SC1015 MINI PROJECT: DRIVE YOUR BUDGET

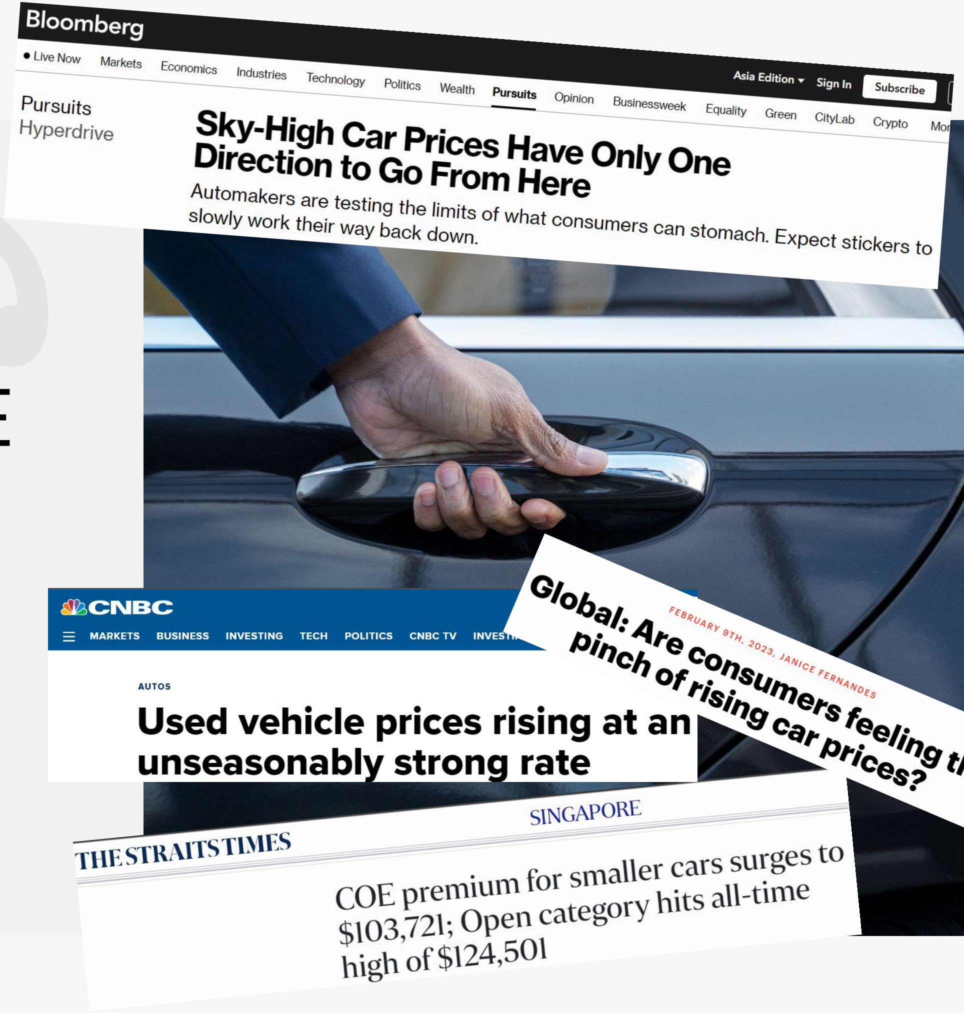
Forecasting Car Prices with Data Science

<u>LAB A137 - Group 6</u>

Sky Lim En Xing - U2223731A
Thomas Tan Keat Hao - U2221472J
Chermine Cheah Xue Min - U22222126F

CAR PRICES ARE **RISING** ACROSS THE WORLD

Can regular joes still afford cars?
How to maximise *value*?



INTRODUCTION

PROBLEM DEFINITION

How do we predict prices using different **features** of a car to help budget-conscious car-buyers ?

