CS5228 Project Report — BoosterShots

Anand Sundaram
A0118897J
e0403448@u.nus.edu

Chaitanya Kumar
A0212228R
e0503507@u.nus.edu

Hoki Fung
A0225572E
e0576206@u.nus.edu

Lok You Tan
A0139248X
e0003133@u.nus.edu

***Abstract*—This is the CS5228 project report for team BoosterShots. In this project, we applied data mining skills on a resale transaction dataset from sgCarMart to gain insights on Singapore's used cars market. For Task 1, we managed a score of 27424 and position 35 on the leaderboard (as of Nov 14 16:00), using an XGBoost ensemble approach with sentiment information from car listing descriptions. For Task 2, we designed and implemented a recommendation system that took into account the price point and car characteristics of the selected listing. For Task 3, we decided to explore the dataset further in 3 directions: finding the best dealers for particular types of vehicles, analysing the market for eco-friendly cars, and finding which car brands hold their values best, i.e., depreciate the least.**

I. Motivation

In this project, we applied data mining skills learned in, and beyond the classroom, on a resale transaction dataset from sgCarMart to gain insights on Singapore's used cars market.

For Task 1, we predicted car resale prices using some common properties in a car listing, such as make, model, and mileage, as features in our models. This task was highly sensitive to data-cleaning and analytical techniques. Therefore, for this task, our team sought to compare different ways of cleaning the data, as well as different methods for regression. In this report, we describe our methods in detail and compare the performances of various cleaning and modeling methods used for this task.

For Task 2, we designed and implemented a recommendation system to show users content that might be of interest to them. We compared different methods for recommendation. In addition to the obvious – simply recommending the most similar cars to what they were browsing – we attempted to see from a potential buyer's perspective and think about what makes a good recommendation. In our system, we included cars at a similar price point and characteristics (e.g., car body, power, color), but most importantly, we also included a small amount of listings that deviated from the above for the user to explore.

For Task 3, we explored the dataset further and extracted non-trivial and useful information from the data to provide insights on topics that different stakeholders, such as car buyers, car sellers, and governmental organizations, might be interested to know. We decided to pursue three directions. First, we wanted to see if any insights could be drawn from the dealers active on sgCarMart, with the intent of identifying popular and reliable dealers for buyers to pick from. Second, we explore the statistics on the car market with regards to the popularity and prices of eco-friendly cars, to identify the trends and consumer habits in the Singapore car market. This information could be of use to governmental organizations, car manufacturers, environmental agencies, for policy-making, reporting and market analysis. Third, we examined price depreciation by make, with the intent of identifying makes that have good long-term value for money in the Singapore market.

II. Exploratory Data Analysis & Preprocessing

We preprocessed the dataset based on the assumption that the data records were legitimate (i.e., no false information and no outliers to be removed), and so we kept as much data as possible during the cleaning. We believed it was a reasonable assumption since these were listings for used car sales, and it would not make sense for the sellers to lie about the specifications of the cars as they could be easily verified. Other pertinent details of the cars (e.g., registration dates, Certificate of Entitlement (COE) expiry) could also be verified through official government sites so there was no point for someone to lie about it.

However, despite our assumption that all data records were legitimate, there were still many missing values for many properties. This could be due to many reasons, for example, sellers neglecting some optional fields when posting their listing, sellers assuming some car details (e.g., standard features of a make and model) could be obtained somewhere else (e.g., manufacturer's website), and sellers intentionally leaving out some potentially negative information such as high mileage to attract more viewers.

A. Exploratory Data Analysis

First, we looked into the dataframe information for an overview of the variables, number of observations, and data types in the dataset. We then took a closer look at the missing data. We found that except for 8 columns including *listing_id*, *title*, *model*, and *price*, out of the 30 variables, the majority of the variables had some missing data, ranging from 1% to 100%. Since our goal was to keep as much data as possible, with a large amount of missing data, we would need ways to fill in these missing data.

Second, we examined the correlations between the numerical features in the dataset as we knew that some features were likely to be highly correlated, for example, *power* and *engine_cap*, as bigger engines usually produce more power. We are unable to exhaustively list all correlated pairs here, but Figure 1 shows the correlation matrix of the numerical features, with red being positively correlated, white being not correlated, and blue being negatively correlated. As can be seen from the figure, there are many highly positively correlated numerical features, including but not limited to:

- *road_tax* and *engine_cap* with a correlation of 0.97
- *arf* and *omv* with a correlation of 0.94
- *engine_cap* and *power* with a correlation of 0.88
- *omv* and *power* with a correlation of 0.85
- *arf* and *dereg_value* with a correlation of 0.85

We kept these correlated numerical features in mind as possible targets for dimension reduction, but had not yet attempted to remove them as the number of features were not extremely high.

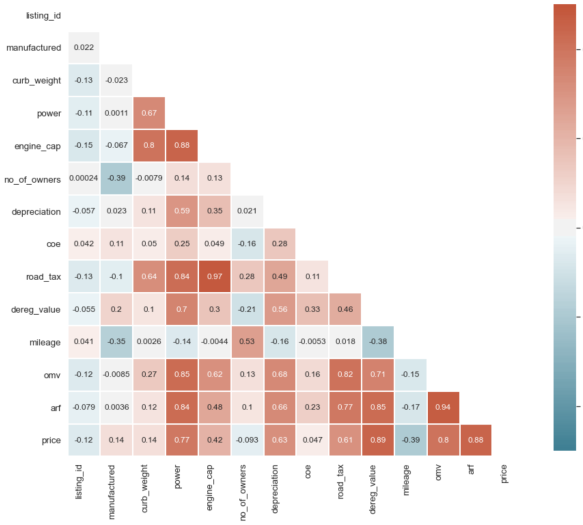


Fig. 1. Correlation Matrix of Numerical Features

Third, we plotted the prices in our training data to get a better idea of the data distribution. Figure 2 shows the histogram of used car prices in our training data. As expected, the data follows some kind of Gaussian distribution — most people buy average cars, some people buy cheap cars, and very few people can afford expensive cars. The same goes for the resale market.

In addition, we plotted the price vs. deregistration value of the cars in our training data. We observed from Figure 3 that the deregistration value serves as a good indicator of the lower bound of the price. Since the deregistration value is the amount that the government will rebate upon deregistering the car, the price of the car should most likely be set higher than the deregistration value if the seller is rational, to account for the value of the car body itself. Otherwise, the seller would profit more from deregistering the car and selling the car body to a dealer.

B. Preprocessing

Based on the insights we gained from the exploratory data analysis, we performed data cleaning and feature construction to prepare the data for our regression models.

