STAT 5102 Project Report

Price analysis and prediction in the HK used car market

Yam Tsz Fai (1155063821)
Chan Ka Hang (1097604892)

2022/05/09

1. Introduction

Despite the small size and convenient transport system of the city, HK has been one of the cities with the busiest traffic conditions in the world. According to the latest figures provided by the transport department of the HK government, the total registration of cars is 659,059 in Mar 2022. Comparing this with the total population of HK, you would have an idea on how active of the car selling market.

One interesting observation is that there were only around 2,300 gross first registrations on average every month in the past few years. This indirectly shows that most of the transactions are in the used car market. Although the used car market has been very active, we found that there is little objective information in the market to precisely evaluate the residual value of a used car.

Throughout the study, we aimed to:

1. Analyze and predict the general used car price with different traditional statistical approaches.
2. Use different machine learning techniques to predict the car price at a level of accuracy of an acceptable level.
3. For each ML approach, optimize the hyper parameters to make sure the model obtains the best prediction performance.
4. Compare the results of 1 and 3 and conclude the best approach.
5. Conclude the analysis and suggest the next steps.

2. Detailed Analysis

2.1 Overall flow

The flow of our study has been depicted in the diagram below.

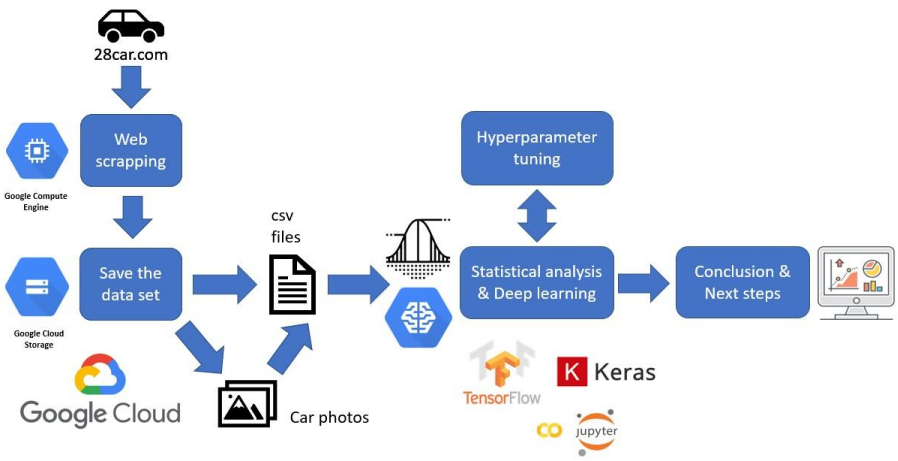


Figure 1: Overall workflow

2.2 Environment Setup

Before we started the analysis, this was important to have the data available. Since we aimed to focus on the local market and there was not a readily used dataset specifically for HK, therefore we had to use web scrapping techniques to collect the data from the most popular used car selling platform (28car.com).

Our web scrapping module utilized the Selenium package. The long duration made us impossible to do this on our home computer. Hence, a Compute Engine (GCE) was provisioned on Google Cloud Platform to do this task. Each job stream would visit the website with a unique header and randomly sleep during the process and check whether the record was already handled before downloading the information. As the website contained around 49000 transactions in the past year, it took around one week to collect the information.

The Python program downloaded the basic information of each deal and saved it all into a csv file. Also, since we are also interested in studying whether its color affects the transaction price, the photos were stored as well.

We saved all needed information into a Google Cloud Storage bucket for further analysis.

2.3 Data Collection

As mentioned above, instead of using the data directly, data extraction and manipulation were required.

Figure 2: 28car.com sample page

Before we performed the statistical analysis, we needed to handle the photos first and determine the vehicle color. Referring to the web materials, we read each photo and stored its pixel values in an array. We used KMeans to fit the color into 3 clusters to obtain the three average colors that appeared the most in the photo. Sample output was as follows.

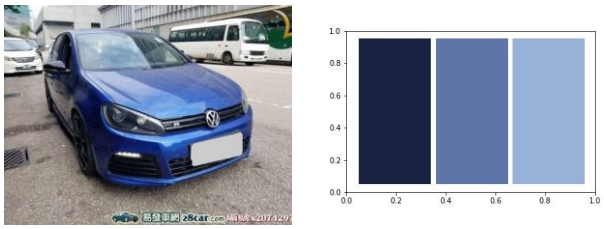


Figure 3: Illustration of color detection module

The reason why we picked the top three colors instead of one was that sometimes the photo was not taken in a way that the car was being put as the main object (for example, too small or under a dark condition). Below images were detected as green and grey respectively if we picked the top single color.

Figure 4: Bad quality photos

Apart from extracting information from photos, another complication in our analysis was that the most useful information was stored in Chinese. The seller inputted them into the comment field as a free text format.

Take the comment below as an example:

17年11月落地、0首、行貨、61000km、原廠保養記錄、牌費到11月、泊車感應、17寸PS4軑 內外極新淨

Above comment contains some useful information like:

1. Number of previous owner (0 首)
2. Travel mileage (61, 000 km)
3. Whether the registration fee is included (牌費)
4. Note that the remaining information was customized for different car models and manufacturers and hence we decided to discard them from our study.

As there were countless variances in the comment, we spent much effort to implement the regular expressions (RE package) to precisely capture the actual meaning of them and convert them from Chinese into numeric format, which was essential for the regression and machine learning analysis.

2.4 Analysis

Before we fitted the data to different models, this was important to have a basic understanding of the data. We should plot the scatter plots (Due to the limited space, please refer this in our program) to understand the pattern and distribution. Visualizing the data also helped us to identify the potential interaction among variables, outliers and influencing points. There were also some assumptions of regression that need to be fulfilled or else the output models will not be reliable.

