PART1

Prices of new cars are relatively more stable compared to used cars as these prices are fixed by the manufacturer. Hence, when buying a new car, customers can be assured that their investment will be worthy.[1] However, Covid-19 had a detrimental impact in the automotive industry resulting from a tremendous fallback in demand during lockdown as well as travel restrictions. Customers become less capable to afford new cars due to the economic impact of Covid-19, ultimately leading to more demand in the used cars industry and contributing to a global increase in the market. Used cars are sold by various organizations such as independent car dealers, rental car companies and leasing offices. Cardekho is one of the biggest car search venture companies in India, buying & selling new and used cars in India.[2]

Global used car market is projected to reach $1,355.15b by 2027 from a valuation of $828.24b in 2019 exhibiting a CAGR of 8.3% within the period. This growth can be attributed to the rise of online sales channels such as Cardekho, as well as the shifting demand in used cars with the financial impact of Covid-19.[3]"However, the used car market has seen a huge jump in sales due to reasons such as fall in income, shortage in money, and increasing preference for private cars to maintain social distancing which is expected to affect the used car industry positively during the COVID-19 outbreak globally." [4]

Therefore, there is great value in generating an accurate price prediction system for used cars that effectively determines the worthiness of the car accounting a variety of factors

The decision-makers of this analysis include used car dealers; as they would be able to increase their sales by understanding the desirable features of the cars and offer better services. There are also individuals interested in buying a used car or selling their car. This analysis would be particularly insightful for these individuals as it would ensure that they don't pay too much or sell less than its market value.

The aim of this project is to build an algorithm that predicts the selling price of used cars. The dataset is a csv file imported from Kaggle; 'Car details v3.csv' [5]. It was uploaded in 2019 and is updated annually. The dataset contains information about used cars listed on www.cardekho.com. Each row/observation represents a different car type with corresponding features as columns. The analysis addresses the continuous target variable; selling price which is already available in the dataset.

---

[1] (Gokce, 2020)
[2] (Joshi, 2022)
[3] (Padalkar and Mutreja, 2021)
[4] (Padalkar and Mutreja, 2021)
[5] (Vehicle dataset, n.d.)