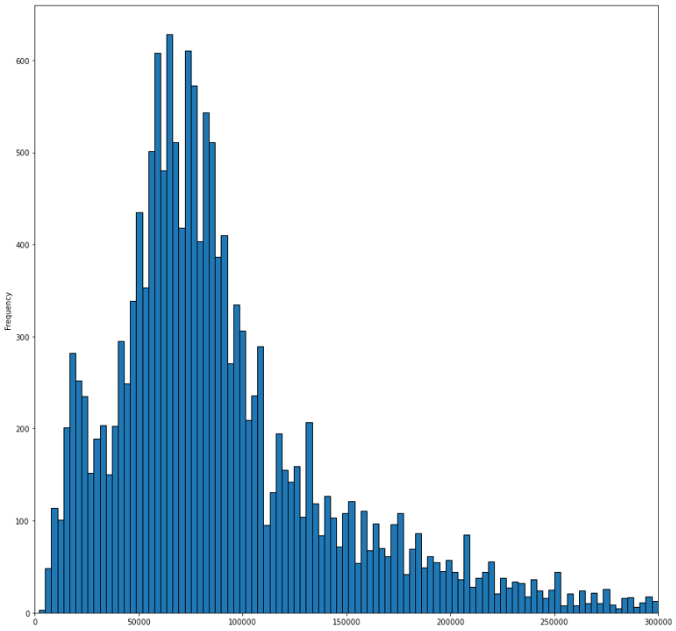


Fig. 2. Histogram of used car prices in training data

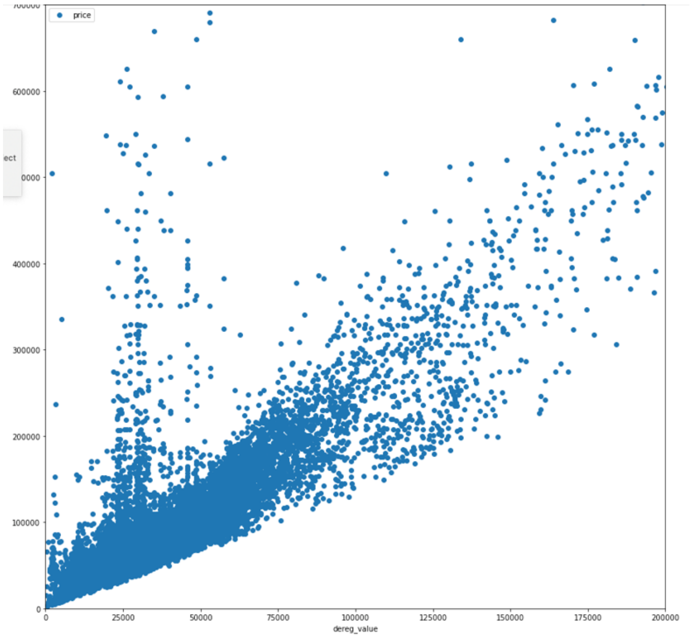


Fig. 3. Price vs. Deregistration Value

1) Data Cleaning: There were a total of 30 columns in our training data (including the price), and 29 columns in our test data (there was no price column since that was what we are trying to predict). As we omitted the price column and cleaned both the train and test data the same way, in this section, the term 'data' refers to both the training and test data. It is important to note that the cleaning methods described below served as the default in our subsequent analyses, but in some models, we further refined our cleaning procedures to fit the context. Please refer to our scripts for the exact cleaning procedures used in each model.

The first column, *listing_id*, did not provide any useful materials for our task of regression, as it was simply a running counter from 0 to the number of listings in the data. Thus, we dropped the *listing_id* column.

Of the remaining 28 columns, 22 columns had null values. The only columns in the training data and test data that had no missing values for all 16,784 and 5,000 records respectively are: *title*, *model*, *type_of_vehicle*, *category*, *transmission*, and *eco_category*.

For these 22 columns with null values, we tried our best to fill the missing cells with reasonable values. We started by filling in the missing values for *original_reg_date* and *reg_date*. We observed that the two values are almost always identical. This is likely because there is only a small minority of owners that will renew their Certificate of Entitlement. If either column is missing, we filled it in with the value from the other column. This worked properly for all records, i.e., there were no records that have both *original_reg_date* and *reg_date* missing. After that, we converted the *original_reg_date* and *reg_date* from a string to the datetime format and extracted the registration year.

We observed that the *manufactured* and *original_reg_date* columns almost always match in the year. Presumably, the vehicles that were manufactured in that year also get registered in the same year. Therefore, to fill in missing values for *manufactured*, we extracted the year from the *original_reg_date* datetime value, then casted *manufactured* as an integer so that it could be properly compared by any method later. We also casted *original_reg_date* and *reg_date* as integer timestamps after using them to fill in the missing values for *manufactured*.

To fill in the missing values for *make*, we observed that the *title* can be used. The *title* always starts with the make, so we extracted the first token of the *title*, converted it to lowercase, and used that to fill our *make*.

For *engine_cap*, *power* and *curb_weight*, we performed the same procedure to fill in missing values. First, if there were other listings of the same *make* and *model*, we take the mean of those listings to get our values. Otherwise, we take the mean of the listings with the same *type_of_vehicle*. We assumed that different cars of the same *make* and *model* should have the same *engine_cap*, *power* and *curb_weight*. However, for slightly different iterations of the same model, there may be variations. This is why we took the mean. If that fails, taking the mean for the same *type_of_vehicle* still makes sense as different *type_of_vehicle* will have different values for the missing columns. For example, a ‘suv’ is likely heavier than a ‘sports car’.

For *depreciation*, *coe*, *dereg_value*, *road_tax*, *mileage*, *omv* and *arf*, we performed the same procedure for missing values. We assumed that the data is representative of the actual distribution, so we performed random sampling from existing values in the combined training and testing data.

For *no_of_owners*, we observed that most listings have it as 1. Since these listings were for used cars, there must be at least one owner, so we set the missing values to be 1.

For *fuel_type*, we observed that most cars are still conventional internal combustion engine cars that take petrol, so we filled the missing values with ‘petrol’.

For *lifespan*, we noted that it marks the day that the vehicle must be deregistered. That day is 20 years from the *original_reg_date* for commercial vehicles, as cars in Singapore can only have their Certificate of Entitlement renewed up to 20 years. We also noted that it is highly unlikely for cars in Singapore to be renewed, so we simply filled in *lifespan* by adding 20 years to the *original_reg_date*.

For *eco_category*, all the records in the data had the same value of ‘uncategorized’. As for *indicative_price*, all the values were missing. We simply dropped these two columns.

Lastly, for *opc_scheme*, we noted there were very few records with values for it (only 207 in training data, and 58 in test data). We also dropped this column.

2) Feature Construction: After we had completed cleaning our data by filling in missing values and dropping unneeded columns, we began to construct the features that we would be using to train our models.

We used `sklearn`’s `OneHotEncoder` to one-hot encode the categorical features — *make*, *model*, *type_of_vehicle*, *fuel_type*, *transmission*. We dropped the original columns and used the one-hot encoded features instead.

We made sure all our dates were converted to integer timestamps — *manufactured*, *original_reg_date*, *reg_date*, *lifespan*.

Finally, we had all our numerical features ready to be used for training.

As for the *title*, *description*, *features*, *accessories*, *category* columns, they were strings that were tough to use although they indubitably contain useful information (e.g., description may contain significant qualities of the car that affects its price). To convert these strings into numerical features, we turned to pre-trained language models that can encode the string for us. This provided an easy method to compute fixed-length vectors for text. The text was embedded in vector space such that similar text strings were close to each other.

We made use of the sentence-transformers[1] library, using the pre-trained *all-mpnet-base-v2* model. This library contains pre-trained language models that are able to encode sentences into a fixed-size vector (in our case, the vector is of size 768). The transformer-based models achieve state-of-the-art performance in various tasks [1].

We applied the sentence encoder to the *title*, *description*, *features* and *accessories* columns, yielding four vectors of size 768. The numerical features alone, when concatenated together, are of dimension 861. When concatenated with the four vectors (encoded sentence features), the dimension of one sample becomes 3933.