2.4.1 Exploratory Data Analysis

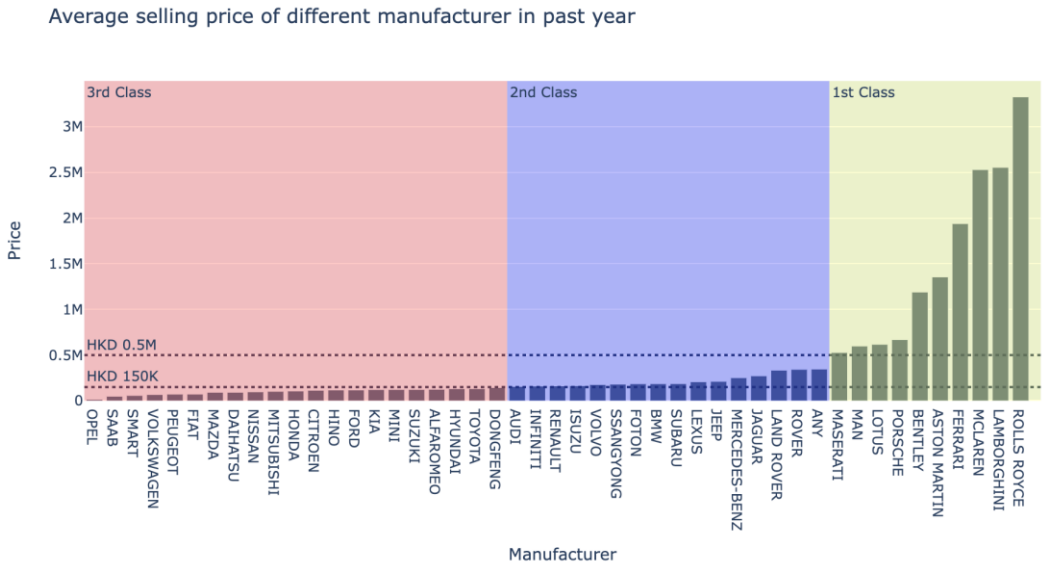


Figure 5: Relationship of manufacturer classes and their average price

As illustrated in figure 5, the distribution of selling price of different brand names can have a significant difference. Based on this, we added an additional variable (Mft_Class) to classify the manufacturer into 3 classes.

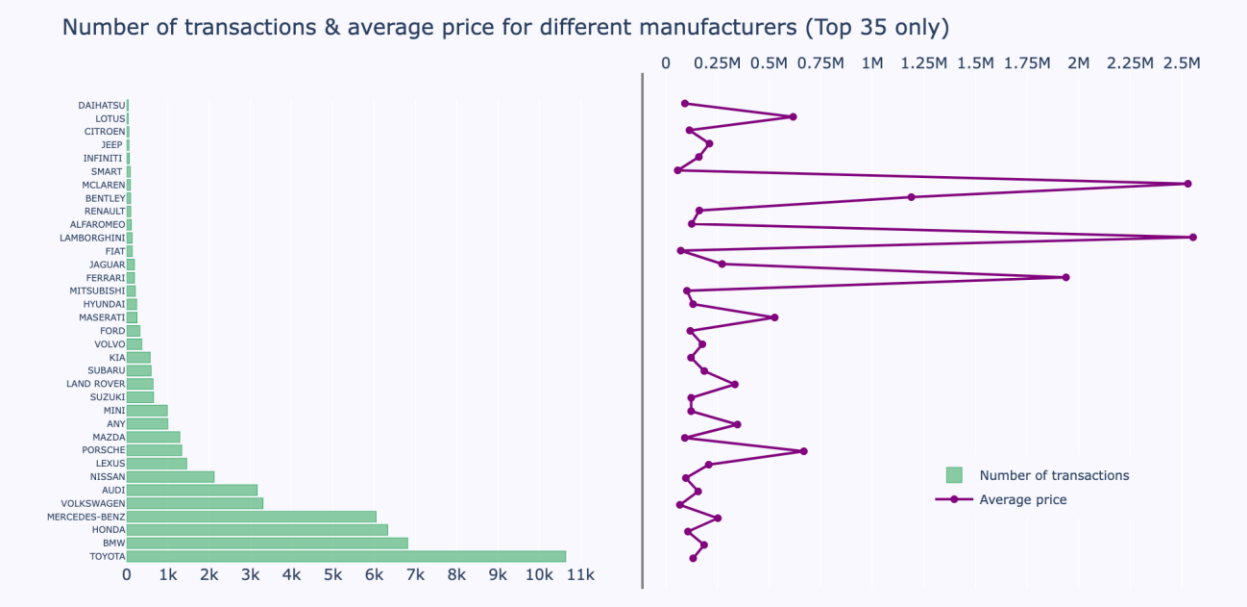


Figure 6: Combined plot of number of transaction and its average price

From figure 6, we knew that:

- In the past year, the 5 most popular brands in the market were Toyota, BMW, Honda, Mercedes-Benz and Volkswagen respectively, which belonged to the 2nd and 3rd class.

