# Used Car Price Predictions

Great Learning
POWER AHEAD

AUGUST 2021

**MIT – APPLIED DATA SCIENCE PROGRAM**

**Capstone Project**

**Kevin Goh**

# INTRODUCTION

There is a huge demand for used cars in the Indian Market today. As sales of new cars have slowed down in the recent past, the pre-owned car market has continued to grow over the past few years and is now larger than the new car market. Cars4U is a budding tech start-up that aims to find footholes in this market.

In 2018-19, while new car sales were recorded at 3.6 million units, around 4 million second-hand cars were bought and sold. There is a slowdown in new car sales and that could mean that the demand is shifting towards the pre-owned market. In fact, some car owners replace their old vehicles with pre-owned cars instead of buying a new automobile.

Unlike new cars, where price and supply are fairly deterministic and managed by OEMs (Original Equipment Manufacturer / except for dealership level discounts which come into play only in the last stage of the customer journey), the used car market is a very different beast, with large uncertainties in both pricing and supply. Several factors, including mileage, brand, model, year, etc. can influence the actual worth of a car. From the perspective of a seller, it is not an easy task to set the correct price of a used car. Keeping this in mind, the pricing scheme of these used cars becomes important in order to grow in the market.

# EXECUTIVE SUMMARY

Data of used car sales from the Indian market was studied and used to develop a machine learning model to predict used car prices. The problem objective was defined as: come up with a pricing model that can effectively predict the price of used cars and can help the business in devising profitable strategies using differential pricing.

***Objective: Come up with a pricing model that can effectively predict the price of used cars and can help the business in devising profitable strategies using differential pricing.***

## KEY INSIGHTS

**INSIGHT 1 - Important Features**: All the models used indicated that ***year***, ***power*** and ***mileage*** were important features. It was also found that ***engine***, ***power*** and ***mileage*** have high correlation between them. The linear regression model indicated to remove ***engine*** due to the presence of multicollinearity. The linear regression, decision tree and random forest models included the additional feature of ***log_kilometers_driven***.

**INSIGHT 2 - Problem with the Brand and Model features**: The study revealed issues with the dataset, particularly the feature ***name***. Because of the large number of brands and models, and with the added fact that there were many models and brands with only 1 data point, it led to data sparsity and hence added to the model complexity. This meant that the linear regression model could not work well with the full 262 features that were created.

**INSIGHT 3 - Best Overall Model:** The tuned random forest model performed the best overall based on the scores of $R^2$, MAE and RMSE. The decision tree was the second best followed by the linear regression model.

# KEY FINDINGS

## 1. EXPLORATORY DATA ANALYSIS

The dataset consists of used cars from 1996 to 2019, from 11 cities across India, which contained other features such as ***name*** (of car which includes brand and model names), ***kilometers_driven***, ***fuel_type*** (diesel, petrol, CNG, LPG or electric), ***transmission*** (manual or automatic), ***owner*** (first, second, third or fourth and above), ***mileage*** (in km/liter or km/kg), ***engine*** (displacement volume in cc), ***power*** (maximum engine power in BHP), ***seats*** (number of seats in the car), ***new_price*** (price of a new car of same model in INR 100,000), and ***price*** (price of the used car in INR 100,000).

There was a total of 7253 data points, with 14 features in total. All data types for the respective features were found to be correct. However, the dataset was found to have missing values for ***mileage***, ***engine***, ***power***, ***seats***, ***new_price*** and ***price***. Two features, ***new_price*** and ***price***, were found to have significantly more missing values than other features.