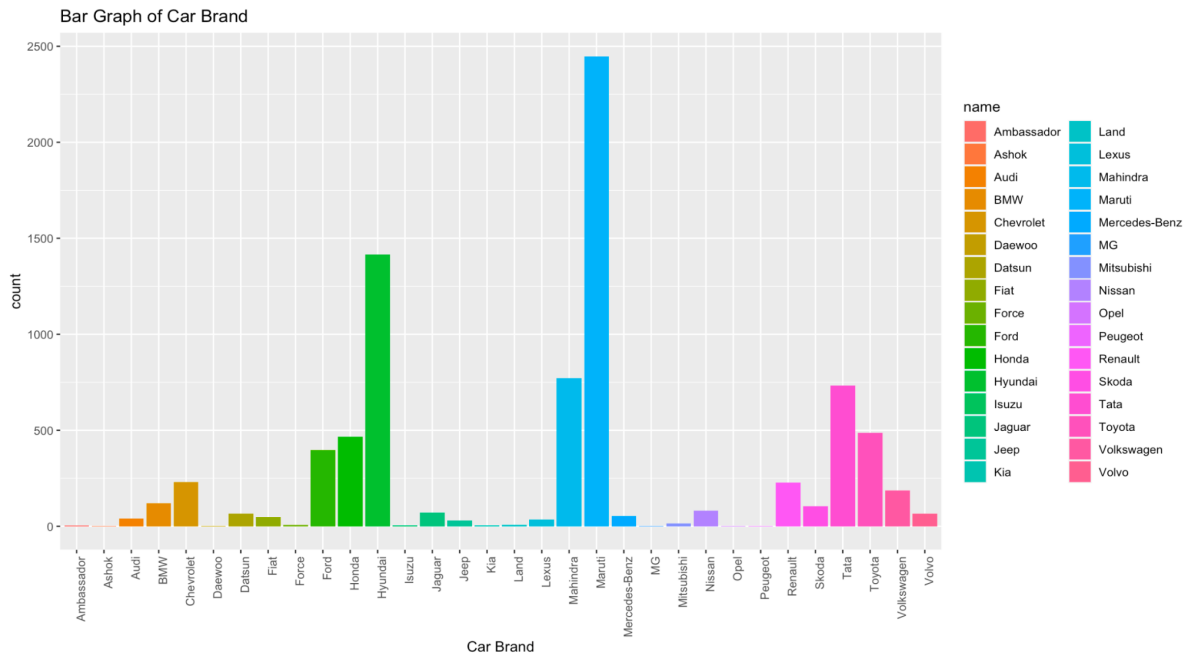
PART2

Initially the dataset contains 13 columns representing different features of the cars and 8128 rows representing unique observations for each car model. The data is unstructured as it consists of categorical variables represented as characters along with numerical variables.

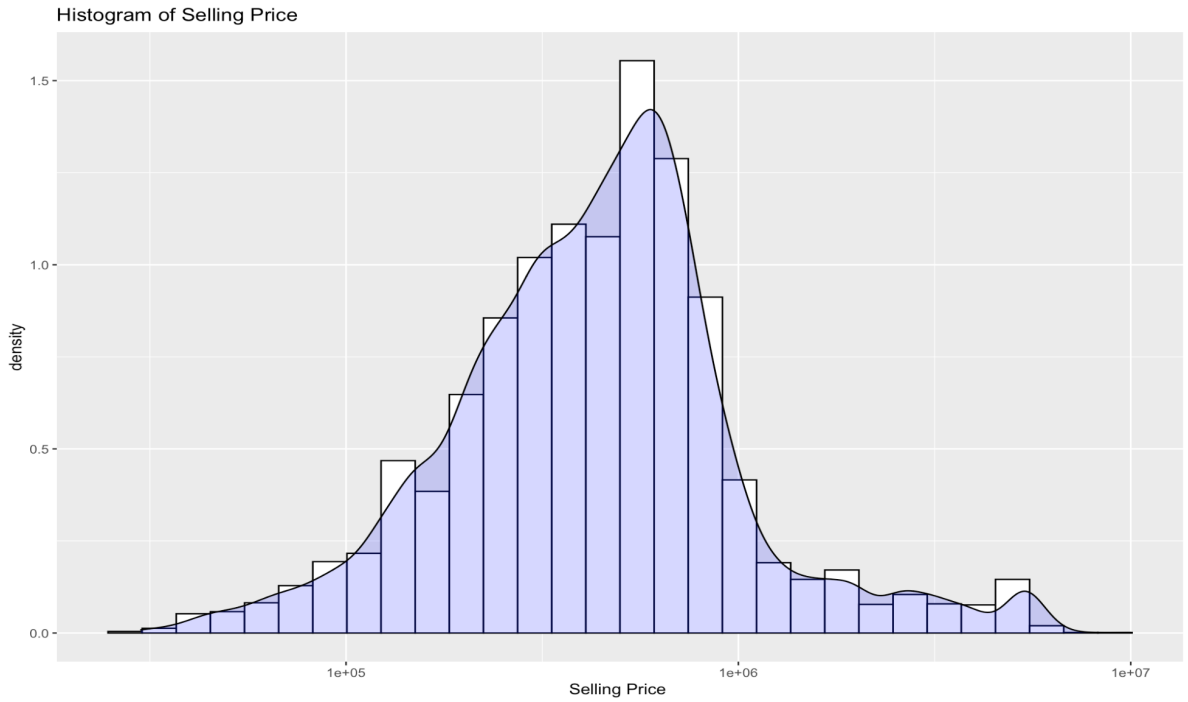
Data visualizations are employed to better understand how certain attributes are distributed/related with each other and with the target variable; selling price. Also as having highly correlated predictors can damage model performance, correlation tables are employed to observe the correlation between attributes. These visualizations help finalize the set of predictors.