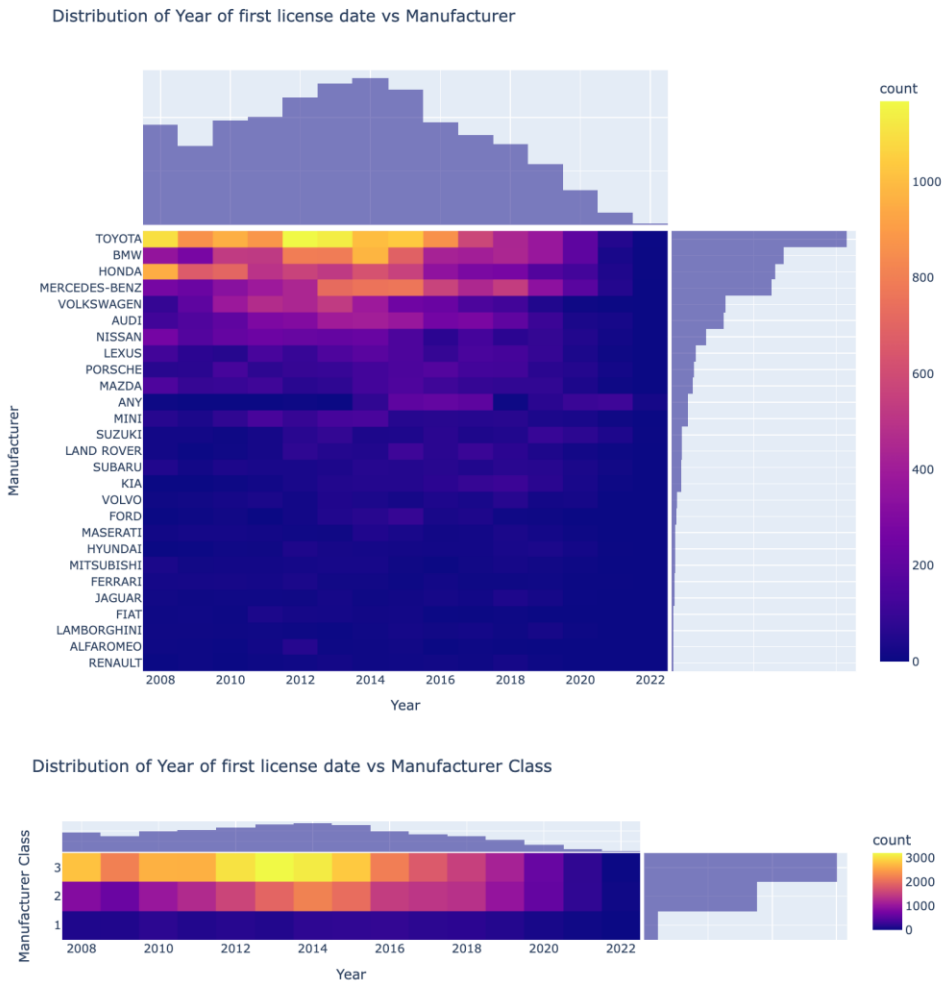


Figure 7: Heatmap of number of transactions against car brands / classes

Above heatmaps also show the histogram of the corresponding dimension. Basically, from them we knew that:

- The car in the third manufacturer class (i.e., with the lowest price range) tended to be more popular in the past year.
- In terms of the distribution of age of the selling car, the owners tended to sell their cars either in the 7th-8th year (around 2014) since first registered or when the car was very old (14th year since first registered). This aligned with the fact that the HK government imposes a compulsory requirement to have full functions examination after the 7th year since first registration.

2.4.2 General Linear Regression Model

2.4.2.1 Data Structure

We collected around 49,000 transactions but more than half of the transactions contained missing value. As we already had a large dataset, for easy handling, we decided to drop all transactions with missing value and only 18,639 transactions remained in dataset. Meanwhile, we created a new variable *Mft_Class* by classifying the manufacturers into 3 classes for further analysis on the relationship between car price and its manufacturer.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 18639 entries, 0 to 49935
Data columns (total 14 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   RecordNo      18639 non-null  object
 1   Manufacturer  18639 non-null  object
 2   Model         18639 non-null  object
 3   Seats         18639 non-null  int64
 4   Engine        18639 non-null  int64
 5   Transmission  18639 non-null  object
 6   Year          18639 non-null  int64
 7   Price         18639 non-null  int64
 8   Sell_date     18639 non-null  object
 9   AD_Import     18639 non-null  object
 10  Lic_fee       18639 non-null  object
 11  Preowner      18639 non-null  float64
 12  TravelDist    18639 non-null  float64
 13  Mft_Class     18639 non-null  int64
dtypes: float64(2), int64(5), object(7)
```

0: Index on the selling platform
1: The Brand of the used car
2: Model of the car
3. No. of seats
4. Engine Capacity (in cc)
5. Type of Transmission ('AT'/'MT')
6. Year of obtaining the first license
7. Selling Price
8. Date posted to selling platform
9. Imported used car
10. If License Fee included (Y/N)
11. No of Previous owner
12. Travelled Distance of the car
13. Classification of Manufacturers

Figure 8: Detailed legends of each predictor field

	RecordNo	Manufacturer	Model	Seats	Engine	Transmission	Year	Price	Sell_date	AD_Import	Lic_fee	Preowner	TravelDist	Mft_Class
0	s1928229	TOYOTA	PRIUS 1.8	5	1800	AT	2013	70000	03/26/2021	N	Y	0.0	170000.0	3
3	s1928303	FORD	KUGA	5	1500	AT	2015	79800	03/14/2021	N	Y	1.0	60000.0	3
6	s1928336	AUDI	A3 1.4T SEDAN	5	1400	AT	2015	120000	12/31/2021	N	Y	4.0	110000.0	3
10	s1928413	HONDA	JAZZ RS GK5	5	1496	AT	2019	143000	03/26/2021	Y	Y	0.0	9000.0	3
12	s1928442	MERCEDES-BENZ	C200 AMG ESTATE	5	1796	AT	2011	108000	03/29/2021	N	Y	0.0	80000.0	2

2.4.2.2 Data Cleansing

Apart from *Mft_Class*, we created or transformed some variables as below:

<u>A. New Variables</u>

- 14. *Age*: year difference on *Sell_Date* and *Year*, and drop *Sell_Date* and *Year*
- 15. *Colour*: We classified the detected colour by calculating the nearest distance between the detected RGB and 16 color labels (include black, white, silver, gray, blue, red, yellow etc.)