In addition, we used GloVe (Global Vectors for Word Representation)[2], an unsupervised learning algorithm, to obtain vector representations for words in the descriptions. This makes sense because words like *(AMG, Luxury, Sports)* are

[1] https://github.com/UKPLab/sentence-transformers

[2] https://nlp.stanford.edu/projects/glove/

directly correlated to a higher price of the transaction. To establish a mapping between these words within the transaction description and the price point, we vectorized these words and trained a word embedding model on the price of the transaction to obtain a `sentiment score` - which is essentially the model's prediction of the price.

Similarly, we applied the same methods to obtain a separate title sentiment score on the title of transaction.

III. Task 1

A. Data Mining Methods

We tried three different families of methods from `sklearn`:

- Decision Trees — GradientBoostingRegressor, RandomForestRegressor, DecisionTreeRegressor
- k-Nearest Neighbours — KNeighborsRegressor
- Multi-layer Perceptrons — MLPRegressor

We used `sklearn`'s `GridSearchCV` to automatically search for the best model parameters with a 5-fold cross validation on the provided train set. Then, we submitted each model with the best parameters to Kaggle.

Finally, to identify the model with best parameters, we used an `XGBoost` ensemble approach mixed with hyperparameter optimization techniques. We trained multiple `XGBoostRegressors` and identified the top few models for bagging by leveraging `RandomSearchCV` (with a 5-fold cross validation) on a matrix of potential hyperparameters such as a range of values for `(learning_rate, sample_bycol, n_trees, etc)` and automating over a thousand trials to identify models with the best outcomes and its corresponding hyperparameters tunings.

To obtain the final prediction for this competition, or colloquially `y_pred`, we ran the 5 best-tuned models (our ensemble) from `RandomSearchCV` on the predictors and calculated the final values by means of a `sumaverage`.

B. Evaluation

Table I shows the comparison between the different models for the test loss (root mean squared error).

TABLE I
Model comparisons for test loss

Model	Test Loss (RMSE)
GradientBoostingRegressor	36931
DecisionTreeRegressor	60421
RandomForestRegressor	32286
KNeighborsRegressor	39730
MLPRegressor	30665
XGBoost Ensemble	29837
XGBoost Ensemble (w/ Embedding)	**27424**

As expected, the ensemble models performed the best with the lowest test losses. In addition, we saw that the inclusion of sentiment information from descriptions further improved the baseline ensemble model.

However, when we took training time (including both `GridSearch` and `RandomSearch`) into account, we found that there were clear time-performance tradeoffs between these methods.

`XGBoost` Ensembles took the longest at roughly 30 minutes to 1 hour. `GradientBoostingRegressor` took 23 minutes, then `MLPRegressor` at 15 minutes, `RandomForestRegressor` at 8 minutes, `DecisionTreeRegressor` at 1 minute and `KNeighborsRegressor` at 30 seconds.

All in all, for high accuracy, we recommend an ensemble approach; for a quick estimate, we recommend `KNeighborsRegressor`; and for a middle ground, we believe `RandomForestRegressor` (optimized with `GridSearch`) would be the best. It achieved an acceptable low loss (ranked 4th out of 7) with a reasonable run time (ranked 3rd out of 7).

IV. Task 2

For this task, we made an assumption that users would want recommendations for similar cars in the same price range. Therefore, we defined the quality of our recommendations by:

- Price (recommended cars should not be too different in price from chosen car)
- Type/power of car (recommended cars should have similar bodies and performance characteristics)
- Variety (recommendations should not solely be the same make and model but different listing)

It would not be good to recommend a car twice as expensive as the user's budget, while it would be prudent to recommend something similar (i.e., do not recommend a truck to someone looking for a hatchback), i.e., with similar power/engine characteristics (i.e., fast vs slow car). Lastly, the recommendations should have some variety, instead of only recommending the same make and model but of a different listing.

A. Preprocessing

We used the cleaned data from before, and made some changes to it to better suit our task.

For this task, we did not use the features encoded by sentence-transformers. Firstly, this simplified the interpretation of our results. Secondly, these features were written by the seller, and were not indicative of the 'true' nature of the car (i.e., the same car in different listings can have wildly different descriptions). Therefore, these features might reduce the quality of our recommendations.

We also dropped the *make* and *model* columns for this task. We noticed that the usage of these features was affecting the recommendations too strongly – the returned recommendations were only the same make and model. This reduces the quality of the recommendations as it provides no variety.

Since we intended to use Nearest Neighbours in the vector space to find similar listings, we scaled all the numerical input features using the `MinMaxScaler` to restrict the values between 0 and 1. This ensured that features with naturally large numerical values would not automatically dominate the positioning of our samples in vector space.

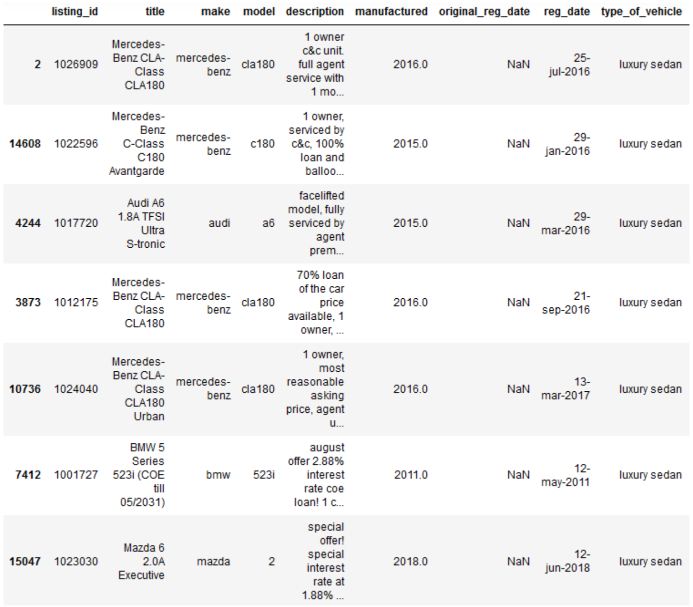
We *overweighted* the price feature to give more emphasis on recommending cars within the same price range.

B. Data Mining Methods

We use `k-Nearest Neigbours` to identify listings similar to the listing searched by the user.

C. Evaluation & Interpretation

Since there were no ground-truth answers for recommendation quality (good recommendation is subjective), we obtained the recommendations for a few different listings and did a manual inspection of the recommendations to check if they matched our idea of ‘good’, as defined earlier.



	listing_id	title	make	model	description	manufactured	original_reg_date	reg_date	type_of_vehicle
2	1026909	Mercedes-Benz CLA-Class CLA180	mercedes-benz	cla180	1 owner c&c unit. full agent service with 1 mo...	2016.0	NaN	25-jul-2016	luxury sedan
14608	1022596	Mercedes-Benz C-Class C180 Avantgarde	mercedes-benz	c180	1 owner, serviced by c&c, 100% loan and balloo...	2015.0	NaN	29-jan-2016	luxury sedan
4244	1017720	Audi A6 1.8A TFSI Ultra S-tronic	audi	a6	facelifted model, fully serviced by agent prem...	2015.0	NaN	29-mar-2016	luxury sedan
3873	1012175	Mercedes-Benz CLA-Class CLA180	mercedes-benz	cla180	70% loan of the car price available, 1 owner, ...	2016.0	NaN	21-sep-2016	luxury sedan
10736	1024040	Mercedes-Benz CLA-Class CLA180 Urban	mercedes-benz	cla180	1 owner, most reasonable asking price, agent u...	2016.0	NaN	13-mar-2017	luxury sedan
7412	1001727	BMW 5 Series 523i (COE till 05/2031)	bmw	523i	august offer 2.88% interest rate coe loan! 1 c...	2011.0	NaN	12-may-2011	luxury sedan
15047	1023030	Mazda 6 2.0A Executive	mazda	2	special offer! special interest rate at 1.88% ...	2018.0	NaN	12-jun-2018	luxury sedan

Fig. 4. Recommendations for row 2 in train set

We checked the returned recommendations, made sure that the recommendations were within the same price bracket, contained a good mix of recommendations (not all the same cars), and that the recommendations were similar (same kind of car, similar engine/power etc.).