R

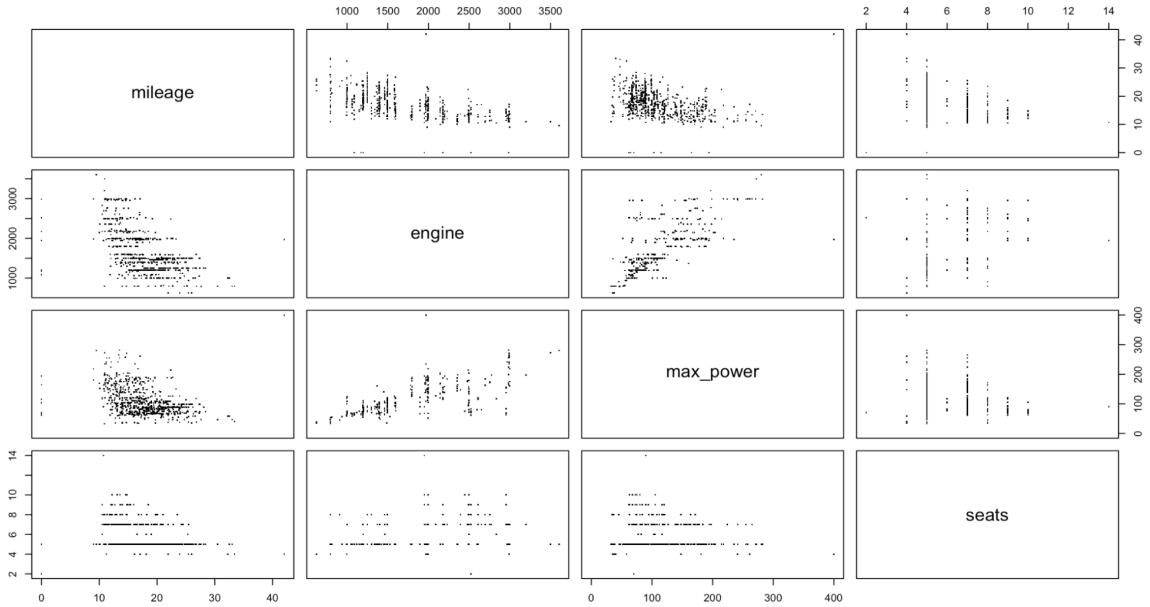
First, plot a bar graph of each brand over total number of observations. Highest numbers of cars are in Maruti brand followed by Hyundai, and Mahindra brands



To ensure that the distribution of the target variable is continuous, plot a histogram and observe that Selling Price follows a normal distribution.

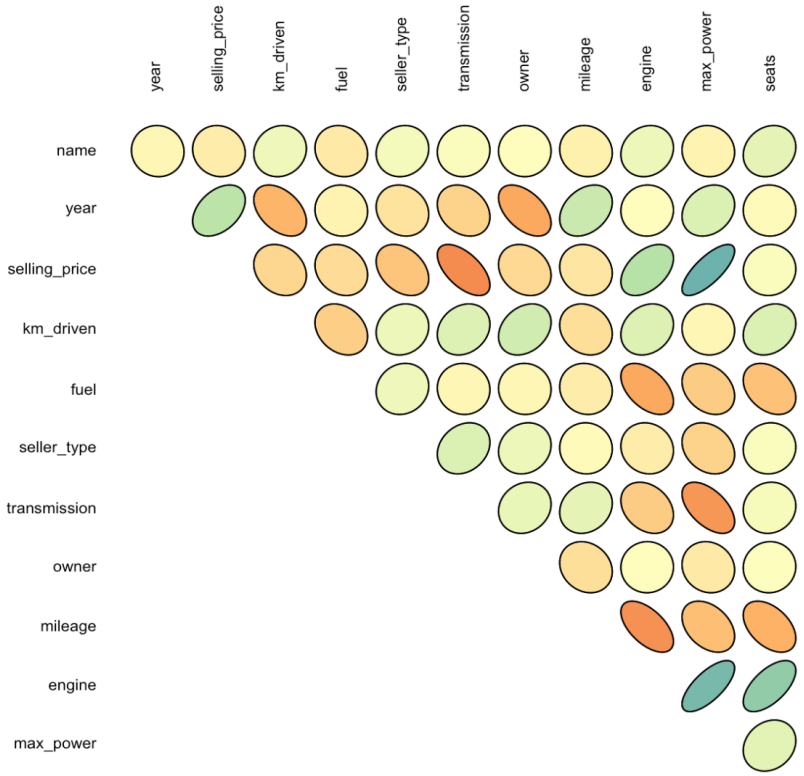


Then, to visualize how values are correlated for different pairs of attributes, generate a scatter plot between selected attributes; mileage, engine, max power and seats. While a negative correlation between mileage and max power is observed, a positive correlation between engine and max power is noted.