<u>B. Outlier/Extreme Values Detection</u>

- Removed 12 data points with its *Engine* more than 7500cc which was rare in used car market
- Remove 3 data points with extremely high *TravelDist* (i.e., more than 80 billion km!)

C. Transformation

- Transformed the *AD_Import*, *Lic_fee*, *Mft_Class* and *Color* as dummy variables
- Cut *Engine* and *TravelDist* into 10 bins instead of using its exact value as the range of both variables were very large and we believed the increase of 1 unit in Engine capacity or Travelled Distance won't affect much on the selling price. (*EV_bin*, *TravelDist_bin*)
- The variation in car price was so big that we used natural log transformation to reduce its variation. [*ln(Price)*]
- We spotted from the scatter plot that the relationship of *ln(Price)* with *TravelDist_bin* and *Seats* was in second order and transformed to *Seats_T* [*Seats* – mean(*Seats*)], *Seats_Squared* [2nd order of *Seats_T*], *TravelDist_bin_T*[*TravelDist_bin* – mean(*TravelDist_bin*)] and *TravelDist_bin_Squared* [2nd order of *TravelDist_bin_T*].

2.4.2.3 Results

The statsmodel.OLS() in Python was used to fit the model. The response *ln(Price)* was fitted to a common intercept, *Preowner, Age, Lic_fee, Transmission, Colour, EV_bin*, dummy variable of *Mft_Class*(*Mft_Class_3* was removed from model to avoid collinearity), *Seat_T, TravelDist_bin_T, Seats_Squared, TravelDist_bin_Squared, and the interaction variable*

Table 1: Fitted result of final regression model

	coefficient	std err	t-value	P-value
constant	12.8061	0.025	510.562	<0.001
Preowner	-0.0513	0.003	-16.376	<0.001
Age	-0.1798	0.001	-153.812	<0.001
AD_Import	0.0207	0.007	3.060	0.002
Lic_fee	-0.1270	0.007	-18.267	<0.001
Transmission_AT	-0.5511	0.019	-29.132	<0.001
aqua	0.4305	0.131	3.283	0.001
black	0.0441	0.013	3.401	<0.001
blue	-0.5004	0.307	-1.631	0.103
navy	0.0376	0.019	1.970	0.049
purple	-0.0930	0.050	-1.856	0.063
red	0.2286	0.043	5.287	<0.001
yellow	0.1278	0.071	1.798	0.072
EV_bin	0.1945	0.002	89.990	<0.001
Mft_Class_1	1.2001	0.020	59.999	<0.001
Mft_Class_2	0.4603	0.015	29.942	<0.001
Seats_T	-0.0230	0.008	-2.895	0.004
TravelDist_bin_T	-0.1377	0.024	-5.655	<0.001
Seats_Squared	0.0406	0.002	21.281	<0.001
TravelDist_Squared	0.0177	0.005	3.769	<0.001

Seats_Class1	0.0293	0.013	2.203	0.028
Seats_Class3	-0.0530	0.009	-5.814	<0.001
Seats_Lic	0.0265	0.006	4.387	<0.001
EV_Class1	0.0522	0.004	12.512	<0.001
EV_Class2	0.0400	0.004	10.020	<0.001
EV_Class3	0.1023	0.004	25.681	<0.001

	R-squared	**Adjusted R-squared**	**F statistics**	**P-value**
Summary	**0.801**	**0.801**	**3117**	**<0.001**

From the above result, overall model is significant at 15% significance level. The Adjusted R –squared indicated that around 80% of variation could be explained by the model.

2.3.2.4 Model Evaluation

A. Is Color a significant predictor of car price?

Initially, we believed that color was an important factor in the car price prediction, hence we put lots of effort to identify the color of each car. However, most of the color labels were dropped by backward elimination and we suspected that if this factor significant for car price prediction.

We carried out an F-test to compare the model with and without color labels and the p-value was less than 0.05. Hence, we did not have sufficient evidence to conclude that the model without color labels as predictors was a better model and colors were significant predictors of car price.

