Project name : Car Price Prediction

By

Md. Ariful Islam
United international University
Ha-Meem Group-IT
arifulislamhabib@gmail.com

Supervised by

Reja E Rabbi Tonmoy

Machine Learning Engineer, Pathao

Submitted to

Byte to Code

Submission date: 12-12-2025

Objective: Predict car prices (msrp) from car features (make, model, year, mileage, engine, fuel_type, transmission, etc.). Use EDA, feature engineering, robust modeling, and evaluation. Deliverables: cleaned dataset, trained models, evaluation metrics, and recommendations.

1. Prject Overview

- Problem type: Regression (predict continuous target msrp).
- Success metrics: Primary — Root Mean Squared Log Error (RMSLE) or RMSE on log1p(msrp); Secondary — MAE, R^2.
- Data assumptions: Typical columns: ‘engine_hp','engine_cylinders', 'highway_mpg', 'city_mpg', 'popularity'

- Missing values handling.
- Dtypes (convert to numeric where possible)
- Obvious anomalies (negative mileage, unrealistic years)

2. Exploratory Data Analysis (EDA)

3. Data Visualization:

4. Numeric feature analysis

- Correlation matrix (heatmap). Look at correlations with msrp.
- Scatterplots: mileage vs msrp, year vs msrp, engine_size vs msrp.
- Boxplots per year or per top brands to visualize spread/outliers.

5. Categorical feature analysis

- Top make by count and mean price.
- fuel_type and transmission price differences (boxplots).
- Frequency tables for rare categories — consider grouping.

6. Missing values and outliers

- Table: count and % missing per column.
- Strategy: Impute numerics with median; categorical with mode or new category "Unknown".
- For outliers: clip or remove entries beyond logical thresholds (e.g., msrp > 10000 depending on data).

---

7. Data Cleaning & Feature Engineering

- Lowercase and underscore column names: df.columns = df.columns.str.lower().str.replace(' ', '_').
- Convert types: df['year'] = df['year'].astype(int) where valid.
- Remove duplicates: df = df.drop_duplicates().

8. Handle missing values

- Numeric: df[num_cols] = df[num_cols].fillna(df[num_cols].median()).
- Categorical: df[cat_cols] = df[cat_cols].fillna('Unknown').

9. Target transform

- Use y = np.log1p(df['msrp']) for model stability.

10. Derived features

- age = current_year - year (or dataset year)
- mileage_per_year = mileage / (age + 1e-6)
- is_luxury = make.isin(['bmw','mercedes','audi','lexus']) (example)
- brand_model = make + '_' + model (useful but high-cardinality)

11. . Train/Validation/Test Split

Use reproducible shuffle-split by index:

12. . Model Evaluation

Report metrics on validation and final test set:

- RMSE on log1p(msrp) (primary)
- MAE and R^2
- If required, back-transform predictions: pred_price = np.expm1(pred_log) and compute RMSE/MAE on the original scale.

Provide error analysis:

- Residual plots vs year, mileage, make

Topics are covered in this project:

- Preparation data and do EDA (Exploratory Data Analysis).
- Use linear regression for predicting price.
- Understanding the internals of Linear Regression
- Evaluating the model with RMSE
- Feature Engineering
- Regularization