### 1.1 UNIVARIATE ANALYSIS

**Numerical Variables**: Description of data showed outliers and possible errors of data.

| Variable | Observations | Data Treatment |
|---|---|---|
| *S.No.* | ▪ Did not provide useful information | ▪ Removed feature |
| *Kilometers_Driven* | ▪ Heavily right-skewed<br>▪ Unlikely value of 6.5 million km | ▪ Performed Log-transformation<br>▪ Removed data point |
| *Mileage* | ▪ Impossible value of 0.0 km/liter<br>▪ 2 missing values | ▪ Imputed median values according to fuel type of car for cases where mileage was 0.0<br>▪ Missing values were found to be from electric cars and were removed[1] |
| *Engine* | ▪ 46 missing values | ▪ Imputed median values |
| *Seats* | ▪ 53 missing values | ▪ Imputed median values according to car model |
| *Power* | ▪ 175 missing values | ▪ Imputed median values |
| *New_price* | ▪ 6247 missing values | ▪ Removed feature[2] |
| *Price* | ▪ 1234 missing values | ▪ Removed data points with null price values |

[1] There were only 2 electric cars in the data set and there was no median value to impute.
[2] Due to the large proportion of missing values (6247 out of 7253), imputing is not expected to be accurate.

**Categorical Variables**: There were no missing values. There is a high number of unique car names as there are 2041 unique names out of 7253. The electric fuel type was removed as there were only 2 data points.

| Variable | Observations | Data Treatment |
|---|---|---|
| *Name* | ▪ The names each consist of the brand name followed by the model name<br>▪ High number of unique car names of 2041 out of 7253 | ▪ Two features were created: *brand* and *model*<br>▪ Name was dropped after creation of *brand* and *model*<br>▪ One-hot encoding |
| *Location* | ▪ 11 unique locations<br>▪ Most frequent: Mumbai<br>▪ Relatively evenly distributed | ▪ One-hot encoding |
| *Fuel_Type* | ▪ 5 unique fuel types<br>▪ Most frequent: Diesel<br>▪ 98.9% either diesel or petrol | ▪ One-hot encoding |
| *Transmission* | ▪ 2 transmission types<br>▪ Most frequent: Manual (71.8%) | ▪ One-hot encoding |
| *Owner_Type* | ▪ 4 owner types<br>▪ Most frequent: First (82.1%) | ▪ One-hot encoding |

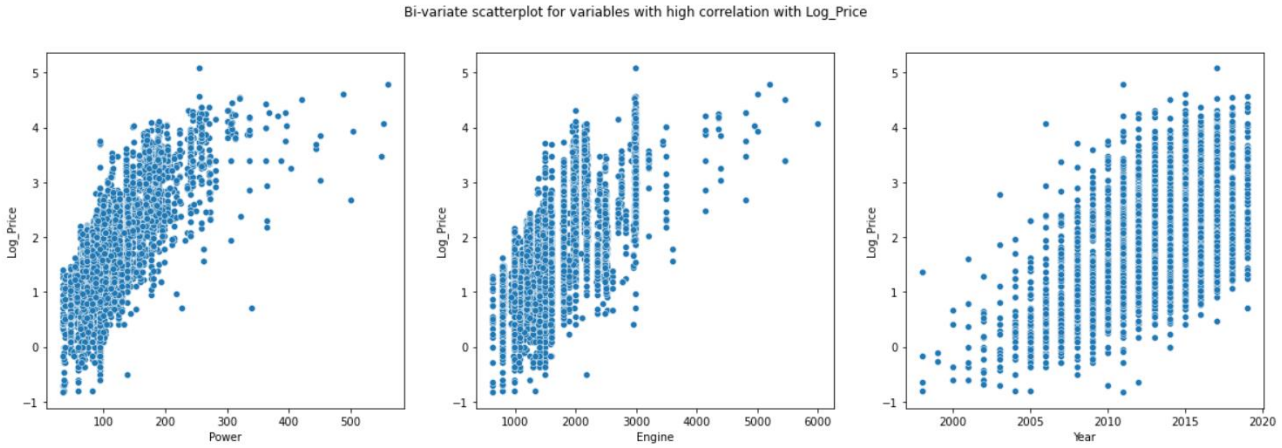
## 1.2 BIVARIATE ANALYSIS

The pairplot reveals the following correlations which have absolute correlation value >0.3. Those relationships with an astrerisk (*) are expected as they are due to the log-transformation.

