Smart Car Price Prediction – Linear Regression

Libraries Used

- pandas
- numpy
- matplotlib.pyplot
- seaborn
- sklearn.model_selection
- sklearn.linear_model
- sklearn.metrics
- sklearn.preprocessing

Dataset

- Dataset source: Kaggle – Car Price Prediction
- Contains features like Make, Model, Year, Transmission, Fuel Type, Mileage, and Engine Size.
- Target variable: Price

Preprocessing

Initial Approach: Label Encoding

- Used LabelEncoder for categorical features.
- Model performed decently but coefficient signs were illogical:
- For example, Make had a negative impact on price which didn't align with real-world expectations.

Fixed Approach: One-Hot Encoding

- Switched to pd.get_dummies() for One-Hot Encoding.
- Coefficients now made logical sense:
  Make ↓, Model ↑, Fuel Type ↑, Transmission ↓, Year ↑, Mileage ↓, Engine Size ↑
- Metrics:
  - MSE: 4,824,426
  - R^2 Score: 0.818

Feature Importance Analysis

- Visualized feature importance using coefficient magnitudes.
- Found Year, Engine Size, and Mileage had the strongest effect.

Iterative Feature Testing

Features Used	MSE	R^2 Score
Only Year	4,133,036	0.835
Year + Engine Size + Mileage	4,810,290	0.789
+ Model	4,689,777	0.843
+ Transmission	3,694,177	0.852
+ Make	4,381,645	0.849
+ Fuel Type	4,552,633	0.835
Final Features (Best combo):	4,645,040	0.833

<u>Note</u>: Although Make and Fuel Type showed high importance in the coefficient plot, they reduced performance and were therefore excluded from the final model.

Polynomial Regression (Tested and Discarded)

- Tried using polynomial features to model non-linear relationships.
- <u>Result</u>: Accuracy dropped. Polynomial features were not used further.

Final Model

- <u>Algorithm</u>: Linear Regression
- <u>Final features:</u> Year, Engine Size, Mileage, Model, Transmission
- <u>R^2 Score</u>: 0.843

Cross-Validation

- 5-Fold Cross-Validation results:
  R² scores: [0.8204, 0.8063, 0.8332, 0.8413, 0.8554]
  Average R²: 0.8313
- Indicates consistent and reliable performance across different data splits.

Conclusion

- One-Hot Encoding improved model interpretability and accuracy.
- Year, Engine Size, and Mileage are the key predictors.
- Final model achieves over 83% R^2 score with strong generalization.
- Dropping less effective features (despite visual importance) improved model performance.

Visualization