```
   df_resid          ssr  df_diff    ss_diff         F        Pr(>F)
0   18558.0  3498.866809      0.0        NaN       NaN           NaN
1   18551.0  3488.476539      7.0  10.390271  7.893331  1.374207e-09
```

B. Multicollinearity

Table 2: VIF of the predictor variables

	VIF	Tolerance
Preowner	1.382562	0.723295
Age	1.507199	0.663482
AD_Import	1.097545	0.911124
Lic_fee	1.181289	0.846533
Transmission_AT	1.056737	0.946309
aqua	1.005225	0.994802
black	1.609388	0.621354
blue	1.001398	0.998604
navy	1.582243	0.632014
purple	1.010284	0.989820
red	1.038300	0.963113
yellow	1.019129	0.981230
EV_bin	inf	0.000000
Mft_Class_1	2.992536	0.334165
Mft_Class_2	5.406432	0.184965
Seats_T	7.984461	0.125243
TravelDist_bin_T	2.830489	0.353296
Seats_Squared	2.035317	0.491324
TravelDist_Squared	2.805028	0.356503
Seats_Class1	3.978781	0.251333
Seats_Class3	6.656954	0.150219
Seats_Lic	1.904969	0.524943
EV_Class1	inf	0.000000
EV_Class2	inf	0.000000
EV_Class3	inf	0.000000

The VIF indicated that multicollinearity did not exist in the model.

C. Residuals Analysis

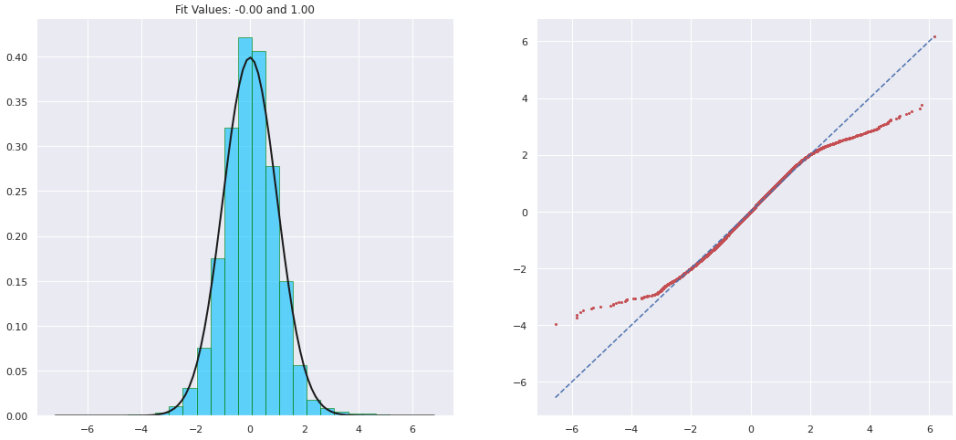


Figure 9: QQ plot of residuals

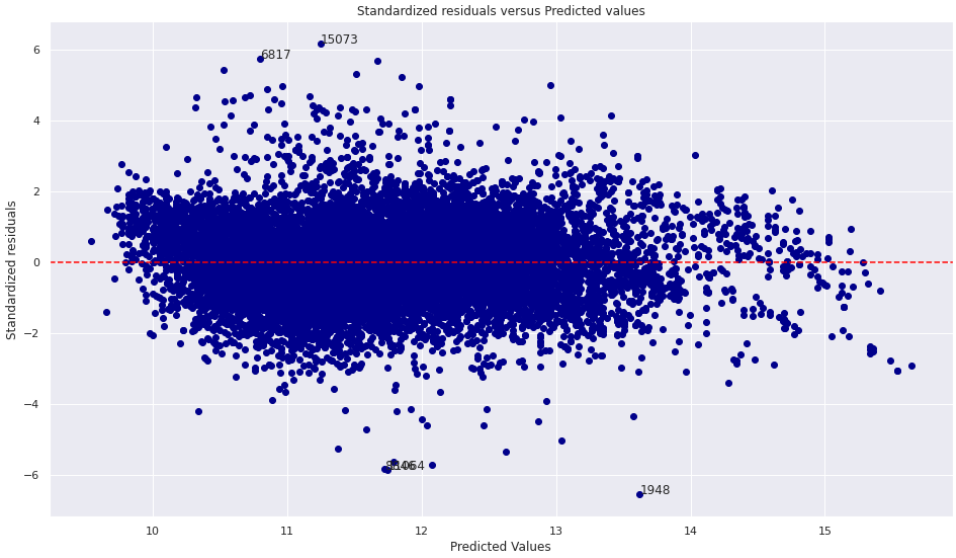


Figure 10: studentized deleted residual plot

The residuals-QQplot indicated that the residuals were in thin-tailed normal distribution with variance from –6 to 6. This evidence showed that the residuals were not in normal distribution. In addition, the p-value of BP test was less than 5% that heteroscedasticity was existed.

We tried to remove outliers which absolute value of their variance were larger than 3.5 and influence points with high leverage value and more than 100 data points were removed. *Aqua* was removed from the model after removing the outliers and the Adjusted R-squared was increased to 80%.

Table 3: short summary of the model after removing outliers and influential points

	R-squared	Adjusted R-squared	F statistics	P-value
Summary	**0.819**	**0.819**	**3629**	**<0.001**

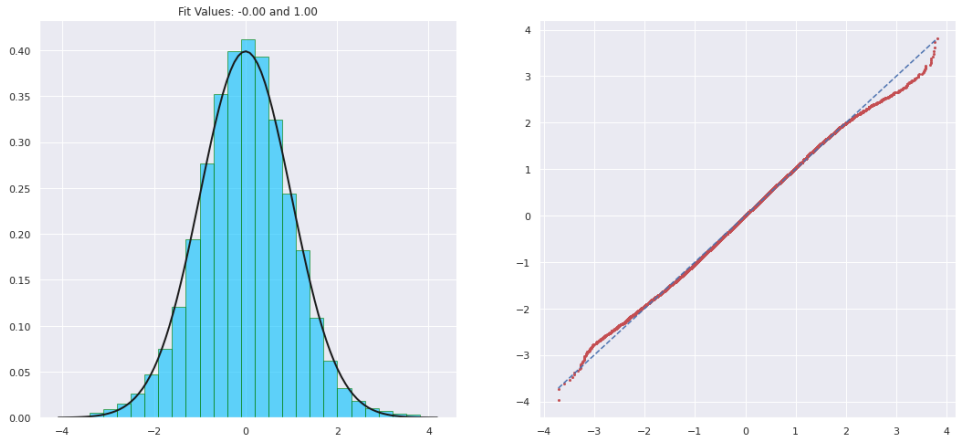


Figure 11: residuals-QQ plot of model after removing outliers and influential points

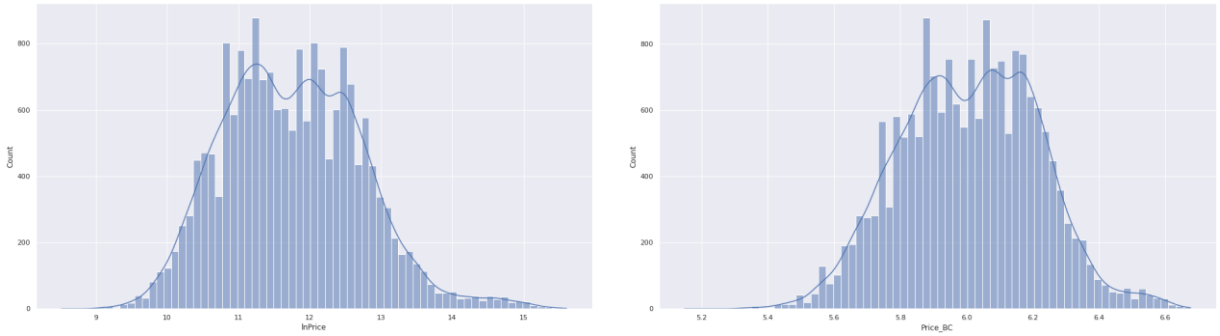
However, the p-value of BP test and Kolmogorov–Smirnov test were lower than 0.05. The assumption of homoscedasticity and normal distributed residuals were violated, hence we looked for new approaches for this dataset.

D. Different approaches to improve the model

(1) Box Cox Transformation

Box Cox Transformation was used to transform the dependent variable when the residuals were not normally distributed.

Figure 4 and 5: Distribution of log transformation and Box Cox transformation



Histogram of Price after log transformation

Histogram of Price after Box Cox transformation

Figure 12: Illustration of price transformation

For this case, log transformation was a better option to transform our response. The distribution after Box Cox transformation was left skewed and log transformation was more likely be normal distributed.

(2) Separated models for 3 Manufacturer Classes

Although log transformation was a better option, its distribution was not in bell shape. We suspected that the price gap between different *Mft_Classes* caused *ln(Price)* was not in normal distribution and we tried to predict the used car price by three separate models instead of a pooled model.

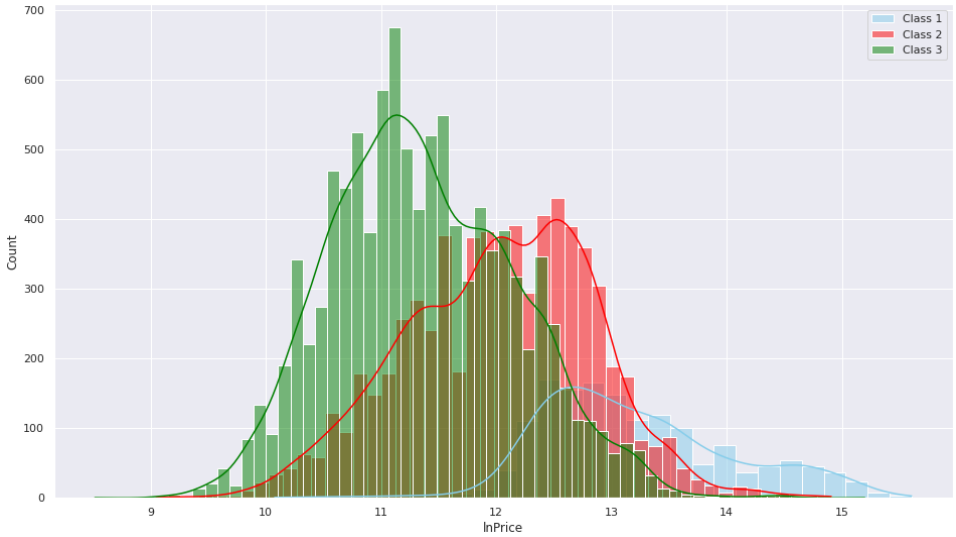
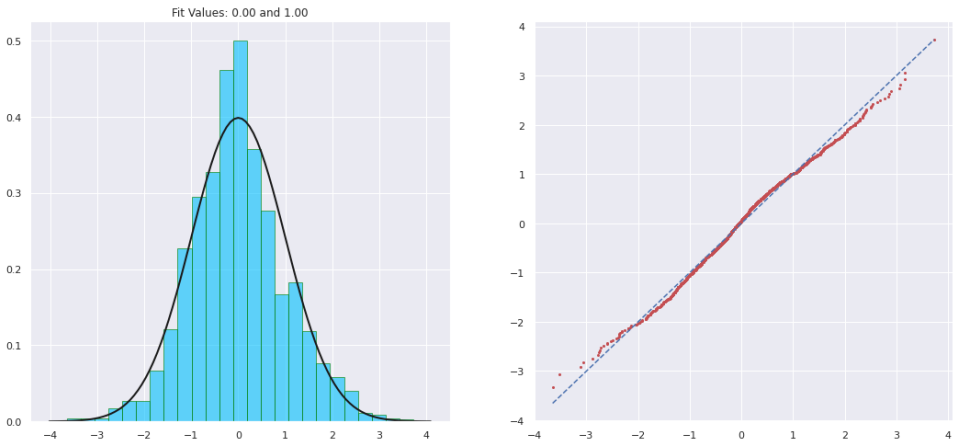


Figure 13: The distribution plot for different Manufacturer Classes

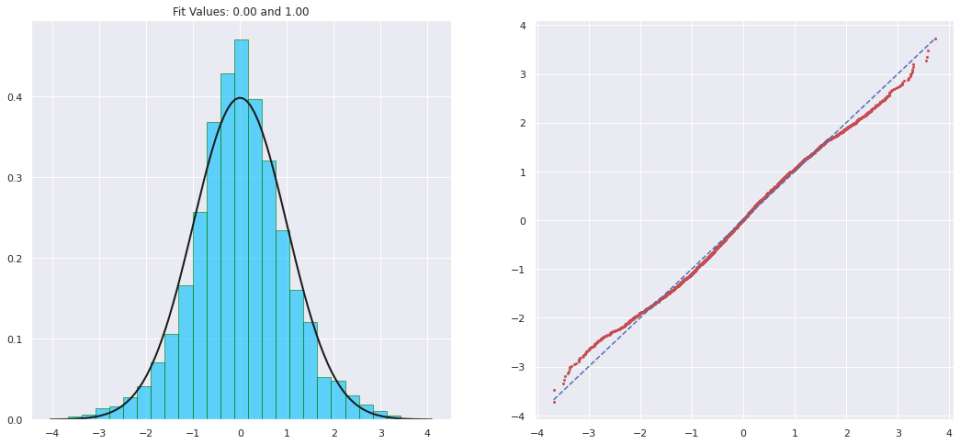
Table 4: Summary for models of each Manufacturer Class after dropping outliers and influential points

	R-squared	Adjusted R-squared	F statistics	P-value
Class 1	**0.713**	**0.710**	**281.7**	**<0.001**