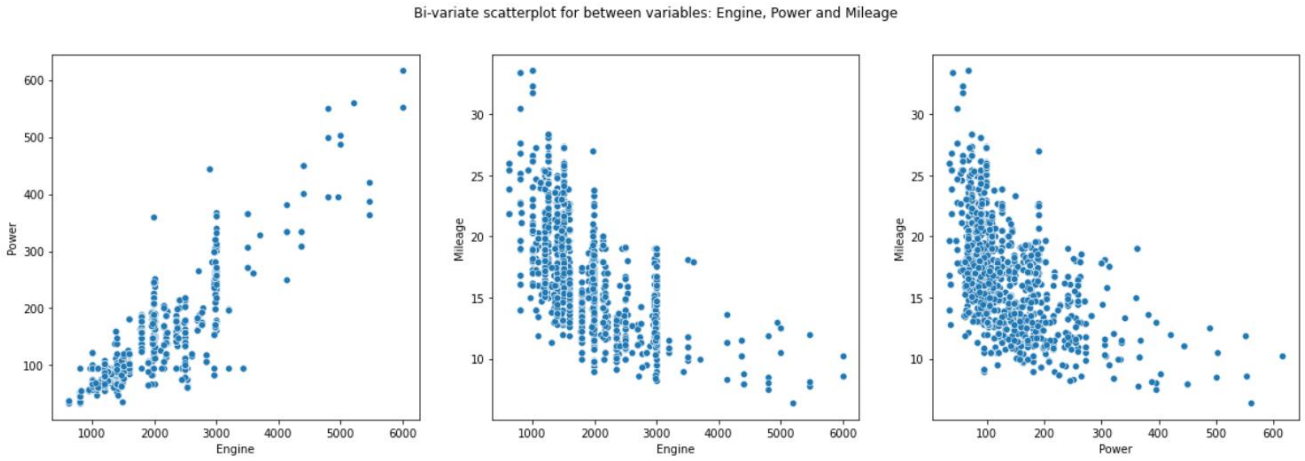
| Postive correlation | Negative correlation |
|---|---|
| Power and New_Price (0.88) | Mileage and Engine (-0.59) |
| New_Price and **Price** (0.87) | Mileage and Power (-0.59) |
| Engine and Power (0.86) | Year and Kilometers_Driven (-0.45) |
| Price and Log_Price (0.85)* | Mileage and New_Price (-0.38) |
| Km_Driven and Log_Km_Driven (0.82)* | Mileage and **Price** (-0.31) |
| New_Price and **Log_Price** (0.78) | Mileage and Seats (-0.31) |
| Power and **Price** (0.77) | |
| Power and **Log_Price** (0.77) | |
| Engine and New_Price (0.74) | |
| Engine and **Log_Price** (0.69) | |
| Engine and Price (0.66) | |
| Year and **Log_Price** (0.5) | |
| Engine and Seats (0.4) | |
| Year and Mileage (0.32) | |
| Year and **Price** (0.31) | |

The relationships which include either ***price*** or ***log_price*** are highlighted in green. From the correlation values, it is observed that the features are positively correlated with ***log_price***: ***new_price***, ***power***,

***engine*** and ***year***. On the other hand, ***mileage*** is negatively correlated with ***price***. The scatter plot for ***power***, ***engine*** and ***year*** against ***log_price*** is shown below.




It was also noticed that there were strong correlations between the variables highlighted in purple: ***engine***, ***power*** and ***mileage***. These correlations are expected as cars of higher engine displacement tend to have higher power. Also, cars of higher engine displacement are expected to have lower mileage and likewise for cars of higher power. The scatter plot for the three relations are shown below. The high correlation between the 3 variables suggest presence of multi-collinearity.




# 2. OUTPUT VARIABLE

The output variable ***Price*** was found to be heavily right-skewed, and therefore a log transformation was performed. Comparing ***log_price*** to ***price*** as the dependent variable, the linear regression model using ***log_price*** had higher $r^2$ value and lower MAE and RMSE. Hence, ***log_price*** was used as the output variable for the machine learning models.

