CARS4YOU — ML PIPELINE (HANDOUT)
GROUP 42

The goal of this project is to build a **pipeline** able to **predict** a **car's price** based on its attributes (brand, model, year, fuel type, mileage, etc.), ensuring a consistent validation strategy.

EDA

1 Data Understanding

Explore the dataset; detect inconsistencies, outliers, variable relationships and key relationships to price.

2 Data Cleaning

Fix data issues and prepare categoricals for preprocessing (consistent categories/labels).

3 Data Visualization

Visualized univariate distributions (histograms, boxplots) and bivariate relationships (scatterplots, heatmaps) to identify trends, outliers, and correlations with price.

PRE-PROCESSING

1 Outliers

Keep ***engineSize*** (strong correlation with price); remove ***mpg*** and ***tax*** outliers; trim only extreme outliers for ***car_age*** and ***mileage***.

2 Missing Values

Numerical → KNN imputer by Brand (requires scaling)

Categorical → Mode imputation by Brand

<u>Apply the **same imputers/scalers from train** to validation.</u>

3 Encoding & Scaling

Brand → One-Hot Encoding

Model → Label Encoding within Brand

Other categoricals → One-Hot Encoding

<u>Fit the scaler only on train; transform validation/test with it.</u>

FEATURE SELECTION

1 Feature Selection

Correlation analysis + **Chi-Square** (for categoricals) + **Wrapper** methods (e.g., RFE) + **Embedded** approaches (e.g., Ridge or Tree-based models).

MODEL & EVALUATION

1 Model & Evaluation

Trained a **Random Forest Regressor** model using a **train/validation split (70/30)**. Evaluated performance using **R^2, MAE** and **RMSE** metrics to assess accuracy and generalization.

1. PREPROCESSING DECISIONS

Outliers (IQR)

Outliers were analysed by correlation with price.

- engineSize outliers were kept, as they carry valuable signal; removing them reduced model variance explanation.
- mpg and tax outliers were removed, given low relevance to price.
- car_age and mileage kept only extreme outliers trimmed to retain variation.

→ Decision based on maintaining information while limiting distortion.

Missing values

- Chosen method: KNN Imputer per Brand (for numericals) to leverage similarity between vehicles of the same brand.
- Mode imputation per Brand for categoricals, preserving realistic label proportions.

→ Brand-aware strategy produced more consistent values than global mean/mode imputation.

Encoding & scaling

- Brand encoded with One-Hot Encoding; Model encoded with LabelEncoder per brand column; other categoricals One-Hot Encoding.
- Scaling applied after splitting to prevent leakage (fit on train, transform on validation/test).

→ Ensures all variables are on comparable scale, crucial for linear algorithms.

2. FEATURE SELECTION — METHODS & OUTCOME

Approach

Three complementary methods were used:

1. Correlation — removed redundant numerical variables and identified strong relationships.
2. Chi-Square — quantified associations between categorical features and price.
3. RFE (Recursive Feature Elimination) — selected the most predictive subset using a linear estimator.

Result

- High-importance features: engineSize, year, mileage, and key categorical dummies (fuelType, transmission, Brand).
- Low-impact or noisy variables were dropped.

→ These steps improved model interpretability and reduced overfitting risk.

3. MODEL EVALUATION — PERFORMANCE & INTERPRETATION

Model used

A Random Forest regressor model trained on the processed dataset (70/30 train/validation).

→ Provides a good baseline for future comparisons.

Metric	Train	Validation
R^2	0.9898	0.9306
RMSE	546.25	1463.64
MAE	985.29	2543.34

Interpretation

- The model shows solid baseline performance with consistent behaviour across splits.
- The Random Forest produces stable and pattern-free residuals, consistent with a well-calibrated non-linear model.

4. KEY INSIGHTS & FUTURE IMPROVEMENTS

Insights gained

- Brand-specific preprocessing (KNN + mode) significantly enhanced data quality.
- The strongest predictors of price are engineSize, year, and mileage.
- Proper preprocessing order (scaling, encoding after split) was critical to avoid data leakage.
- The pipeline achieved stable results with minimal variance across datasets.

Next steps

- Apply cross-validation to ensure generalization.
- Compare regularized linear models (Ridge, Lasso) for improved bias-variance trade-off.
- Package preprocessing + model using Pipeline() for streamlined deployment.