SMRITI · UPDATED 3 YEARS AGO

kaggle

car_sales.csv

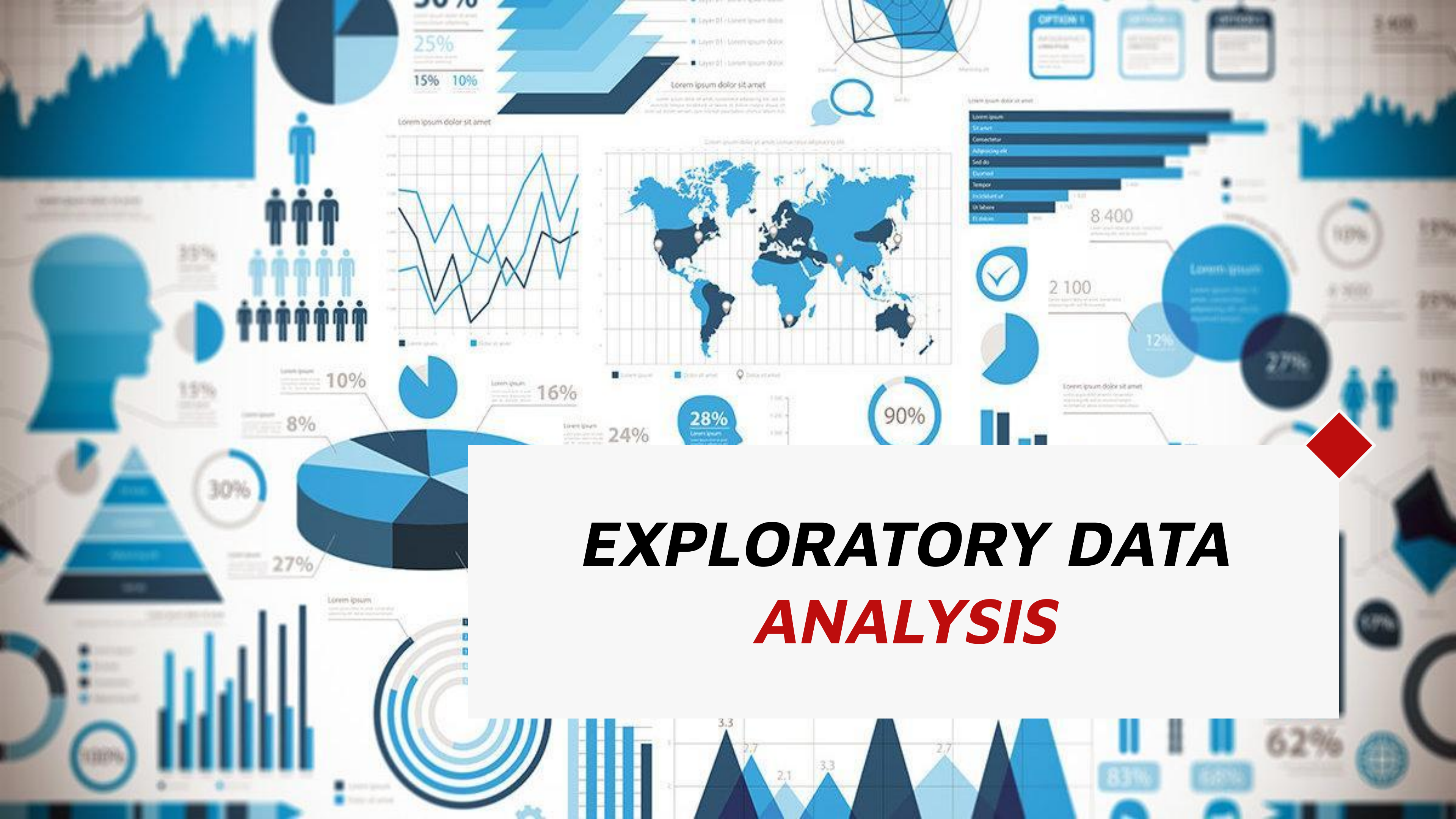
Data Card | Code (5) | Discussion (0)

About Dataset

Context

This data contains data related to Car Sales

DATASET USED



EXPLORATORY DATA ANALYSIS

RAW DATATYPES

	Brand	Price	Body	Mileage	EngineV	Engine Type	Registration	Year	Model
0	BMW	4200.0	sedan	277	2.0	Petrol	yes	1991	320
1	Mercedes-Benz	7900.0	van	427	2.9	Diesel	yes	1999	Sprinter 212
2	Mercedes-Benz	13300.0	sedan	358	5.0	Gas	yes	2003	S 500
3	Audi	23000.0	crossover	240	4.2	Petrol	yes	2007	Q7
4	Toyota	18300.0	crossover	120	2.0	Petrol	yes	2011	Rav 4

```
Brand             object
Price            float64
Body              object
Mileage            int64
EngineV          float64
Engine Type       object
Registration      object
Year               int64
Model             object
dtype: object
```

DATASET **FEATURES**

(Predictors)

CATEGORICAL

- **Brand**
- **Model**
- **Body** (Sedan/Van/Hatchback)
- **Registration** (Year of registration)
- **Engine Type** (Petrol/Gas/Diesel)