As we can see from Figure 4, our recommendations, according to our evaluation criteria, are quite good. There is a good mix of cars (Mercedes, Audi, Mazda, BMW) of the same type and in the same price bracket.

Nonetheless, there is one major drawback of our recommendations. Because we emphasized returning results within the same price range, a user would not be able to use our method to find bargains (i.e., similar cars but at much lower price points). However, assuming the market is competitive, this should not be an issue.

V. Task 3

A. Dealer Analysis

1) Motivation: After browsing through used car listings on SgCarMart, we realised most listings were through a car Dealer. Purchasing a used car from a Dealer would be preferable for most buyers, since Dealers would help clean, fix-up the used car and take care of the paperwork.

The SgCarMart website provides the details of the Dealer for each individual listing. It also has a directory of all used car Dealers, which only contains information provided by the dealer themselves, and the list of cars currently on sale or sold in the last 3 months by the Dealer.

Given that most buyers would ultimately purchase a car from a Dealer, it seemed to us that the popularity, reliability and price competitiveness of a Dealer would factor into the buyer’s decision making process. However, this information is not directly available on SgCarMart. For each listing that the buyer was interested in, they would have to look up the Dealer online to gather reviews.

For this task, our aim was to use the available data for Dealers on SgCarMart to be able to address the following requirements for a buyer:

- As someone looking to buy a certain type of used car (say Luxury Sedan or SUV), I would like to know which Dealers specialise in the sale of that type of vehicle.
- Given a set of listings from SgCarMart matching my search criteria, I want to be able to sort them by a metric indicative of the popularity, reliability and price competitiveness of the Dealer

2) Data Collection: Since the dataset provided as part of the project does not contain any information about Dealers, we had to extract the data we needed from the SgCarMart website ourselves. We scraped the following data points:

1) The list of used car Dealers on the SgCarMart. We needed at least two attributes: Dealer Name (for display purposes) and Dealer Id (for uniquely identifying a Dealer and querying for other details below)
2) The list of used cars on sale for each Dealer, along with the type of car (Luxury, SUV, Bus etc.)
3) The list of used cars sold in the past 3 months for each Dealer, along with the type of car (Luxury, SUV, Bus etc.)
4) The Dealer Id for each listing in our training dataset

We decided to use the number of cars on sale currently/sold in the past as a proxy for how popular and reliable a Dealer is.

In order to determine price competitiveness, we decided to use our price prediction model from Task 1 to estimate the price for all the listings in our training dataset and then compare it to the actual price to determine if it was being sold at a discount or premium. Then, by associating each listing to a Dealer, we could determine whether the Dealer gave good deals or not on average.

3) Exploratory Data Analysis: As one might anticipate, not all Dealers sell all types of vehicles, probably as a function of demand and supply. Roughly half the Dealers sell Luxury vehicles and Sedans, while only a handful sell Buses.

The correlation heatmap (Figure 5) indicates that there is some correlation between certain vehicle types sold by Dealers. Dealers selling buses also tend to sell other commercial

TABLE II
NUMBER OF DEALERS SELLING USED CARS, BY VEHICLE TYPE
('RECENTLY' REFERS TO THE PAST 3 MONTHS)

Vehicle Type	#dealers with available vehicles	#dealers with vehicles sold recently
Bus	34	26
Hatchback	496	551
Luxury	600	611
MPV	487	537
SUV	577	586
Sedan	555	606
Sports	455	442
Stationwagon	202	220
Truck	118	87
Van	179	143
Others	12	1

purpose vehicles such as vans and trucks, while Dealers selling vehicles for private use such as Luxury also sell SUVs and Sports vehicles.

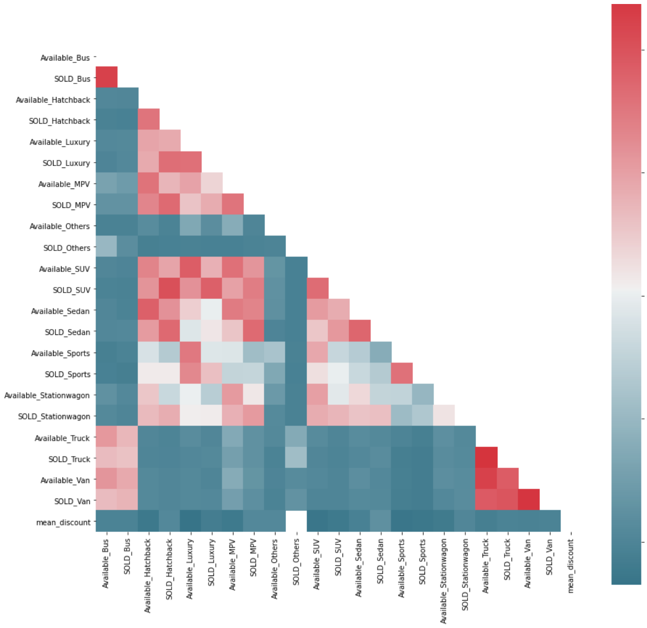


Fig. 5. Correlation Matrix of Numerical Features for Dealers

There is also a noticeable correlation between the number of vehicles available and the number of vehicles sold (as seen in Table II, however that does not mean we can simply substitute one for the other in our metric, since the number of available cars represents supply, while number of sold cars represents "popularity" among other buyers.

The distribution of Dealers by number of vehicles available (for a particular vehicle type) is fairly typical as shown in the histogram (Figure 6). Majority of the Dealers have single-digit number of vehicles available, with the distribution having a long tail as the number of Dealers selling more vehicles is fewer. This highlights the presence of a small group of Dealers having much greater choice and availability, specialising in certain vehicle types, that the buyer would be more interested in approaching.