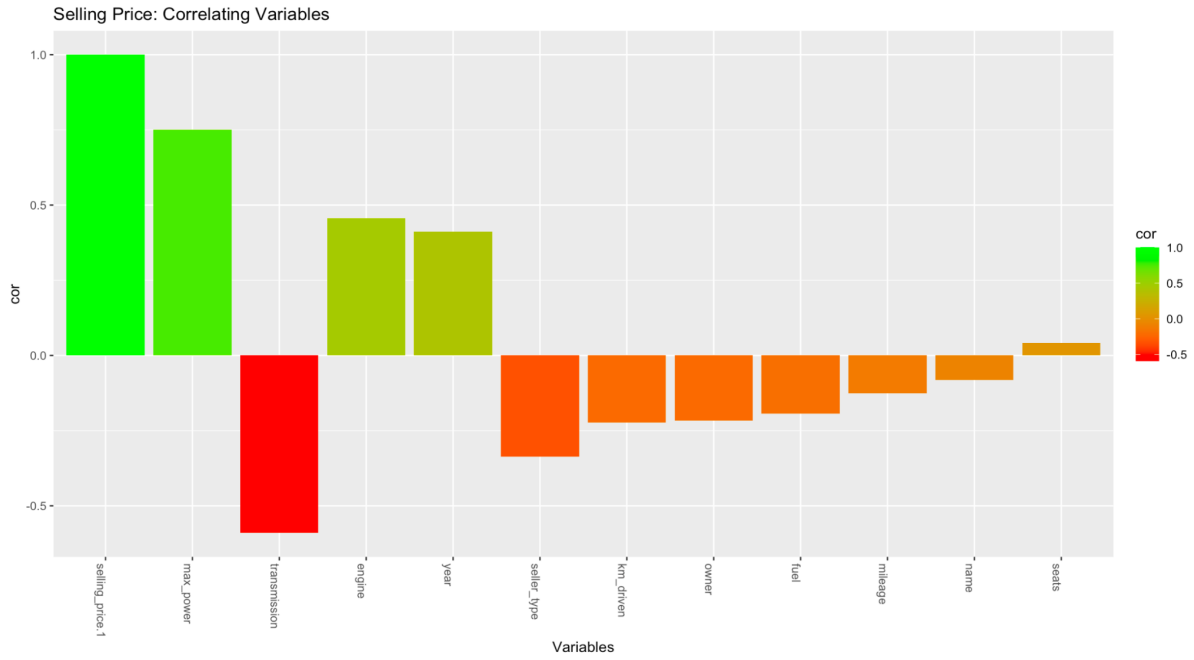


To visualize the correlations between all attributes, we generate a correlation plot. Selling Price has a strong positive correlation with max power, while it has moderate negative

correlation with transmission. Similarly, engine and max power have a strong positive correlation. Overall, there aren't many notably significant correlations between pairs.