Class 2	**0.782**	**0.782**	**2195**	**<0.001**
Class 3	**0.733**	**0.733**	**2550**	**<0.001**

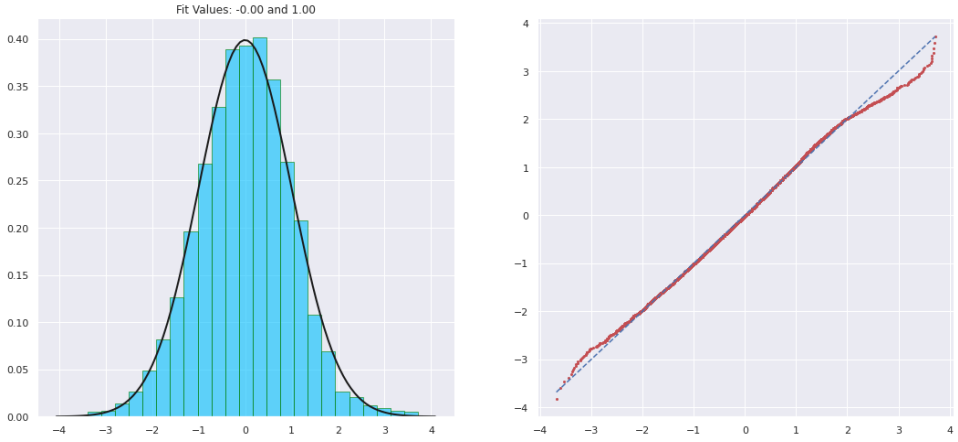
Figure 14-16: residuals-QQ plot for 3 models



Model 1



Model 2



Model 3

The results of separating 3 models were not significant as we expected, the Adjusted R-squared of the pooled model was higher than all 3 separated models and only model 3 passed the Kolmogorov–Smirnov test and heteroscedasticity existed in all three models. The separated models still violated the assumption of linear regression.

(3) Weighted Regression

Weighted regression was used to control residuals variance. We estimated the weight by the coefficient of fitting the residuals and predicted responses or predictors. We discovered that residuals were independent with predictors except the squared of *TravelDist_bin* and its results were like the pooled model.

2.4.2.5 Summary

The pooled model was a significant model, its results were the best in terms of adjusted R-squared among the models that we built. However, it violated the assumption of normal distributed residuals and homoscedasticity. We tried several ways to improve our models including Box Cox transformation, separating model into three model by its manufacturer class and weighted regression but we couldn't deal with these issues. Hence, we explored in the following for used car price prediction.

2.4.3 Machine Learning & Deep Learning approach

Regression is a type of supervised machine learning algorithm used to predict a continuous label. The goal is to produce a model that represents the 'best fit' to some observed data, according to an evaluation criterion.

Before we fitted the data into different models, some preprocessing were done like removing outliers, encoding categorical variables with ordinal encoder, standardizing the variables with standard scaler and taking log transformation on the price. For details, please refer to our program.

We used the regressors below in our study:

- Linear Regressor
- Extreme Gradient Boosting Regressor
- Gradient-Boosting Regressor
- RandomForest Regressor

We also wanted to take this chance to understand the deep learning usage and performance. As such, two neural networks were built with Tensorflow Keras library.

- Deep neural networks (small)
- Deep neural networks (large)

The major difference between small and large DNN was the number of output vectors of each hidden layer. Small DNN had 16, 8, 4 and 1 output units at each layer whereas Large DNN had 64, 32, 16 and 1 for the same.

2.4.3.1 Evaluation Metrics

We evaluated the performance of the deep learning model using Mean Squared Error (MSE), a commonly used metric for regression problems. In simple terms, MSE measures the average magnitude of the residuals or error. Mathematically, it is computed as an average of squared differences between predicted and actual values.

2.4.3.2 Hyperparameter optimization

For both ML and DL, we used all algorithms with their default settings as the initial baseline (Table 5). Then, in order to have the best-to-best comparison, hyperparameter tuning was included in our project study. Practically, for each algorithm, we used cross-validation technique (i.e., GridSearchCV) to find out the optimal parameter sets (Table 6) that provided the model best prediction performance (Table 7).

	Model	MSE	MAE	MAPE	MSLE
0	LR	8208752156.98	53392.79	0.34	0.18
1	GB	3515771843.73	36956.69	0.25	0.1
2	XGB	1895393788.68	25957.77	0.18	0.06
3	RF	2016701825.71	25266.86	0.18	0.07
4	DL_SMALL	24613109410.35	111771.11	0.95	0.78
5	DL_LARGE	4592791664.06	44753.94	0.34	0.14

Table 5: Model prediction behavior baseline

Below are the results that we obtained from the grid search.