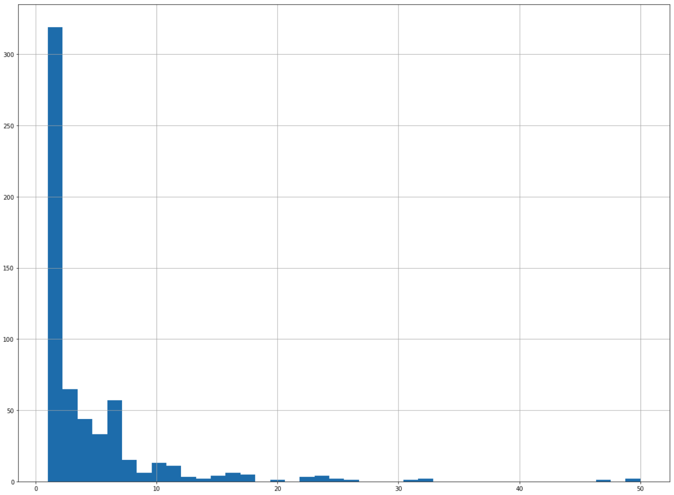


Fig. 6. Histogram of Dealers by number of luxury vehicle available

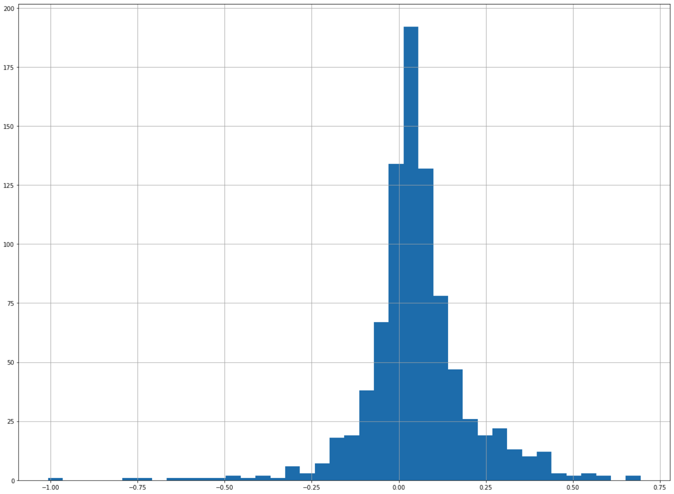


Fig. 7. Histogram of Average Discount on car prices by Dealers

The histogram of the mean discount offered (Figure 7) appears to be a somewhat normal distribution centered roughly around zero. However, we should take this distribution with a pinch of salt since we calculate the discount as the percentage difference between the actual listing price and the price as predicted by our Task 1 Model. In fact, this distribution is technically an error distribution of our price predictions for the training dataset. Nevertheless, for the purpose of Task 3, we treat our predicted price as the true value of the car, and the listing price as having a discount or premium on that true value.

4) Methodology: Based on the data collected above, we were able to construct the following features for each Dealer:

1) Number of available cars, by vehicle type (A_v): This is a rough measure of how popular they are as a Dealer of used cars, since that many people have chosen to sell their cars to them
2) Number of cars sold in the past 3 months, by vehicle type (S_v): This is a proxy for how reliable they are, since that many buyers have chosen to purchase cars

from them

3) Average discount applied across the listings available in the training dataset (D): This is a measure of their price competitiveness
4) Total number of listings for the Dealer in the training dataset (C): This is factored in to balance exaggerated discounts in the case where the dealer has very few listings

We normalized the 4 features above using a MinMaxScaler to restrict all the numerical values to range between 0 and 1.

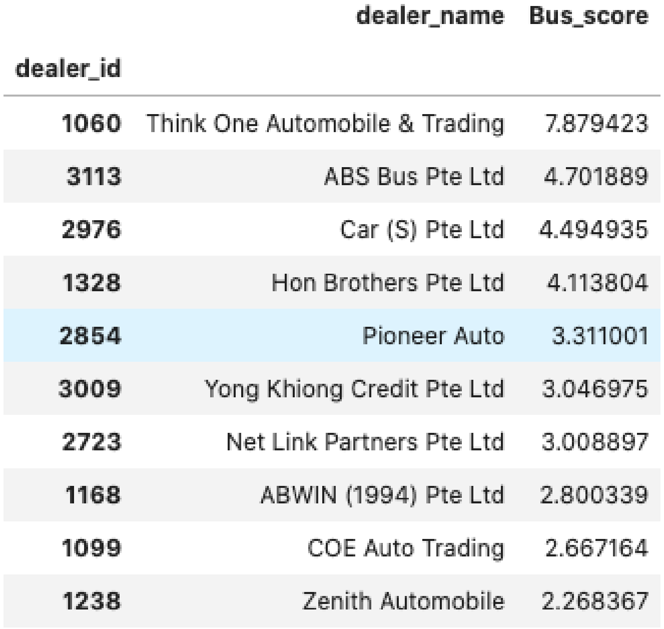
We then calculate a rating for each Dealer, for each vehicle type as:

$$\text{if } A_v > 0 : [R_v = 3 \times A_v + 3 \times S_v + 3 \times D + C]$$
$$\text{else, } [R_v = 0]$$

The above formula gives us a rating out of 10 for each Dealer and each vehicle type.

This rating can also be incorporated into our recommendation. Our recommendation of similar listings can now include the Dealer information (where available), and the order of the recommendations can be in decreasing order of Dealer ratings.

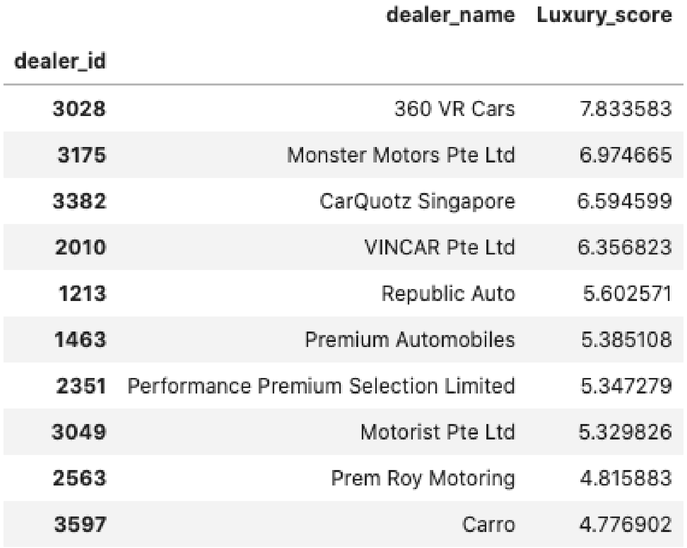
5) Evaluation: After calculating the ratings for each Dealer by vehicle type, we can now query which Dealers have the highest ratings for a particular vehicle type. Figure 8 and Figure 9 show the top dealers for buses and luxury vehicles.



dealer_id	dealer_name	Bus_score
1060	Think One Automobile & Trading	7.879423
3113	ABS Bus Pte Ltd	4.701889
2976	Car (S) Pte Ltd	4.494935
1328	Hon Brothers Pte Ltd	4.113804
2854	Pioneer Auto	3.311001
3009	Yong Khiong Credit Pte Ltd	3.046975
2723	Net Link Partners Pte Ltd	3.008897
1168	ABWIN (1994) Pte Ltd	2.800339
1099	COE Auto Trading	2.667164
1238	Zenith Automobile	2.268367

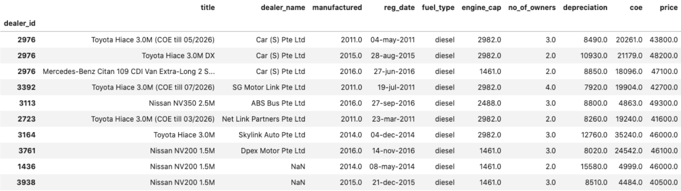
Fig. 8. Top-10 Dealers for Buses, with the last column indicating their rating

A quick lookup of these Dealers on the SgCarMart website as well as looking them up online validates that these are indeed among the most popular Dealers in Singapore for those categories. Many of them also feature on SgCarMart's list of *Premium Dealer* winners (although it is not clear what the criteria for the *Premium Dealer* label is).