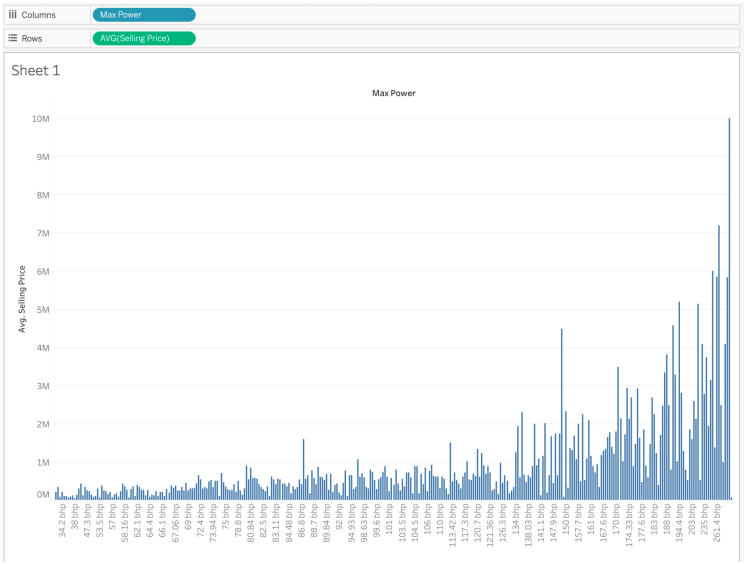


To check the correlation of other attributes with the target variable; selling price, plot the bar chart of the absolute correlations with the target variable. Highest correlation is a positive correlation with max power followed by a negative correlation with transmission. The variables with the smallest correlations are seats and name.