NUMERICAL

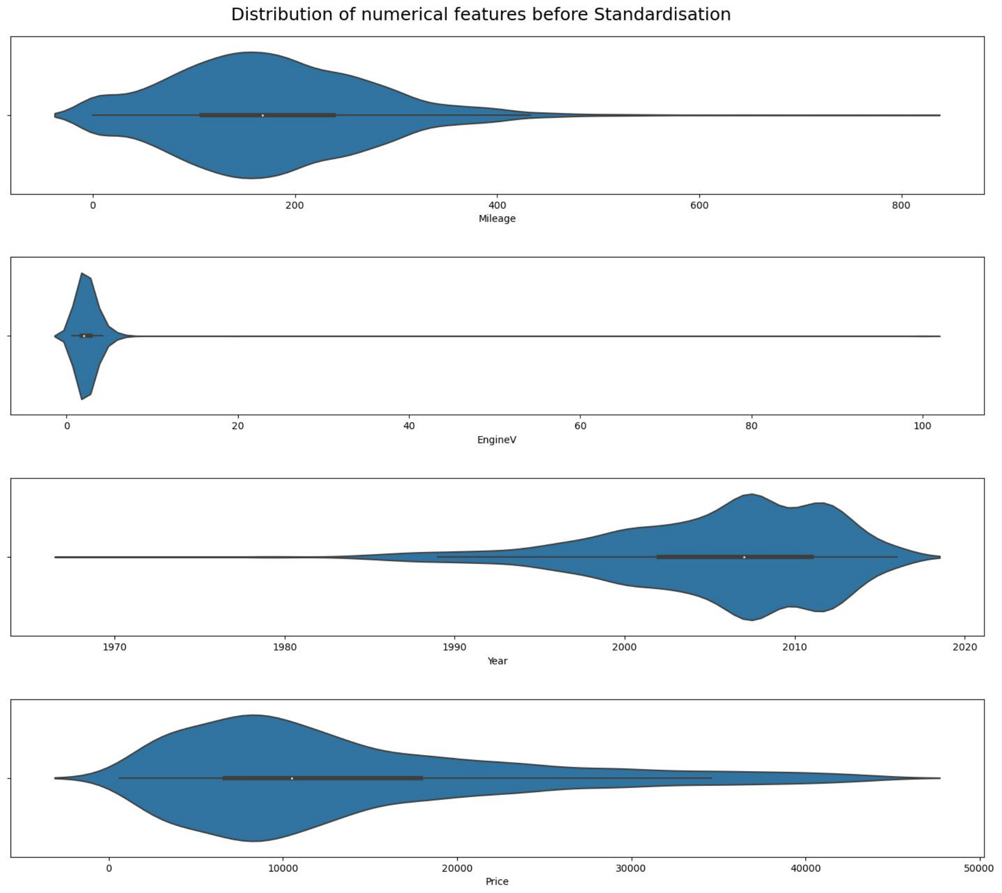
- **Mileage**
- **Year** (Year of manufacturing
- **EngineV** (Engine version of Car))

(Response)

Price
(Numerical)

Distribution of Numerical FTs

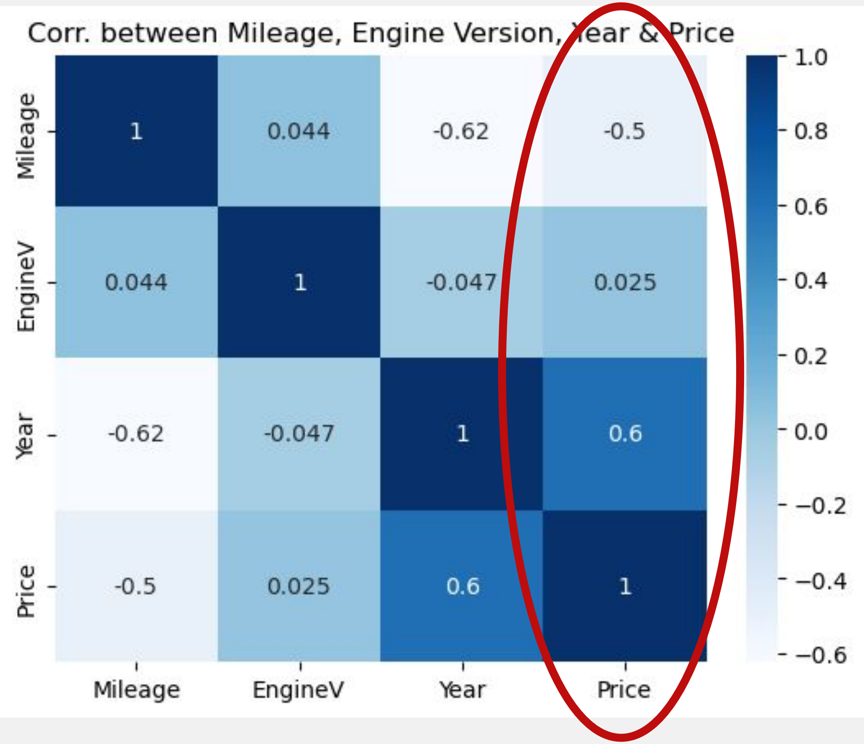
- Note skewness of each variable
- Distributed over a large range of values for each variable
- Issue of **scale**