Algorithm	Candidates	Best estimator	Score
Linear Regression	'fit_intercept':[True,False], 'positive':[True,False], 'copy_X':[True, False]	LinearRegression(copy_X=True, fit_intercept=True, positive=False)	76.04%
Gradient Boosting Regressor	'learning_rate': [0.01,0.02,0.05,0.1], 'subsample' : [0.95, 0.9, 0.5, 0.2], 'n_estimators' : [50,100,500], 'max_depth' : [3,4,6,8]	GradientBoostingRegressor(max_depth=6, n_estimators=500, subsample=0.95)	91.20%
Extreme Gradient Boosting Regressor	'learning_rate': [0.01,0.02,0.03,0.04], 'subsample' : [0.9, 0.5, 0.2, 0.1], 'n_estimators' : [100,500,1000, 1500], 'max_depth' : [4,6,8,10]	XGBRegressor(learning_rate=0.03, max_depth=8, n_estimators=1500, subsample=0.9)	91.58%
RandomForest Regressor	'n_estimators' : [10, 30, 100, 300, 500], 'max_features' : [2, 4, 6, 8, 10, 12, 14]	RandomForestRegressor(max_features=10, n_estimators=500)	89.12%
Deep Neural Network (Small)	'batch_size': [10, 20, 40, 60, 80, 100], 'epochs' :[10, 20, 30, 50]	batch_size=80 epochs=20	N/A
Deep Neural Network (Large)	'batch_size': [10, 20, 40, 60, 80, 100], 'epochs' :[10, 20, 30, 50]	batch_size=80 epochs=20	N/A

Table 6: Grid search results

With these selected parameters, we did the model fitting again and not surprisingly, all the results were better than the default setup, which is illustrated below.

	Model	MSE	MAE	MAPE	MSLE
0	LR	8208752156.98	53392.79	0.34	0.18
1	GB	1717280689.52	24023.74	0.17	0.05
2	XGB	1516432318.45	22592.62	0.16	0.05
3	RF	2053553160.24	25596.19	0.18	0.07
4	DL_SMALL	11167211959.04	59459.56	0.37	0.18
5	DL_LARGE	5846459984.43	47126.71	0.3	0.13

Table 7: Model prediction behavior after hyperparameter tuning

** Note: the program may generate slightly different result for each run*

3. Conclusion(s)/Discussion

3.1 Prediction accuracy

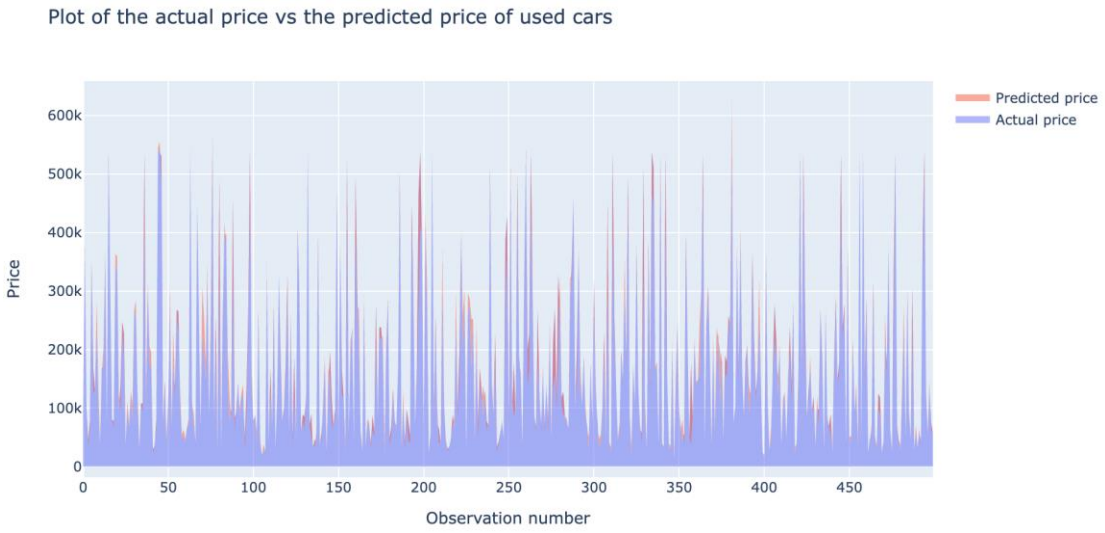
At the beginning of the study, we tried to use traditional statistical approaches to do the analysis. However, after many trials we were unable to make the residual as normal distributed, the variance of residuals was larger than expected and heteroscedasticity existed. Hence the predicted results should not be considered reliable.

Thus, we considered redoing the prediction analysis with machine learning approaches, which basically is an extension of regression analysis. A few algorithms in Keras were chosen in our study.

The Random Forest Regressor (RFR) came second in the list of algorithms with prediction performance, behind the Extreme Gradient Boosting Regressor (XGBR). DNN overall did not perform good based on our data set.

This was also observed that the small DNN was not producing promising results when compared with large DNN and other ML algorithms.

If we choose 500 observations, plot the prediction results that produced from XGBR with their actual selling price, below area chart is obtained. As we can see, the two areas overlap most of the time.



3.2 Other findings / potential improvement

3.2.1 Statistical Regression

Adding useful predictors

In the statistical regression analysis, one possible reason for the wide range of residuals variance was some important predictors were missing from the regression model and made the residuals unexplained. We should consider including more useful predictors and the status of the cars that were sold or being sold would be one of the possible predictors to be added.

Narrowing the scope for prediction

We found that the price range in our dataset was very large, and it is one of the possible reasons for violating the regression assumption. We suggest filtering before data collection, for example focusing on four or five common car brands or similar classes for regression.