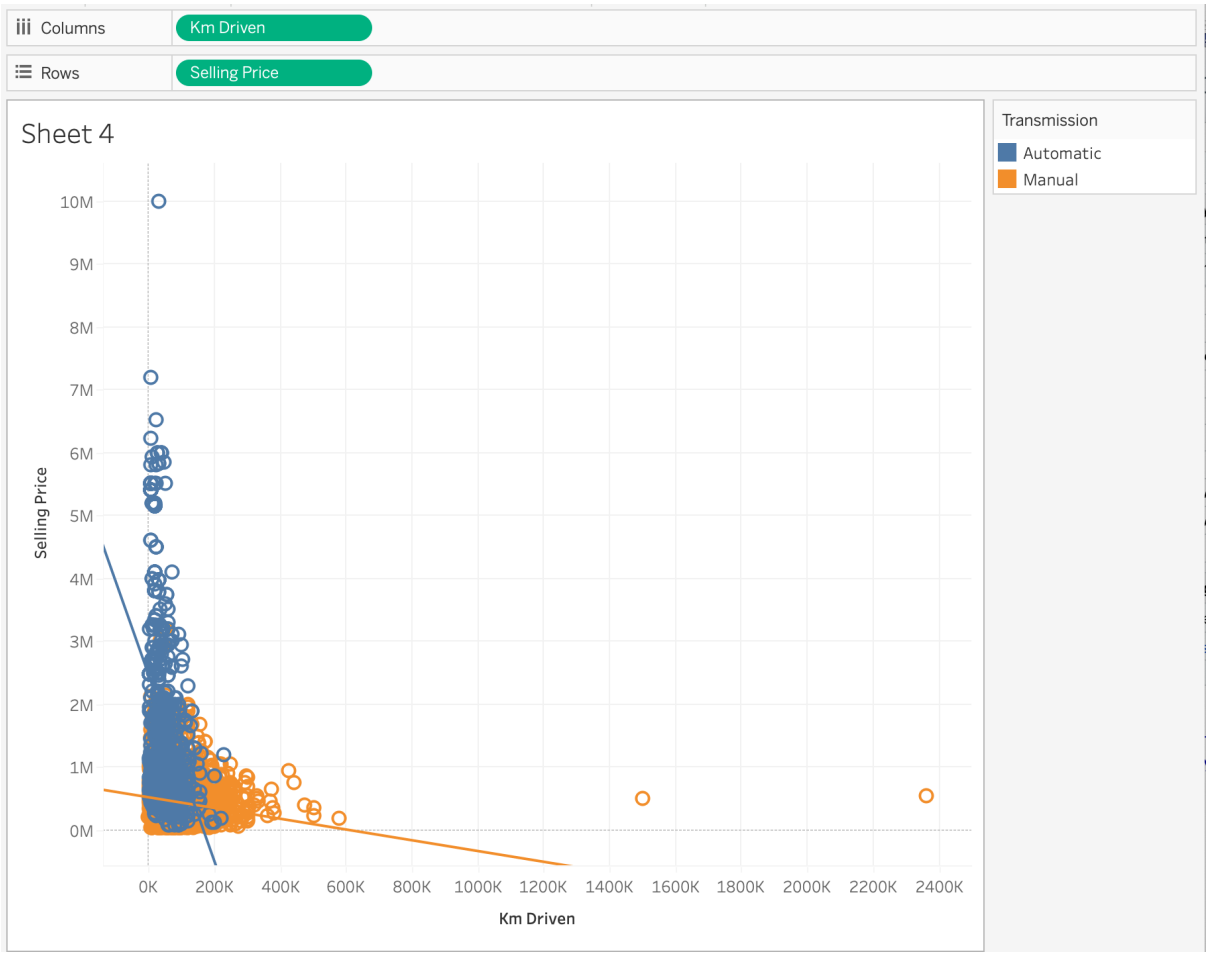
Tableau

Plot another bar chart to visualize the distribution of max power over selling price. Expectedly, there is a strong increasing trend between the variables, implying that max power is an important feature in predicting the price of a used car

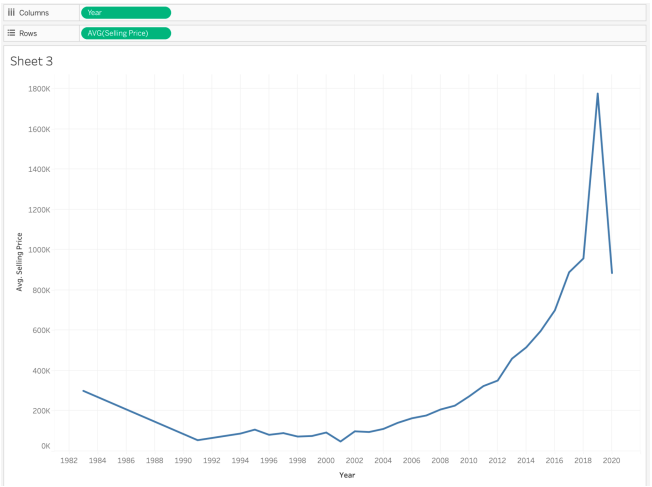


Plot a scatter plot to observe the correlation between km driven and selling price, a moderate negative correlation between variables is observed. An additional variable;

transmission is included. A much stronger negative correlation between transmission and selling price is observed from the distribution of colors.



Finally, generate a line plot to visualize how average selling price changes over the years. Although, overall an increasing trend is observed, from 2019 there is an abrupt downward trend which can be explained with the economic crisis resulting from the pandemic.



PART3

To apply the models, data must be preprocessed into a structured format by converting all variables to numeric. As there are different car types for each car brand, extract the first word of each car type and change them to factor to represent a categorical variable. 7 columns in the dataset have a 'character' class. First, convert the character columns; fuel, seller type, transmission and owner to categorical variables (factor) by assigning levels to the strings. The columns, mileage, engine and max power have string extensions, removing them and converting them to numerics ensures all variables to be stored in numerics. As the column torque is in a complex combination, remove it. Ensure that the observations are complete by defining a function to count the number of NA's. 4 variables have missing values, calculate their proportions. As all proportions are < 0.03, don't eliminate any columns, instead we impute the missing entries with their column mean. Finally, randomly split the dataset into training and test set using their row index. Then again randomly, split the training set into a smaller training and a validation set in order to evaluate the model performance.

