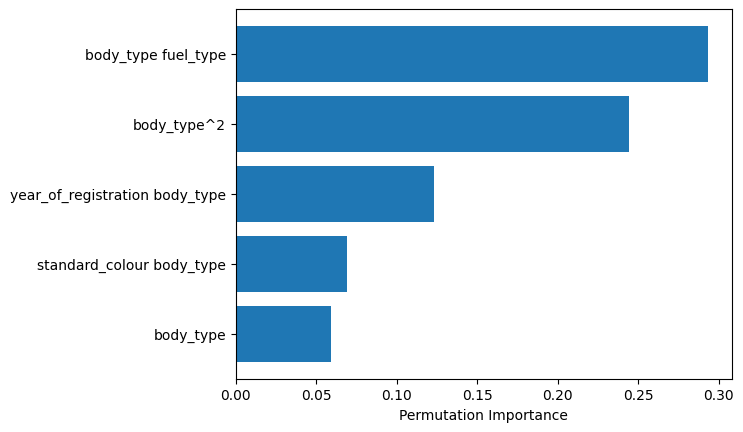
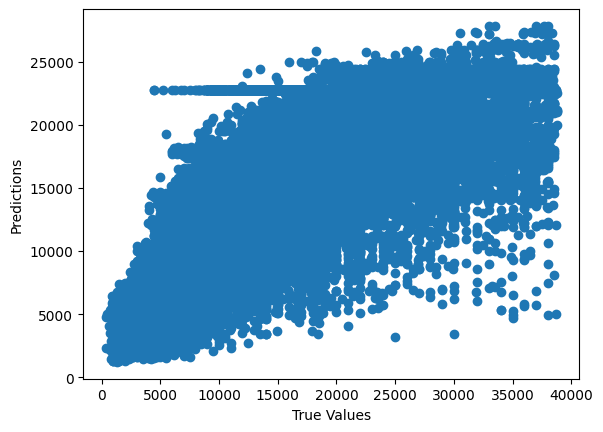
# Model Evaluation and Analysis

The evaluation and analysis of the ensemble model using cross-validation and various metrics is described in this portion of the project report. The ensemble model's cross-validation scores are produced using the cross\_val\_score function with cv=5, which divides the data into 5 folds and analyses the model's performance on each fold. The model's generalisation performance is measured using the mean cross-validation score, which is computed and reported.

The performance of the model on the test set is then evaluated using the mean squared error (MSE) and the coefficient of determination (R2). The MSE is the average squared difference between predicted and actual values, whereas the R2 is the percentage of variance in the target variable explained by the model. The MSE of the ensemble model is 24,195,072, and the R2 score is 0.60, indicating that the model fits the test data reasonably well.

Permutation importance is calculated using the permutation\_importance function to gain insight into the importance of the features utilised in the model. The permutation significance evaluates the loss in model performance when the values of each feature are randomly shuffled while all other characteristics remain intact. To see which features are most significant in forecasting the target variable, the feature importances are plotted using a horizontal bar plot.

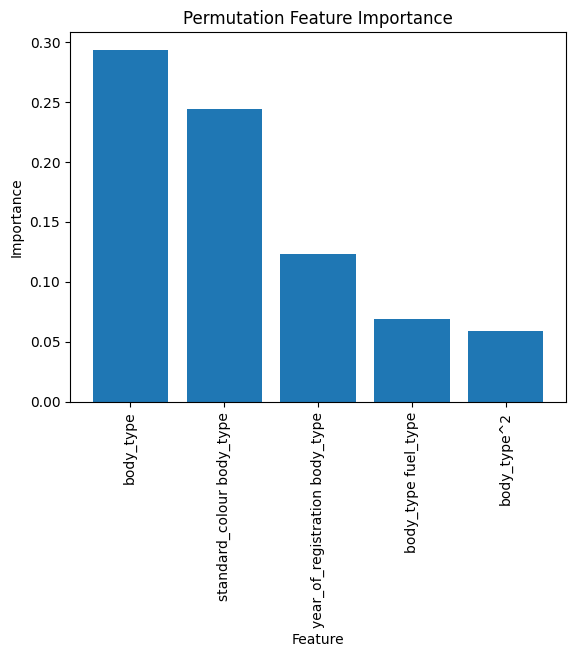
# Model Evaluation and Analysis

Overall, the evaluation and analysis of the ensemble model indicate that it predicts the target variable quite well, with a modest level of generalisation performance and a decent fit to the test data. Further more, the feature importance analysis reveals which features are most essential in predicting the target variable.

# Model Evaluation and Analysis

The ensemble model's feature importance was assessed using permutation importance. Permutation importance is a method for determining a model's feature relevance by permuting the values of each feature and evaluating how much the permutation impacts the model's performance.

A horizontal bar plot was built to visualise the feature importance results, with the features sorted by importance in descending order. The plot revealed that the top features were substantially more important than the other features, indicating that these features had a greater impact on the model's predictions.



Overall, the permutation importance results revealed which features were most essential in forecasting car price values using the ensemble model. This information could be beneficial in enhancing the model's performance or making informed judgements when picking characteristics for future models.