EXPLORATORY DATA ANALYSIS

Relationship between other Numeric features & Prices

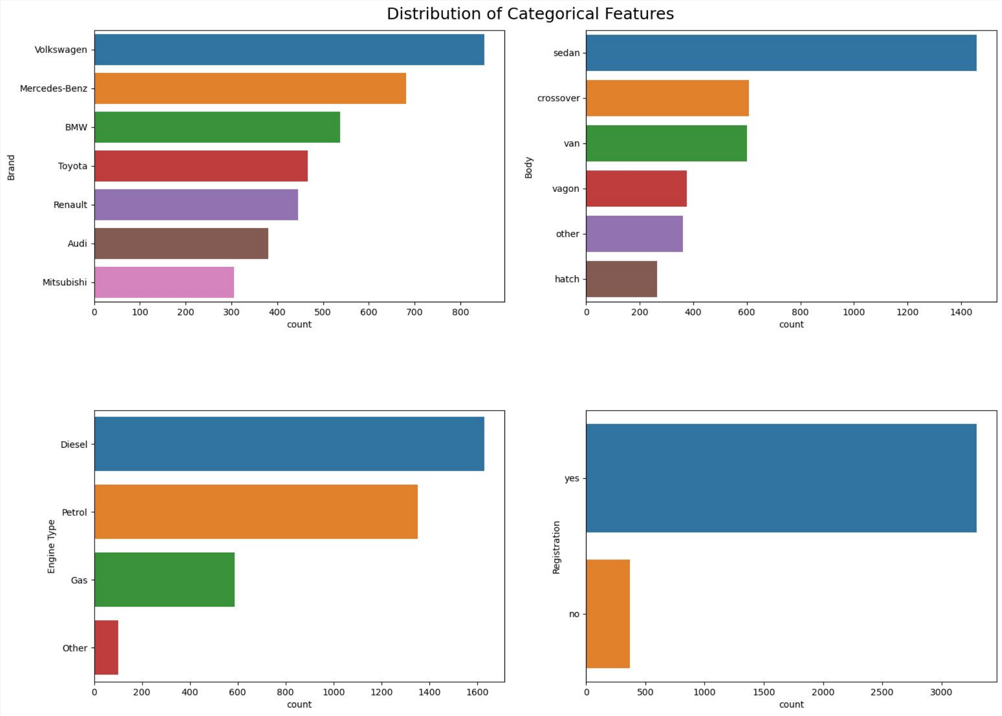
- Only **moderately strong** relationships b/w numerical predictors and response Price
- Later the year of production , ▲ Price
- ▼ Mileage, ▲ Price
- Very **weak** r/s b/w EngineV & Year



EXPLORATORY DATA ANALYSIS

Distribution of Categorical Features

- **Sedans** most popular
- Most cars sold **registered**
- **Diesel** is preferred
- Buyer tendency towards Volkswagen and Mercedes-Benz