dealer_id	dealer_name	Luxury_score
3028	360 VR Cars	7.833583
3175	Monster Motors Pte Ltd	6.974665
3382	CarQuotz Singapore	6.594599
2010	VINCAR Pte Ltd	6.356823
1213	Republic Auto	5.602571
1463	Premium Automobiles	5.385108
2351	Performance Premium Selection Limited	5.347279
3049	Motorist Pte Ltd	5.329826
2563	Prem Roy Motoring	4.815883
3597	Carro	4.776902

Fig. 9. Top-10 Dealers for Luxury Vehicles, with the last column indicating their rating



dealer_id	title	dealer_name	manufactured	reg_date	fuel_type	engine_cap	no_of_owners	depreciation	coe	price
2976	Toyota Hiace 3.0M (COE till 05/2026)	Car (S) Pte Ltd	2011.0	04-may-2011	diesel	2982.0	3.0	8490.0	20261.0	43800.0
2976	Toyota Hiace 3.0M DX	Car (S) Pte Ltd	2015.0	28-aug-2015	diesel	2982.0	2.0	10930.0	21179.0	48200.0
2976	Mercedes-Benz Citan 109 CDI Van Extra-Long 2 S...	Car (S) Pte Ltd	2016.0	27-jun-2016	diesel	1461.0	2.0	8850.0	18096.0	47100.0
3392	Toyota Hiace 3.0M (COE till 07/2026)	SG Motor Link Pte Ltd	2011.0	19-jul-2011	diesel	2982.0	4.0	7920.0	19904.0	42700.0
3113	Nissan NV350 2.5M	ABS Bus Pte Ltd	2016.0	27-sep-2016	diesel	2488.0	3.0	8800.0	4863.0	49300.0
2723	Toyota Hiace 3.0M (COE till 03/2026)	Net Link Partners Pte Ltd	2011.0	23-mar-2011	diesel	2982.0	2.0	8260.0	19240.0	41600.0
3164	Toyota Hiace 3.0M	Skylink Auto Pte Ltd	2014.0	04-dec-2014	diesel	2982.0	3.0	12760.0	35240.0	46000.0
3761	Nissan NV200 1.5M	Dpex Motor Pte Ltd	2016.0	14-nov-2016	diesel	1461.0	3.0	8020.0	24542.0	46100.0
1436	Nissan NV200 1.5M	NaN	2014.0	08-may-2014	diesel	1461.0	2.0	15580.0	4999.0	46000.0
3938	Nissan NV200 1.5M	NaN	2015.0	21-dec-2015	diesel	1461.0	3.0	8510.0	4484.0	40500.0

Fig. 10. Recommendations for listing ID (1021510) Toyota Hiace (2015), ranked by Dealer rating for Van category

As mentioned earlier, we can now also include the Dealer information in our list of recommendations. Further, we can sort the recommendations from Task 2 by the Dealer rankings, meaning that the top recommendations are from highly rated Dealers (Figure 10 shows an example). This helps buyers to prioritize their search, and opt for directly getting in touch with the top Dealers that have models similar to what they searched for.

6) Limitations: Our Dealer ratings model in its current implementation has a number of limitations:

- Missing Data: Some of the Dealers that we scraped off of SgCarMart do not have any listings in the training dataset. For these dealers, we chose to measure average discount as 0. Ideally, we should have attempted to scrape the data needed for price prediction for each listing associated with a Dealer, instead of relying on the training dataset which is only a subsample. We settled for the training dataset due to time constraints. Out of 1149 total used car Dealers on SgCarMart, we managed to find listings for only 868 Dealers in the training dataset
- Propagated limitations of Task 1 Model: Our price prediction model from Task 1 is not perfect, and using it to determine whether a listing is priced at a discount or

premium would definitely not be accurate. Further, since our model is trained on the same dataset, over-fitting issues could mean that our predictions are too close to the actual price and do not actually detect a deviation. A better way of estimating price competitiveness might be to compare with similar listings from other Dealers

- Arbitrary weights in the calculation of ratings: The weights assigned to the inputs in the ratings formula is arbitrary. We've intuited the weights roughly based on how much importance we (a group of students that have never purchased a car in Singapore) would place on the different parameters, but the approach needs to be more rigorous. Ideally, the weights would be learnable parameters in a supervised learning setup where we would already have a ratings dataset for Dealers, that we could train our model on. In fact, an extension of this project could be to scrape ratings for the Dealers from SgCarMart (a handful of Dealers do have ratings), or even scrape it off of social media.

B. *Popularity of Eco-Friendly Cars*

1) Motivation: As per a report by the International Energy Agency (IEA), there has been a 41% increase in the number of electric car registrations done globally, over the year 2020 [2]. This is indeed a good trend, given the rise in global temperatures and awareness on the ongoing climate crisis. As global electric car sales continue to grow, we were curious to evaluate if a similar trend was visible in the car resale data from SgCarMart.

In this section, we look for trends on the car market with regards to the popularity and price disparity among cars, categorising them as eco-friendly and non-eco-friendly. As discussed towards the end of this section, this information could be of use to governmental organizations and environmental agencies for policy-making and reporting. In the following subsections, we compare the volume of cars sold in each of the four fuel type categories present in the dataset. We also look at the difference in the average pricing across these categories. For this analysis, no additional data was collected and we used the training dataset provided for Task 1. As the `eco-category` attribute in the dataset had only the `uncategorized` value, we chose the `fuel_type` attribute to categorise a given model as Eco-Friendly or Non-Eco-Friendly. To that end, cars with `fuel_type` set to *electric* or *petrol-electric* were considered Eco-Friendly while *diesel* and *petrol* ones were counted as Non-Eco-Friendly.

2) Observations: Figure 11 shows the proportion of cars sold in each of the *electric*, *petrol-electric*, *diesel* and *petrol* categories. It is evident that the proportion of Eco-Friendly cars is very small in comparison to the other conventional Non-Eco-Friendly ones. Moreover, the proportion of Electric cars within the Eco-Friendly category is even tinier.

Note that despite the volume of electric cars sold being so low, the average price in this category was still the highest, as seen in Fig. 12. The average values reported for each fuel type category were weighted averages over the `type_of_vehicle` categories, with the number of cars within the latter category acting as weights. For example, when calculating the average price for the Electric (fuel type) category, we averaged over all mean price values within the *hatchback*, *luxury sedan*, *mid-sized sedan*, *mpv*, *sports car*, *suv* and *van* subcategories, with the number of cars sold in each of these subcategories applied as weights to those mean price values. Note that for calculating these values, we considered only those subcategories which were present in **all** four of the fuel types, to ensure consistency when arriving at the average values.

