Regression Analysis of Used Car Prices

Tancredi Bosi

Alma Mater Studiorum Bologna

November 15, 2024

Outline

Problem Definition

- Predicting the price of used cars based on various features
- Data sourced from Kaggle competition

Figure: Kaggle competition image

Dataset Overview

- Dataset details:
  - 188,533 rows, 12 features, and 1 target column (price)
  - Numerical features: *id, model_year, milage*
  - Categorical features: *brand, model, fuel_type, engine, transmission, ext_col, int_col, accident, clean_title*

```
RangeIndex: 188533 entries, 0 to 188532
Data columns (total 13 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   id            188533 non-null  int64
 1   brand         188533 non-null  object
 2   model         188533 non-null  object
 3   model_year    188533 non-null  int64
 4   milage        188533 non-null  int64
 5   fuel_type     183450 non-null  object
 6   engine        188533 non-null  object
 7   transmission  188533 non-null  object
 8   ext_col       188533 non-null  object
 9   int_col       188533 non-null  object
 10  accident      186081 non-null  object
 11  clean_title   167114 non-null  object
 12  price         188533 non-null  int64
dtypes: int64(4), object(9)
memory usage: 18.7+ MB
```

Figure: Dataset info

Data Exploration

	id	brand	model	model_year	milage	fuel_type	engine	transmission	ext_col	int_col	accident	clean_title	price
0	0	MINI	Cooper S Base	2007	213000	Gasoline	172.0HP 1.6L 4 Cylinder Engine Gasoline Fuel	A/T	Yellow	Gray	None reported	Yes	4200
1	1	Lincoln	LS V8	2002	143250	Gasoline	252.0HP 3.9L 8 Cylinder Engine Gasoline Fuel	A/T	Silver	Beige	At least 1 accident or damage reported	Yes	4999
2	2	Chevrolet	Silverado 2500 LT	2002	136731	E85 Flex Fuel	320.0HP 5.3L 8 Cylinder Engine Flex Fuel Capab...	A/T	Blue	Gray	None reported	Yes	13900
3	3	Genesis	G90 5.0 Ultimate	2017	19500	Gasoline	420.0HP 5.0L 8 Cylinder Engine Gasoline Fuel	Transmission w/Dual Shift Mode	Black	Black	None reported	Yes	45000
4	4	Mercedes-Benz	Metris Base	2021	7388	Gasoline	208.0HP 2.0L 4 Cylinder Engine Gasoline Fuel	7-Speed A/T	Black	Beige	None reported	Yes	97500

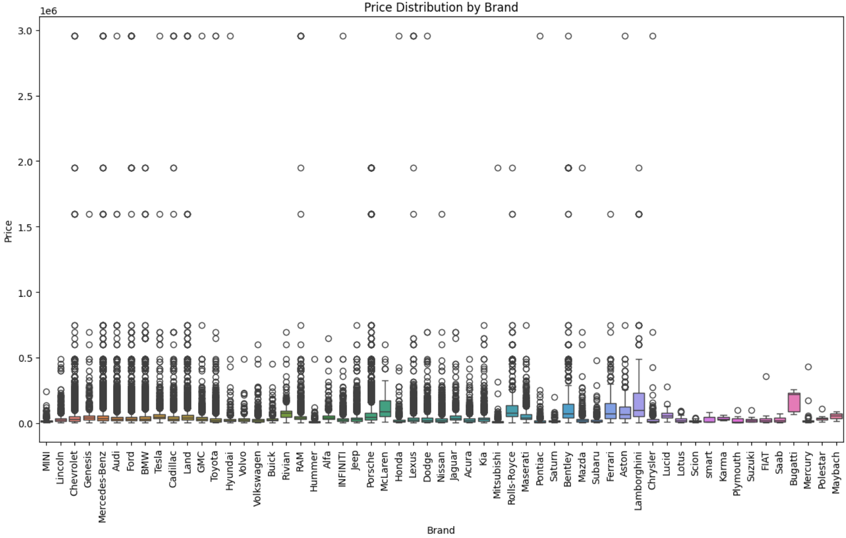
Figure: Head of the dataset

We can already see that:

- ”id” column can be dropped as it refers only to the index of the car.
- ”brand” and ”model” columns seem to have a lot of different unique values.
- ”engine” column has useful and different information abridged in one string.
- ”ext_col” and ”int_col” columns are the colors of the cars and they may be not so useful.