EXPLORATORY DATA ANALYSIS

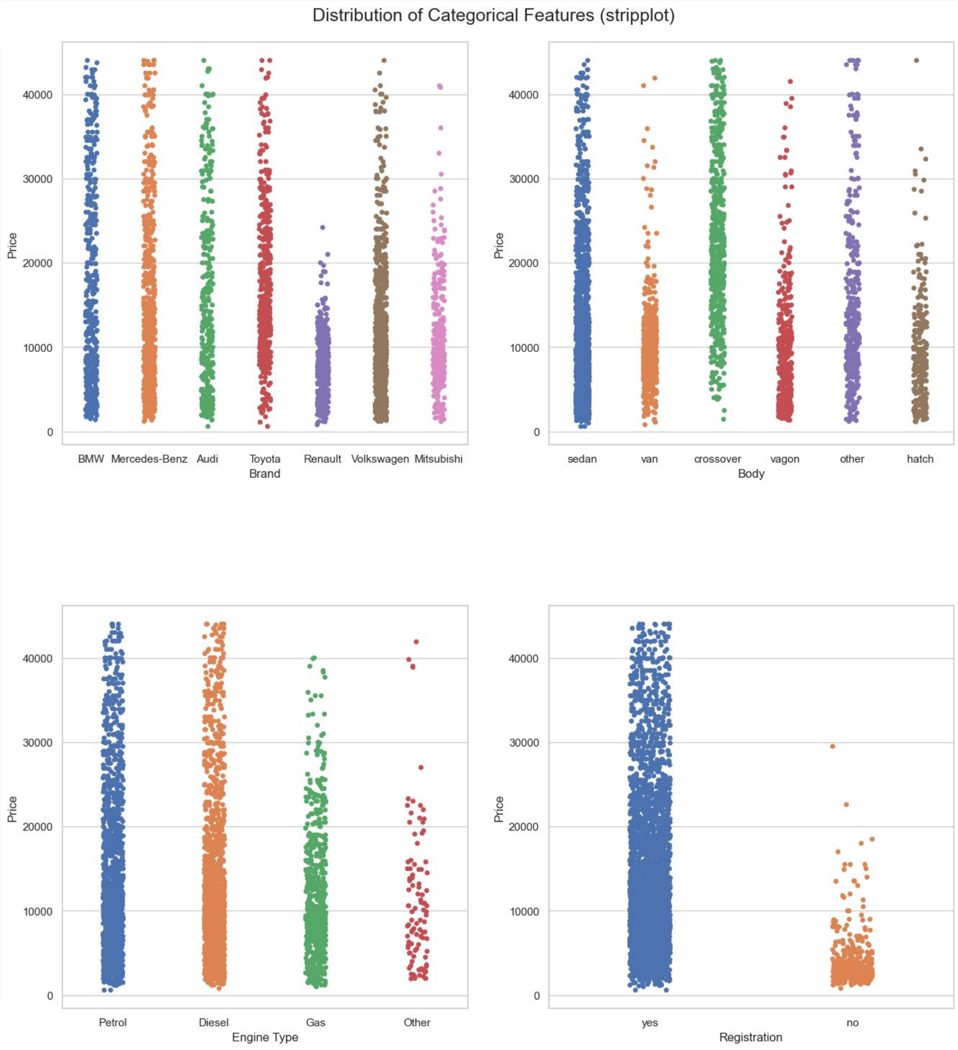
Distribution of Categorical Features

- More data points clustered @ Price <20,000
    - dataset has cheaper cars on avg

- **Renault** and **Mitsubishi** sells more cheaper cars

- **Van**, **Vagon**, **Hatch Body** generally have more cheaper cars

- **Gas** Engine Types cars tends to be cheaper



EXPLORATORY DATA ANALYSIS

DATA-PREPROCESSING

DATA ENGINEERING

= Collecting, cleaning, and transforming raw data (Preparation)

FEATURE ENGINEERING

= Transforming raw data into meaningful features

DATA-PREPROCESSING

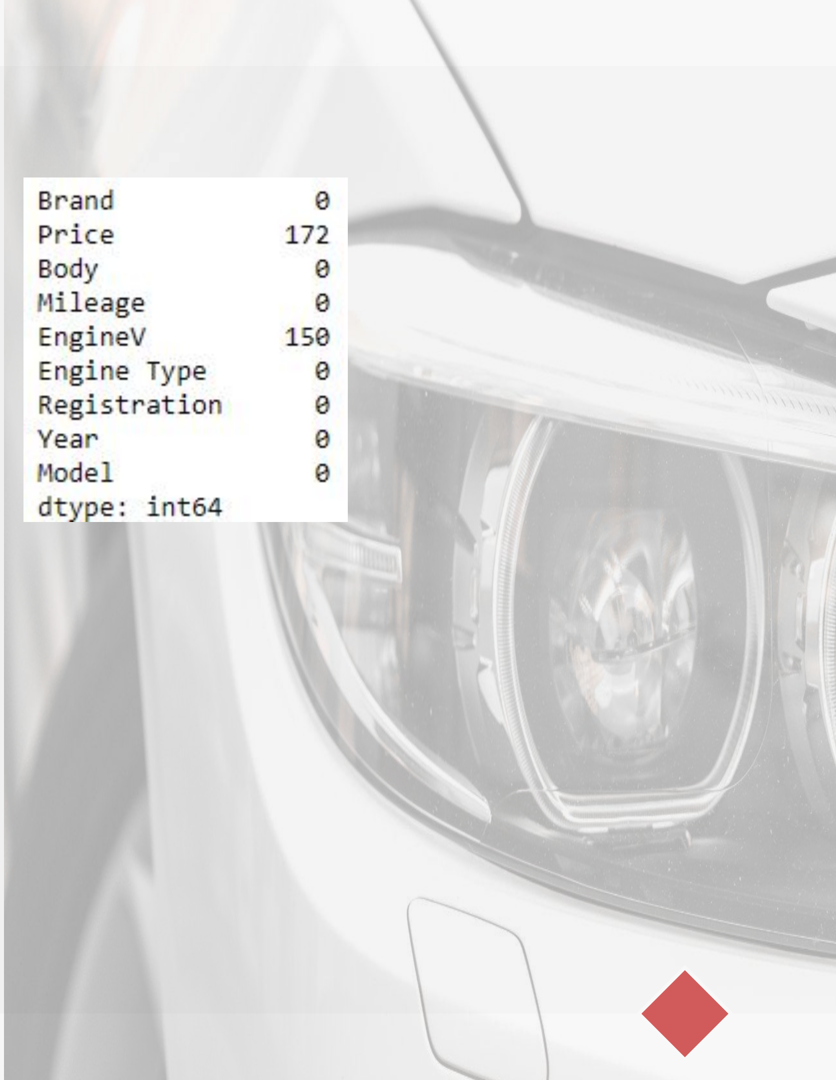
DATA ENGINEERING

- Remove *NULL* values
- Remove *Outliers*
- Appropriate Data Type

```
Brand             0
Price           172
Body              0
Mileage           0
EngineV         150
Engine Type       0
Registration      0
Year              0
Model             0
dtype: int64
```