Since averages abstract away the underlying distribution, Figures 13 and 14 show the proportion of volumes sold within

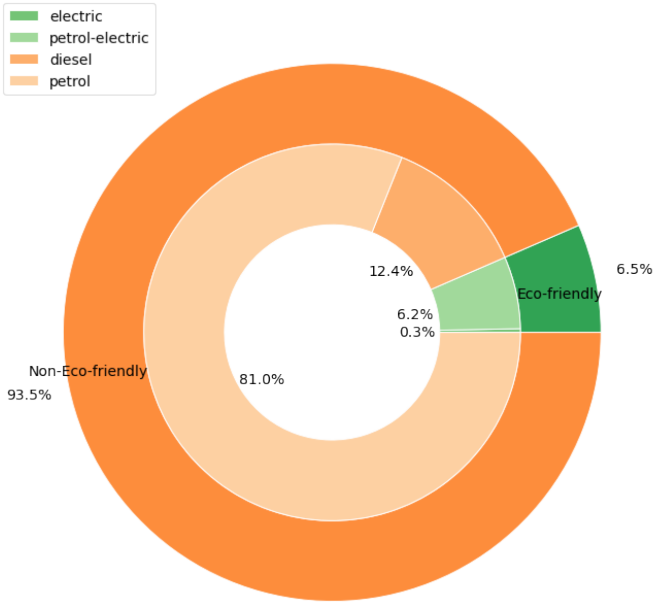


Fig. 11. Volume of cars in each `fuel_type` category, with the outer ring showing the eco-friendly and non-eco-friendly categorisation

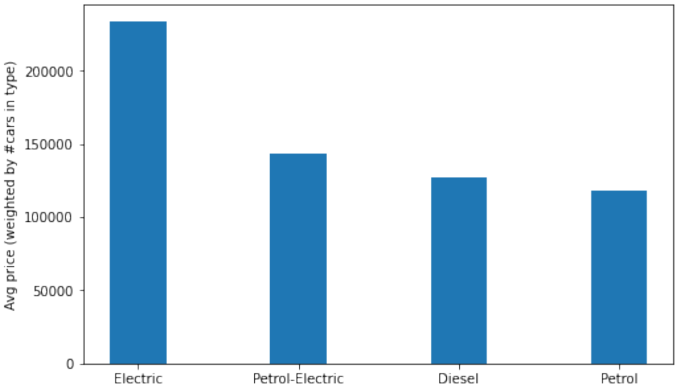


Fig. 12. Weighted average price across the `fuel_type` categories, weighted by the #cars sold within the subcategory (sedan, hatchback, etc.)

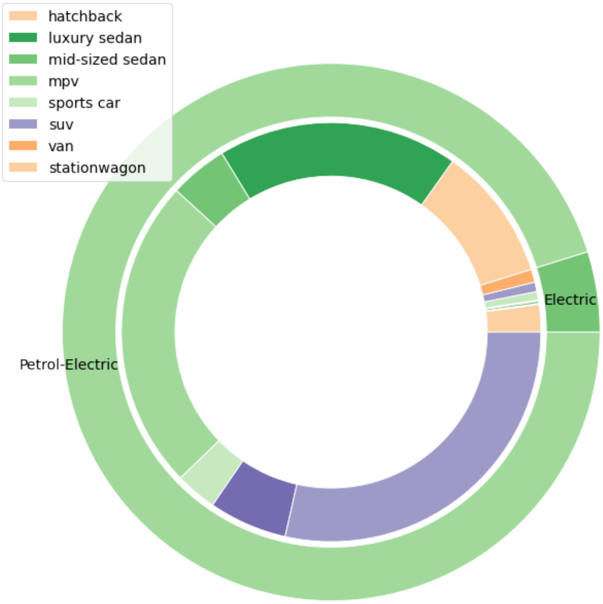


Fig. 13. Volume of cars within the Eco-Friendly category, with the outer ring showing the Electric and Petrol-Electric categorisation

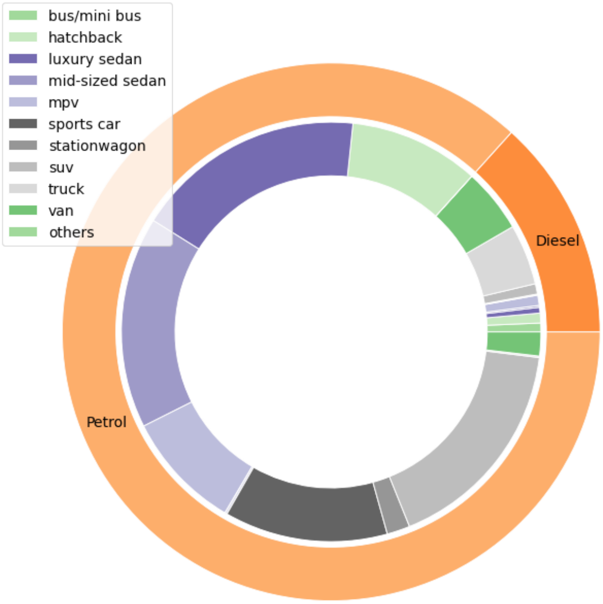


Fig. 14. Volume of cars within the Non-Eco-Friendly category, with the outer ring showing the Diesel and Petrol categorisation

the four fuel type categories, with the fuel type categorisation shown on the outer ring. While the subcategories considered when calculating the average prices were those common across the four fuel types, we show all subcategories in these figures for completeness.

To summarise, what we conclude from this analysis is that conventional fuel types, i.e., petrol and diesel continue to be the popular choices, with them taking over 90% of the share, at least in the given dataset. The disparity in the (average) price for these categories also supports this choice, as the consumer is disincentivized by having to pay more for the Eco-Friendly alternatives.

3) Limitations and Discussion: Given our price comparison across the four fuel type categories, it'd have been better to draw such a comparison across each of the subcategories (*hatchback*, *sedan*, etc.). That could have helped cater to specific consumer segments. For instance, looking at the *hatchback* subcategory across fuel types could help determine what sort of a subsidy or tax credit may help trigger a behavior change in that specific segment. This is because a consumer buying a *hatchback* may be willing to switch to an Eco-Friendly alternative, within the subcategory owing to her capacity and usage needs. However, some of the subcategories within the fuel-types had significantly low representation, restricting us from conducting the analysis at that granularity. Given the difference in average price from Fig. 12, this analysis could be complimented with other policy data to arrive at the incentive mechanism necessary in order to nudge consumer behavior towards choosing Eco-Friendly alternatives. For instance, several states in India have a 100% road-tax exemption for electric two-wheelers, along with subsidies per kWh of battery capacity with varying caps [3]. Interestingly, the Land Transport Authority (LTA) of Singapore introduced several incentives and schemes to encourage early adoption of electric vehicles [4] in early 2021. As per these, the ARF floor would be lowered from $5,000 to $0, for fully electric cars and taxis registered from 1 January 2022 to 31 December 2023, with revisions and reductions in Road Tax for existing electric cars as well.

C. Depreciation Analysis

1) Motivation: Cars in Singapore are extremely expensive to purchase and maintain [5]. There are several main factors – some unique to Singapore – that contribute to the high prices. For example, Open Market Value (OMV), Additional Registration Fee (ARF), Excise Duty & GST, Certificate of Entitlement (COE), Vehicular Emission Scheme (VES) rebate or surcharge, and dealers' margin. Of which, the COE is a certificate that allows a car to be driven on Singapore road for 10 years, after which, the owner has to either deregister the car or pay again to renew the COE for another 5 or 10 years. The costs of COE vary across car categories and are market-driven, but a typical COE for a sedan car in 2021 costs around $30,000 to $50,000 [6] – which is sometimes more costly than the car itself. Due to the fact that a brand-new COE is only good for 10 years, car values in Singapore reduce significantly over a 10 year period – a very high depreciation.

Given this, many car buyers in Singapore are more concerned about the annual depreciation than the price of the car at the time of purchase. For instance, if a car costs $100,000 now and depreciates by only $10,000 after 10 years, versus another that costs $50,000 now but depreciates by $40,000 after 10 years, we believe many people, given enough initial capital, would choose the first car even though the initial cost is higher.

Therefore, it would be very helpful for the users if sgCarMart can include some statistics on depreciation on their website. Currently, it is possible to see the depreciation per car if the seller provides it and sort listings by this metric. However, there is no information regarding what makes, models, or bodies on average depreciate the least or most. In this subsection, as a simple exploration, we look at depreciation by make.