3.2.2 Machine Learning & Deep Learning

Collinearity in Machine Learning

In our course we learned that collinearity should be carefully handled as this would affect the reliability of the coefficients of the model and hence the prediction result. During the project, we learned that this is not mandatory to handle in ML world as it is purely a curve fitting technique and only produces the best possible prediction scheme to drive a particular objective like accuracy. This is a very new concept to us, and we will further explore this topic after the project.

Neural Network tuning

According to the article, the behavior of DNN could be improved by changing the activation function, number of neurons, changing the training parameters etc. We may explore this area in detail to seek further improvement.

Grid Search vs Randomized Search

While the parameter tuning is essential, the grid search cross validation techniques has been taken very long time as it searched all parameter combinations. In future, we should consider using another technique like randomized search for cross validation, which will generate the combination randomly instead of trying all combinations extensively. We believe this is especially useful when we deal with some algorithms that have more than 5 tunable parameters (e.g. Random Forest).

3.2.3 Some other ideas / retrospectives

Improving the data quality

Throughout the study, we realized that the prediction quality and meaningfulness of a data science project would depend on the quality of data. But in the live case, this is rare to have a well-formatted data set available for use. Sometimes the data may only be indirectly available from other sources like in our case and we could not eliminate input errors from our dataset. All these factors may affect our prediction accuracy.

Usage of cloud computing for training models

We found that this was extremely slow when we fitted the data to the machine learning models with our home pc. The situation was even worse when we did the hyperparameter optimization with grid search to find optimal parms. In future, we should consider using cloud computing which effectively has unlimited computing power. Of course, this comes with a cost.

About the vehicle color detection

In our project, we used KMeans classifier for vehicle color detection. Although this was proven to be insignificant to the price in the statistical analysis, this may be due to few factors like non-standard quality of the photos, the heavy skew on the data (The traditional colors like grey, black and white are too popular in the market), which may all affect the outcome. In future, if we continue the analysis, we may consider collecting data in equal numbers of distinct colors with good photo quality. Also, we should consider using some licensed libraries for main object color detection to generate a more precise output.

4. Reference

Transport Department figures

https://www.td.gov.hk/filemanager/en/content_4883/table41a.pdf
https://www.td.gov.hk/filemanager/en/content_4884/table41c.pdf

Used Car Price Prediction using Machine Learning

https://towardsdatascience.com/used-car-price-prediction-using-machine-learning-e3be02d977b2

How to improve the performance of Neural Networks

https://d4datascience.in/2016/09/29/fbf/

How to Grid Search Hyperparameters for Deep Learning Models in Python With Keras

https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/

Why is multicollinearity not checked in modern statistics/machine learning

https://stats.stackexchange.com/questions/168622/why-is-multicollinearity-not-checked-in-modern-statistics-machine-learning

Weighted Least Squares

https://online.stat.psu.edu/stat501/lesson/13/13.1