FEATURE ENGINEERING

- Data Encoding
  for ***Categorical*** data
- Scaling ***Numeric*** data



DATA ENCODING (CATEGORICAL)

ONE HOT ENCODING

For **Brands, Body, Engine Type and Model**

Engine Type
Gas
Petrol
Diesel

is_Gas	is_Petrol	is_Diesel
1	0	0
0	1	0
0	0	1

Example of One-hot encoding for Engine Type

LABEL ENCODING

For **Registration**

Registration
Yes
No

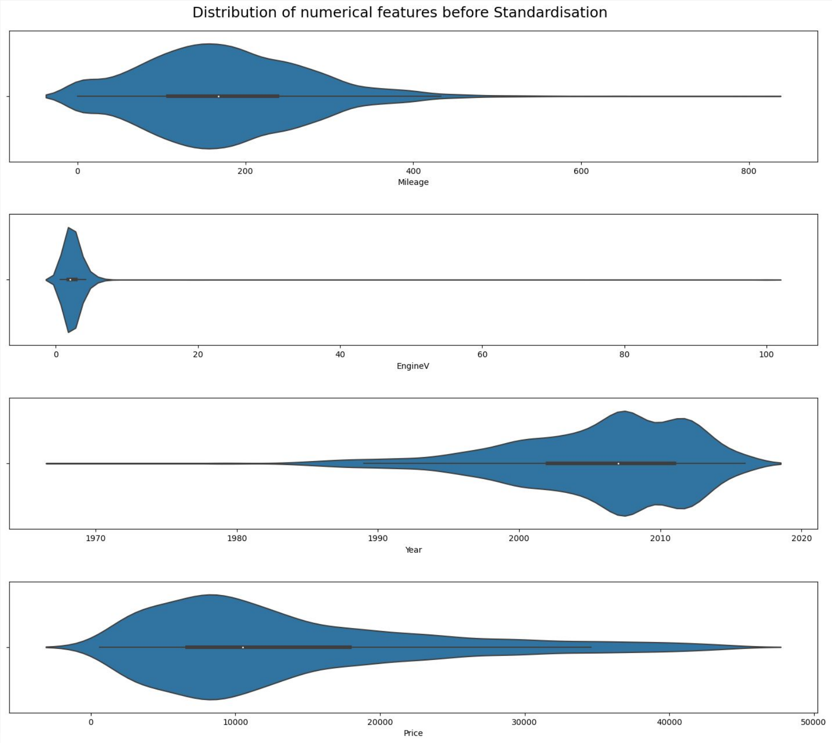
Registration
1
0

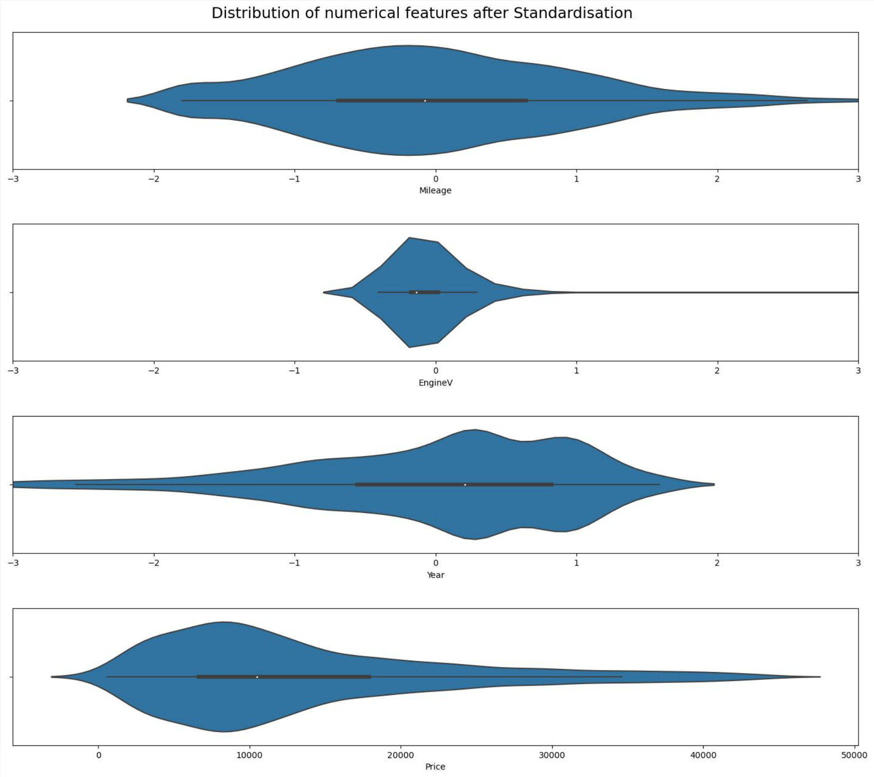
Example of Label encoding for Registration