# 3. BRAND AND MODEL NAMES

Car names were split into brand names and model names. It was observed that there was a large variation in number of brand names as there were a total of 4 brand names that exceeded a count of 500: Maruti, Hyundai, Honda, and Toyota. On the other extreme, there were a total of 5 brand names with a count of one each: Smart, Ambassador, Lamborghini, Hindustan and Opelcorsa.

Model names also had large variation and there were a total of 32 models which only had a count of one. The use of model names as features may not be preferable as it may cause the machine learning model to be overly complex, which while causes model bias to be the variance and error of the model will be high.

A linear regression model was trained on training data and tested on test data, using a partition ratio of 70:30. It was observed that the model did not perform well on test data which was likely due to the model complexity and the existence of high sparsity of columns in brand and model columns. This led to the need for feature selection to reduce model complexity.

# 4. FEATURE SELECTION USING LASSO REGRESSION

Lasso Regression was used in order to identify coefficients of features which were zero. Using LassoCV, it was found that there were only three non-zero features: ***year***, ***power*** and ***engine***. It suggested that features such as model names may be dropped to reduce model complexity.

# 5. LINEAR REGRESSION MODEL

After completion of missing value treatment there were a total of 6016 data points. With one-hot encoding, but before the removal of model names, there were a total of 262 features. After removing model names, the number of features was reduced to 52. The dataset was partitioned into training and test data on a ratio of 70:30 and using ***log_price*** as the output variable. A linear regression model was fitted on training data and the model achieved a high $R^2$ score of **0.923**. Features were iteratively removed using hypothesis testing with p-value of 0.05. After completion of two iterations, there were no further insignificant features.

## 5.1 MODEL ASSUMPTIONS

Using Variance Inflation Factor (VIF), it was found that there was multicollinearity among the independent variables. The feature ***Engine*** was removed as it had the highest VIF score. This finding also agrees with the earlier correlation findings in section 1.2 on bivariate analysis. After removal, the model was re-trained and the final model had neither insignificant features nor presence of

multicollinearity. The final model, ***ols_res_3***, achieved a high $R^2$ score of **0.909** with a low MAE of **0.195** and low RMSE of **0.265** on training data.

Remaining assumptions were checked: mean of residuals, normality of residuals, test for linearity and homoskedasticity. It was found that all assumptions for linear regression were satisfied except for homoskedasticity. This indicates that the linear model may not be the best and this gave rise to the need to explore other non-linear models such as decision trees and random forests.

### 5.2 PERFORMANCE ON TEST DATA

The model was tested on test data and achieved a high $R^2$ score of **0.844** with a low MAE of **0.261** and low RMSE of **0.341**. In addition, the $R^2$ score on the cross validation is **0.907** which is almost similar to the $R^2$ score on the training dataset, which indicates that the model is neither over or under fitted.

| | $R^2$ | MAE | RMSE |
|---|---|---|---|
| Linear Regression (on training data) | 0.909 | 0.195 | 0.265 |
| Linear Regression (on test data) | 0.844 | 0.261 | 0.341 |

### 5.3 FINAL MODEL FEATURES

The final model had the following features: ***Year***, ***Mileage***, ***Power***, ***Seats***, ***Log_Kilometers_Driven***. It also included 7 locations, the diesel fuel type, manual transmission, 2 owner types, and 7 brands.

## 6. DECISION TREE REGRESSOR

A decision tree was trained on the dataset which included model names, in order to assess if the model would be able to handle the large number of features of 262. Again, the dataset was partitioned into training and test data on a ratio of 70:30 and using ***log_price*** as the output variable. It was found that the decision tree was overfitting the training data as it had a high $R^2$ score of **1.000** and close to zero errors. In comparison, on the test data, the decision tree had a lower $R^2$ score of **0.887**, MAE of 0.198 and RMSE of 0.291. A total of There was a need to perform hyperparameter tuning of the decision tree to avoid overfitting.

| | $R^2$ | MAE | RMSE | Features |
|---|---|---|---|---|
| Decision Tree (on training data) | 1.000 | 0.000 | 0.002 | 170 |
| Decision Tree (on test data) | 0.887 | 0.198 | 0.291 | 170 |