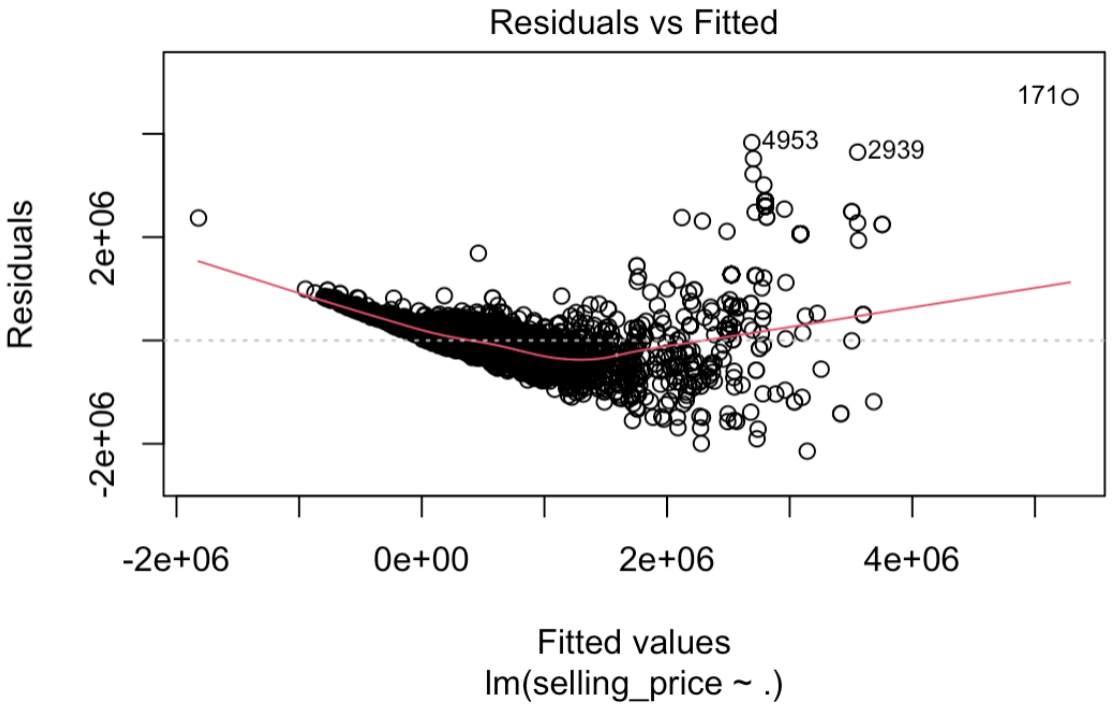
PART4

As no significant absolute correlation between independent variables was observed in the correlation table, all variables excluding the dependent variable are included as predictors.. However, by our visualizations we note that selling price is highly correlated to max power then transmission and name. These variables are expected to be good predictors in predicting the selling price of cars.

3 different base models that would likely fit well to regression analysis were built using the used car dataframe. Each generated model was trained in the smaller training set, and each model's performance was evaluated on the validation set based on the custom-generated function RMSE. As the target variable is continuous, the root mean squared error (RMSE) function is used to measure the spread of the predictions by comparing the realizations with the predictions from each model.

Initially a linear regression model including all predictors was fitted. The performance in the validation set was measured as 461558.4. As the error was significantly high, in order to reduce it by tuning the model, relative importance of attributes was printed. Another linear regression model was built, eliminating the least important 4 attributes. However, the error increased to 464183.3, damaging the model performance. Because of the presence of outliers in the data, a linear line couldn't fit well as seen in the below plot of the linear

regression model with all predictors. To reduce the error, alternative models were considered



Regression tree was built as our initial base model as it works well with continuous variables. Tuning the complexity parameter to 0.0000001 yielded the lowest error. Initial model included all predictors and the generated error was 199861. Another model was built, removing the least important variable; 'transmission' which damaged the model performance as the error increased to 200005.8. Hence, the final regression tree model includes all predictors. In hopes to further tune the model, log-transformation on the target variable was applied (when generating predictions it was converted back to normal scale). The error reduced to 182805.9. Hence the best regression tree model is with all predictors and log-transformed price data.

The next base models built are both from boosted tree families and as they build a forest with hundreds of trees when training it takes some time to run. Therefore, in order to speed up the run time with parallel processing we set up multiple cores as separate workers and make them a cluster, giving us the total number of available cores as 8.

Second base model; randomForest is an older boosted tree model, hence parallel processing is required to be set up manually to use all available cores. Number of trees in the model was kept as default(500) as it generated the lowest error, as regression is applied, maximum number of terminal nodes is set to 4 and the number of variables randomly sampled for each tree is set to 3. To ensure the same random sampling across models, the set.seed

function was fixed to 103.With the model with all predictors, the error generated was 135526.9 . To reduce the error and tune the model, the least important variable from the variable importance plot (fuel) was eliminated in the second model. As the error increased to 137308.7, fuel was kept. Then, log-transformation on price data was applied. As the error; 136123.7 was still larger than the initial model, the final random forest model included all predictors and no log-transformation.