FEATURE ENGINEERING

Scaling ***Numeric*** data

Using StandardScaler( ) from sklearn to scale numerical features





DATA PRE-PROCESSING

MACHINE LEARNING

Problem Type

- Reponse: Price, (**continuous** variable)

⇒ Regression problem

Models Used

- Linear Regression
- Lasso
- Elastic Net
- Ridge Regression

Performance Measure

- R-squared (R^2) score
- Mean Squared Error (MSE) score
- Root Mean Squared Error (RMSE)

1. Linear Regression Model

- ➔ A regression model used to predict continuous numerical values (car price) based on one or more independent feature.
- ➔ Finds linear relationship between the independent variables and the dependent variable (car price)
- ➔ Estimates the values of the coefficients that multiply each independent variable, such that the sum of the product of these coefficients and independent variables, along with an intercept term, results in the predicted value of the dependent variable (car price).

R^2	MSE	RMSE
0.8417	14073181	3751

Training

R^2	MSE	RMSE
-4.2488e+20	4.3812	209314725481944

Testing

2. Lasso Model

- ➔ A regression model that can be used for predicting car prices based on different factors.
- ➔ Finds a linear relationship between the independent variables and the dependent variable (i.e. car price)
- ➔ Minimizes the sum of the squared errors between the predicted and actual values As well as adding a penalty term to the loss function (multiple of the sum of the absolute values of the coefficients). This penalty term encourages the model to keep only the important features and reduce the effect of irrelevant features.

R^2	MSE	RMSE
0.8384	14366139	3790

Training

R^2	MSE	RMSE
0.7909	21559213	4643

Testing

3. Gradient Boosting Regressor Model

- A regression model that uses an ensemble method that combines multiple decision trees to form a strong predictive model.
- The algorithm works by iteratively adding decision trees to the model, each one correcting the errors of the previous tree, hence improving the predictions of the previous trees.

R^2	MSE	RMSE
0.9086	8131409	2852

Training

R^2	MSE	RMSE
0.8757	12821857	3581

Testing

4. Ridge Regression Model

- A regression model that identifies the most important factors for predicting the car prices and to estimate the effect of each factor on the car prices.
- Minimizes the sum of the squared errors between the predicted and actual values As well as adding a penalty term to the loss function (multiple of the sum of the absolute values of the coefficients). This penalty term encourages the model to keep only the important features and reduce the effect of irrelevant features.

R^2	MSE	RMSE
0.8336	14792064	3846

Training

R^2	MSE	RMSE
0.7912	21527686	4640

Testing

Tuning Hyperparameters with **GridSearchCV**

Hyperparameters for different models:

Linear Regression

```
{'copy_X': True, 'fit_intercept': True, 'n_jobs': None, 'positive': False}
```

Lasso

```
{'alpha': 1.0,
 'copy_X': True,
 'fit_intercept': True,
 'max_iter': 1000,
 'positive': False,
 'precompute': False,
 'random_state': 128,
 'selection': 'cyclic',
 'tol': 0.0001,
 'warm_start': False}
```

Ridge Regression

```
{'alpha': 1.0,
 'copy_X': True,
 'fit_intercept': True,
 'max_iter': None,
 'positive': False,
 'random_state': None,
 'solver': 'auto',
 'tol': 0.0001}
```

GBR

```
{'alpha': 0.9,
 'ccp_alpha': 0.0,
 'criterion': 'friedman_mse',
 'init': None,
 'learning_rate': 0.1,
 'loss': 'squared_error',
 'max_depth': 3,
 'max_features': None,
 'max_leaf_nodes': None,
 'min_impurity_decrease': 0.0,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 100,
 'n_iter_no_change': None,
 'random_state': None,
 'subsample': 1.0,
 'tol': 0.0001,
 'validation_fraction': 0.1,
 'verbose': 0,
 'warm_start': False}
```