GridSearchCV was used to tune the decision tree and a tuned decision tree, dtree_tuned, had a max depth of 12, with improved scores on the test data.

| | $R^2$ | MAE | RMSE | Features |
|---|---|---|---|---|
| Dtree_tuned (on training data) | 0.976 | 0.090 | 0.137 | 123 |
| Dtree_tuned (on test data) | 0.889 | 0.196 | 0.288 | 123 |

The tuned model reduced the number of features to 123 from 170. The top 5 features are: ***power***, ***year***, ***engine***, ***log_kilometers_driven***, and ***mileage***. While the decision tree performed well on the training data, it did not perform as well on the test data and hence another model had to be explored.

# 7. RANDOM FOREST REGRESSOR

The random forest regressor was trained on the same dataset as the decision tree, with all 6016 data points and 262 features in total. Again, the dataset was partitioned into training and test data on a ratio of 70:30 and using ***log_price*** as the output variable. In comparison with the decision tree, the random forest performed better on test data. It was also noticed that it used more features of 256 instead of 170 in the decision tree.

| | $R^2$ | MAE | RMSE | Features |
|---|---|---|---|---|
| Random Forest (on training data) | 0.991 | 0.055 | 0.083 | 256 |
| Random Forest (on test data) | 0.943 | 0.144 | 0.206 | 256 |

Again, GridSearchCV was used to tune the random forest and a tuned decision tree, rf_tuned, had improved scores on the test data.

| | $R^2$ | MAE | RMSE | Features |
|---|---|---|---|---|
| Rf_tuned (on training data) | 0.991 | 0.055 | 0.081 | 256 |
| Rf_tuned (on test data) | 0.946 | 0.143 | 0.201 | 256 |

# 8. MODEL COMPARISONS

**Best Overall Model:** The best overall model was the tuned random forest, as it had the best overall performance on the test data and did well on the training data as well. The model rf_tuned also had a cross validation score of $R^2$ of **0.935**.

**Common Features**: All models in this study included the following four:

1. Year
2. Power
3. Mileage
4. Log_Kilometers_Driven

| | $R^2$ | MAE | RMSE | Features |
|---|---|---|---|---|
| **TRAINING** | | | | |
| Linear Regression (on training data) | 0.909 | 0.195 | 0.265 | 23 |
| Decision Tree (on training data) | 1.000 | 0.000 | 0.002 | 170 |
| Dtree_tuned (on training data) | 0.976 | 0.090 | 0.137 | 123 |
| Random Forest (on training data) | 0.991 | 0.055 | 0.083 | 256 |
| Rf_tuned (on training data) | 0.991 | 0.055 | 0.081 | 256 |
| **TESTING** | | | | |
| Linear Regression (on test data) | 0.844 | 0.261 | 0.341 | 23 |
| Decision Tree (on test data) | 0.887 | 0.198 | 0.291 | 170 |
| Dtree_tuned (on test data) | 0.889 | 0.196 | 0.288 | 123 |
| Random Forest (on test data) | 0.943 | 0.144 | 0.206 | 256 |
| Rf_tuned (on test data) | 0.946 | 0.143 | 0.201 | 256 |

# CONCLUSION

## DATA HARVESTING

There needs to be more data collected in order to build better models. This is to reduce brands and model names which have low counts. Also, while this study removed the feature ***New_Price*** due to the absence of a large proportion of its values, these data points should be collected and added to the existing data set for it to be complete. The same should be done for ***Price***, which had more than 1000 missing values.

## FURTHER MODEL REFINEMENT

The linear regression model can be further refined by performing additional hyperparameter tuning such as performing GridSearchCV to determine the optimal number of features. In addition, the brand and model names which only have a count of 1 can be removed to explore how the model performs. The random forest and decision tree models can also be further refined with hyperparameter tuning.

## DATA HANDLING

The study revealed issues with the dataset, particularly the handling of the feature ***name***. The first issue is the requirement to perform feature engineering such as splitting the brand names from the model names, which requires a data engineer to check through and validate the correctness of each brand name. The second issue is If data is keyed in manually, data validation methods such as drop-down lists may help to avoid errors.