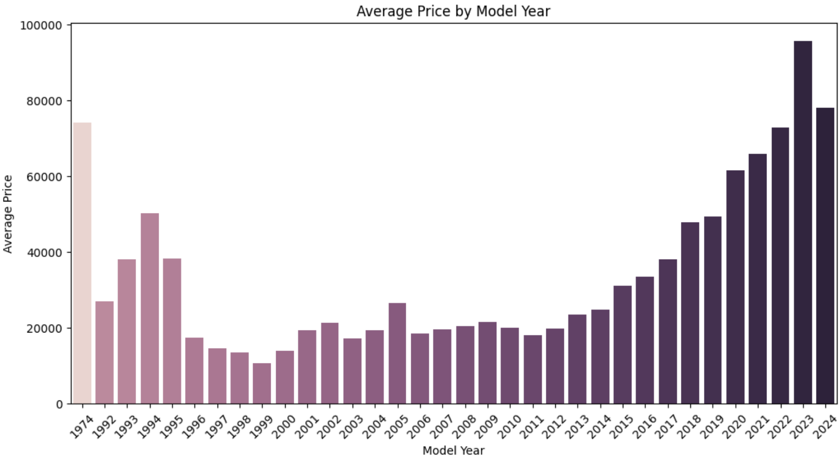
Data Visualization

- Price distribution by brand:



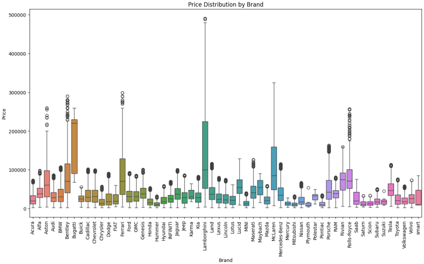
Data Visualization

- Average price by model year:



Data Preprocessing

- Remove outliers



- Extract *vehicle_age* as $vehicle\_age = 2024 - model\_year$
- Extract *HP*, *engine_size* and *cylinders* from *engine*
- Extract *speed* and *transmission_type* from *transmission*
- Extract *luxury_brand* from *brand* and *model_category* from *model*, to reduce the unique values in the two columns

Data Preprocessing

- Fill missing values with 'Unknown' or 0
- Remove *id, ext_col, model* and *int_col* columns
- Scale *milage, vehicle_age, HP* and *engine_size* with **RobustScaler()**
- Enconde:
  - *accident* in 0/1
  - *speed* in numerical values
  - *transmission_type* in 0/1
  - *clean_title* in 0/1
  - *fuel_type, luxury_brand* and *model_category* with One-Hot Encoding

The final features for each sample are:

```
['milage', 'accident', 'clean_title', 'price', 'vehicle_age', 'HP', 'engine_size', 'cylinders']
['speed', 'transmission_type', 'luxury_brand_1', 'luxury_brand_2', 'fuel_type_1', 'fuel_type_2']
['fuel_type_3', 'fuel_type_4', 'model_category_Luxury', 'model_category_Other', 'model_category_Sport']
```

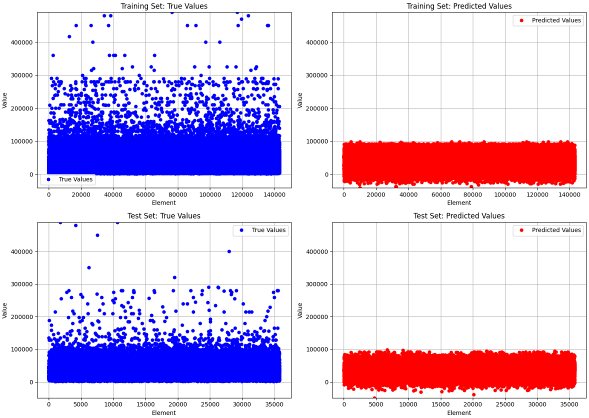
Model Selection

- Dataset division: 80% training set, 20% test set.
- Measures in output: Train-RMSE, Test-RMSE
- Models considered:
  - Ridge Regressor (least squares with l2 regularization)
  - Random Forest Regressor with standard hyperparameters
  - Support Vector Regressor
  - Random Forest Regressor with Grid Seach
  - MLP Regressor
  - AdaBoost Regressor with Decision Tree
  - AdaBoost Regressor with Random Forest Regressor

Ridge Regressor

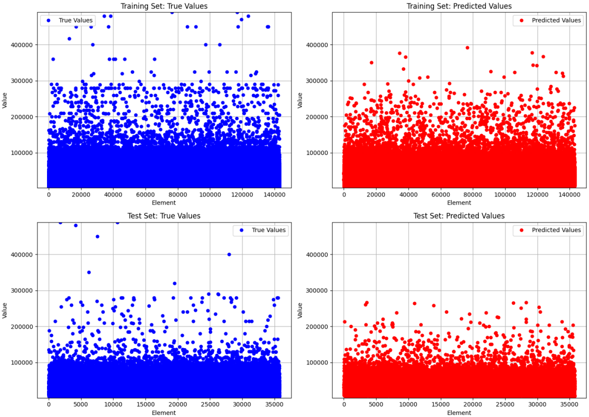
Ridge() performance:

- Train RMSE: 19192
- Test RMSE: 18877



RandomForestRegressor() performance:

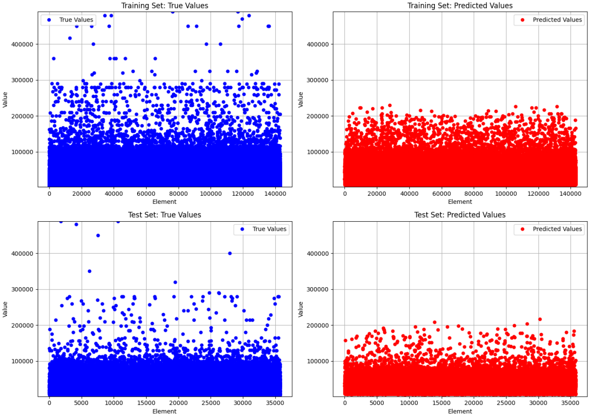
- Train RMSE: 8134
- Test RMSE: 17625



Grid Search Random Forest Regressor

RandomForestRegressor(n_estimators=100, max_depth= 12, min_samples_split= 14, min_samples_leaf= 3**)** performance:

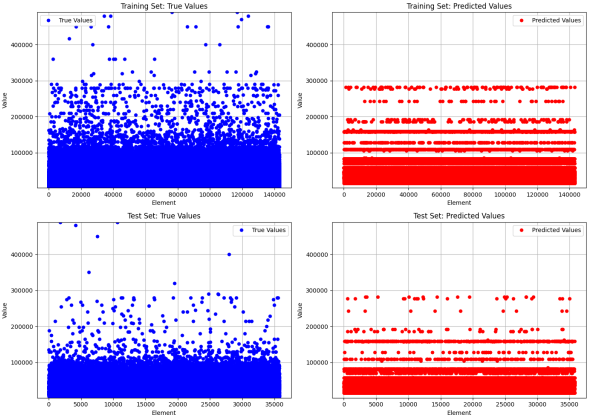
- Train RMSE: 15129
- Test RMSE: 16518



AdaBoost Regressor - DT

AdaBoostRegressor() performance:

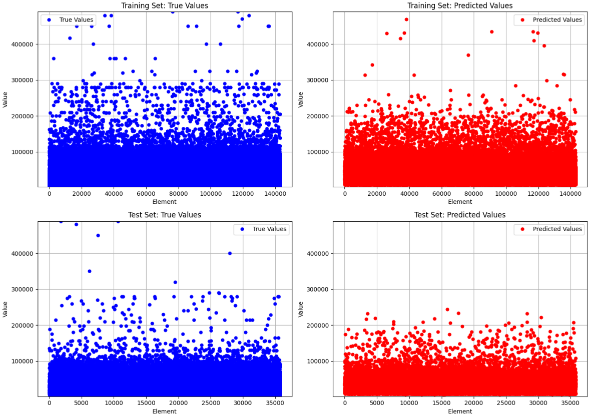
- Train RMSE: 19941
- Test RMSE: 19990



AdaBoost Regressor - RF

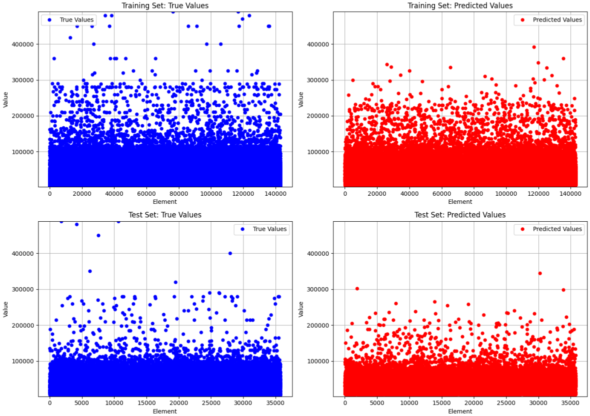
AdaBoostRegressor(estimator=RandomForestRegressor()**)** (with grid-search hyperparameters) performance:

- Train RMSE: 14432
- Test RMSE: 16710



MLPRegressor(hidden_layer_sizes=(128, 256, 512, 256, 128), max_iter=1000, learning_rate='adaptive'**)** performance:

- Train RMSE: 14572
- Test RMSE: 17955



Results Conclusions

Here the models for a comparison:

Model	Train RMSE	Test RMSE
Ridge Regressor	19192	18877
Random Forest Regressor	8134	17625
Random Forest Regressor GS	15129	16518
Ada Boost Regressor DT	19941	19990
Ada Boost Regressor RF	14432	16710
MLP Regressor	14572	17955

Table: Model performance comparison