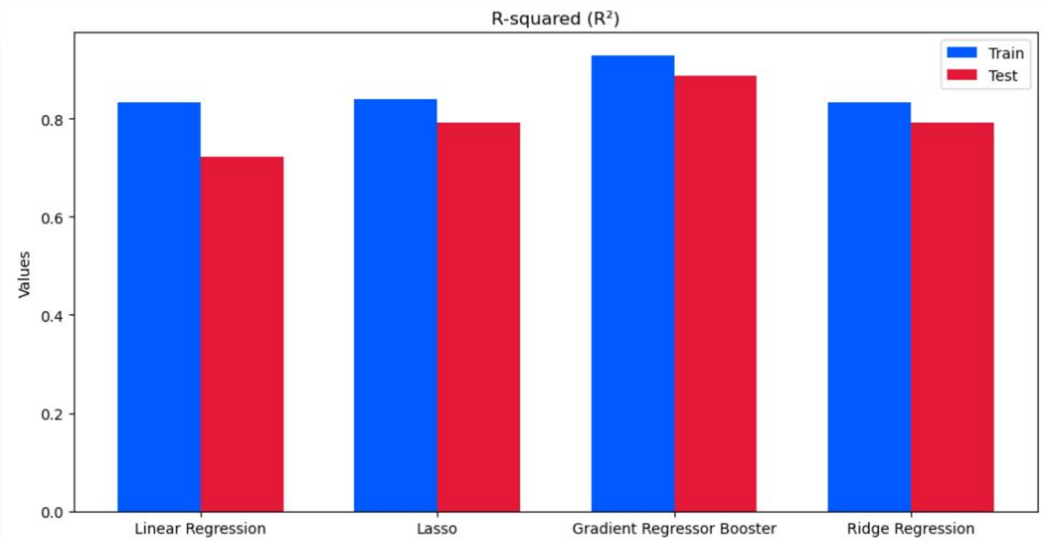
Re-training with Tuned Hyperparameters

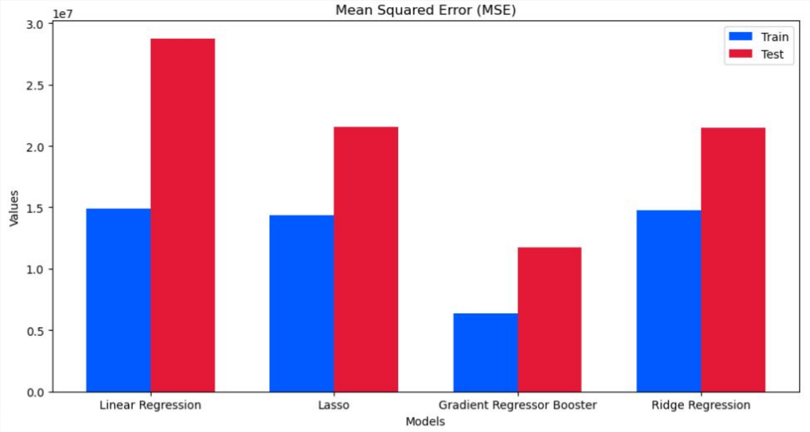
Train

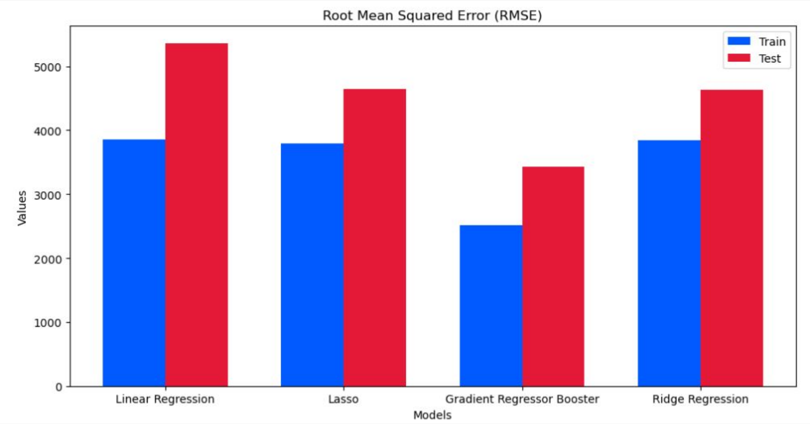
Model	R-squared (R^2)	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)
Linear regression	0.8328	14865860	3855
Lasso	0.8384	14366139	3790
Gradient Regressor Booster	0.9456	4833325	2198
Ridge Regression	0.8336	14792063	3846

Test

Model	R-squared (R^2)	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)
Linear regression	0.7211	28751536	5362
Lasso	0.7909	21559213	4643
Gradient Regressor Booster	0.8984	10472153	3236
Ridge Regression	0.7912	21527686	4639


R-squared (R^2)
Train
Test
Values
0.0
0.2
0.4
0.6
0.8
Linear Regression
Lasso
Gradient Regressor Booster
Ridge Regression


Mean Squared Error (MSE)
1e7
Train
Test
Values
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Linear Regression
Lasso
Gradient Regressor Booster
Ridge Regression
Models


Root Mean Squared Error (RMSE)
Train
Test
Values
0
1000
2000
3000
4000
5000
Linear Regression
Lasso
Gradient Regressor Booster
Ridge Regression
Models

CONCLUSION

Summary of Findings

- Successfully predicted car price based on its features at a high accuracy
- Gradient Boosting Regressor is the best available model

Limitations

- Overfitting
- Models fit to the noise present in training data
- Hence unable to generalise as well to new and unseen test data.

Improvements

- Ensemble methods (bootstrap aggregating, stacking)
- Trains multiple sub-models, combines sub-results, giving more accurate final answer

THANK YOU