As the final base model, XGboost was selected as its a powerful boosted tree that also works with regression. As XGboost only works with numerics, the predictor columns are converted to matrix and the response variable to vector in both training and validation sets. As XGboost is a more advanced tree, the parallel processing is set up directly by setting the nthreads to workers. Other parameters such as the maximum number of trees and maximum tree depth were decided based on trial and error. Setting these parameters higher results in the model becoming more complex, reducing error but also increases run time. Set the objective suitable for regression to train the model for a continuous response variable. Computing RMSE of the model with all predictors yields an error of 124438.1. To further tune the model, print variable importance and remove the variable with the lowest gain; seller_type, keeping all parameters the same. The error slightly increased to 124706.8 hence, keep the seller type. In another attempt to tune the model, we built another model applying log-transformation to the response variable. This model reduces the error to 123810.7. Hence, the best XGBoost model is with all predictors and with log-transformed price data.

Because of the nature and complexity of the dataset, the base models still don't generate satisfactorily low error rates. Therefore, in order to improve model performance, 3 ensemble models were fitted combining the base models. First, weighted average was applied, where each individual base model is assigned different weights. Define our best models for regression tree, random forest and xgBoost respectively as $M1$, $M2$ and $M3$ with corresponding weights $w1$, $w2$, $1 - w1 - w2$ (As the weights of all models add up to 1). Define 2 sequences for $w1$ and $w2$ starting from 0.1, ending at 0.9 jumping stepwise with a rate of 0.1. Create an RMSE matrix with 9 rows (possible values of $w1$), 9 columns (possible values of $w2$), and initialize each entry to 0. Within a nested for loop, define the condition that $w1 + w2 > 1$ is not a valid combination, if the condition is not true, the formula for weighted average is applied using the predictions from each of the base models and their corresponding assigned weights. The smallest entry in the matrix is the error of the weighted average ensemble model; 119072.7

To further reduce the RMSE, 2 stacking models are applied where the best regression tree and XGBoost models are used as the stacker model. In both cases, base models are trained in the smaller training set and predictions are generated in the validation set. Then, fit the

stacker model respectively to regression tree and XGBoost to our predictions generated from our base models, keeping the dependent variable same. While the regression tree stacker model decreased the error slightly to 114278.20, the XGBoost stacker model decreased the error significantly to 11526.64. We conclude that the XGBoost stacker model is the best fit model.

Description: df [6 × 3]

Model_No <dbl>	Model <chr>	RMSE_valid <dbl>
1	Regression Tree	182805.91
2	randomForest	135526.91
3	xgBoost	123810.74
4	Weighted Average	119072.75
5	Stacking - Regression Tree	114278.20
6	Stacking - xgBoost	11526.64

Finally, retrain the XGBoost stacker model using the entire training set and generate predictions in the testing set. Convert the predictors and response variables to numerics in the training set and testing sets. We observe that the error decreased to 9185.89 when the predictions from the best model was generated in a larger set of observation

PART5

I would recommend the decision-maker to particularly look at the year and the max power of a particular used car as these features are the most important variables generated by the best xgboost model which is the stacker model for our overall best fit model. However, as the stacker model includes the predictions from the base models as inputs, also keeping track of the most important variables for all best base models is also important (transmission, name, engine, km_driven).

References

Gokce, E., 2020. Predicting Used Car Prices with Machine Learning Techniques. [online] Medium. Available at: <https://towardsdatascience.com/predicting-used-car-prices-with-machine-learning-techniques-8a9d8313952> [Accessed 6 May 2022]

Joshi, K., 2022. Take a look at top 10 car websites in India. [online] ThePrint. Available at: <https://theprint.in/theprint-valuead-initiative/take-a-look-at-top-10-car-websites-in-india/913511/> [Accessed 10 May 2022].

Kaggle.com. n.d. Vehicle dataset. [online] Available at: <https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho?select=Car+details+v3.cs> [Accessed 13 May 2022].

Padalkar, P. and Mutreja, S., 2021. Used Cars Market Sales, Value, Trends, Forecast, Share - 2027. [online] Allied Market Research. Available at: <https://www.alliedmarketresearch.com/used-cars-market-A06429> [Accessed 9 May 2022].

Code Appendix