2) Data: For this task, we merged the train and test datasets provided, as this analysis did not involve *prices*.

3) Data Cleaning: For data cleaning, we first filled in the missing data in *make* using the method described in Task 1. We then dropped observations that did not have depreciation values. Lastly, we only kept columns that were useful to this analysis, i.e., *make* and *depreciation*.

4) Data Analysis: To examine the relationship between make and depreciation, we first grouped *depreciation* by *make*. We then obtained a summary table with *count*, *mean*, *min*, and *max* over the depreciation values per make. Lastly, we removed rows with fewer than 50 counts, and sorted the mean depreciation values from high to low (Figure 15).

5) Results: The resulting table (in Appendix B) included 33 different car makes. Makes with the lowest depreciation are *suzuki*, *chevrolet*, *opel*, *hyundai*, and *citroen*; while makes with the highest depreciation are *ferrari*, *lamborghini*, *bentley*, *porsche*, and *maserati*.

6) Discussion and Limitations: The rankings we obtained were not surprising at all. In fact, a simple table like this on the website with summary statistics based on past transactions would be very helpful for the users to compare similar brands. However, since this was a very simple proof-of-concept, it was not without its limitations. As we can infer from the rankings, it seems that car price directly affects the absolute depreciation value, and we are unable tell the relative depreciation from this table alone. We also included cars from all years. If we allow users to choose their preferred age range, the ranking would likely be slightly different.

References

[1] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Nov. 2019. [Online]. Available: https://arxiv.org/abs/1908.10084.

[2] *IEA (2021), Global EV Outlook 2021, IEA, Paris*. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2021.

[3] A. Ahmed, "State-wise EV subsidies in India: A handy list of incentives and benefits for electric vehicles in each state," *FirstPost*, Sep. 13, 2021. [Online]. Available: https://www.firstpost.com/tech/auto-tech/state-wise-ev-subsidies-in-india-a-handy-list-of-incentives-and-benefits-for-electric-vehicles-in-each-state-9952771.html (visited on 11/13/2021).

[4] "Factsheet: Encouraging the Adoption of Electric Cars for a More Sustainable Land Transport Sector," Mar. 4, 2021. [Online]. Available: https://www.lta.gov.sg/content/ltagov/en/newsroom/2021/3/news-release/Encouraging_the_adoption_of_electric_cars.html (visited on 11/13/2021).

[5] *An explanation on why cars in singapore are so expensive (2021)*. [Online]. Available: https://dollarsandsense.sg/no-nonsense-explanation-on-why-cars-in-singapore-are-so-expensive/.

[6] *COE Prices Since 2002*. [Online]. Available: https://coe.sgcharts.com/.

VI. Appendix A

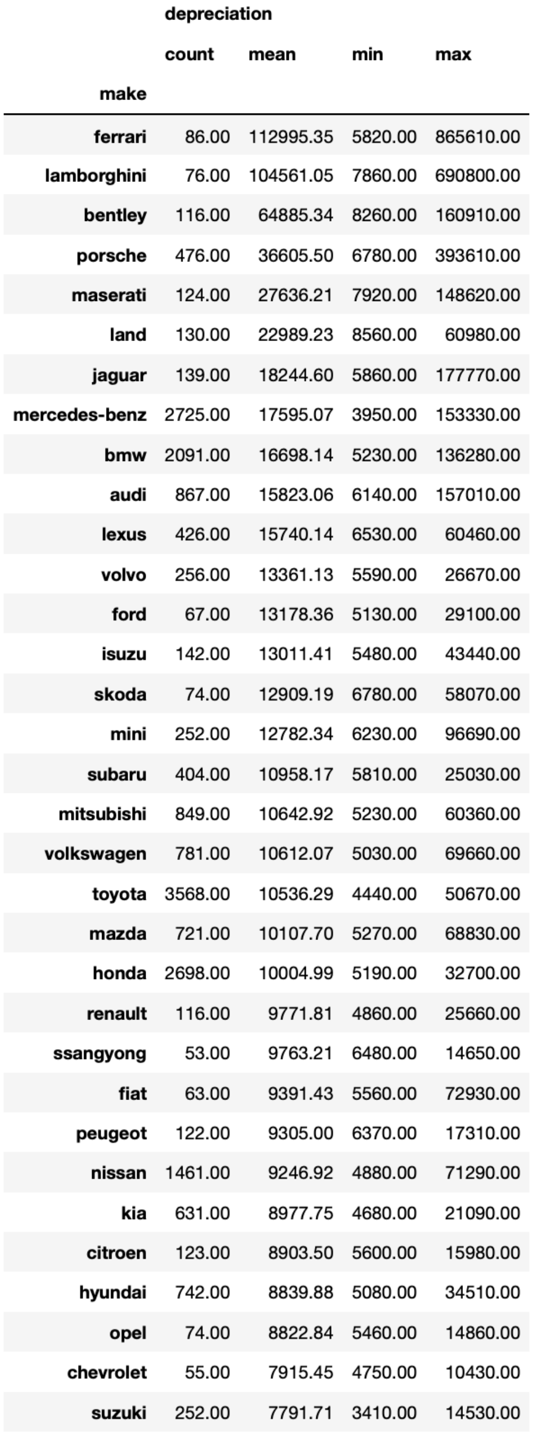
A. Work Breakdown

TABLE III
WORK BREAKDOWN

Parts	Members			
	Anand	Chaitanya	Hoki	Lok You
EDA	✓	✓	✓	✓
Preprocessing	✓		✓	✓
Task 1	✓	✓	✓	✓
Task 2	✓			✓
Task 3	✓	✓	✓	
Report	✓	✓	✓	✓

VII. Appendix B

A. Additional Figure



	depreciation			
	count	mean	min	max
make				
ferrari	86.00	112995.35	5820.00	865610.00
lamborghini	76.00	104561.05	7860.00	690800.00
bentley	116.00	64885.34	8260.00	160910.00
porsche	476.00	36605.50	6780.00	393610.00
maserati	124.00	27636.21	7920.00	148620.00
land	130.00	22989.23	8560.00	60980.00
jaguar	139.00	18244.60	5860.00	177770.00
mercedes-benz	2725.00	17595.07	3950.00	153330.00
bmw	2091.00	16698.14	5230.00	136280.00
audi	867.00	15823.06	6140.00	157010.00
lexus	426.00	15740.14	6530.00	60460.00
volvo	256.00	13361.13	5590.00	26670.00
ford	67.00	13178.36	5130.00	29100.00
isuzu	142.00	13011.41	5480.00	43440.00
skoda	74.00	12909.19	6780.00	58070.00
mini	252.00	12782.34	6230.00	96690.00
subaru	404.00	10958.17	5810.00	25030.00
mitsubishi	849.00	10642.92	5230.00	60360.00
volkswagen	781.00	10612.07	5030.00	69660.00
toyota	3568.00	10536.29	4440.00	50670.00
mazda	721.00	10107.70	5270.00	68830.00
honda	2698.00	10004.99	5190.00	32700.00
renault	116.00	9771.81	4860.00	25660.00
ssangyong	53.00	9763.21	6480.00	14650.00
fiat	63.00	9391.43	5560.00	72930.00
peugeot	122.00	9305.00	6370.00	17310.00
nissan	1461.00	9246.92	4880.00	71290.00
kia	631.00	8977.75	4680.00	21090.00
citroen	123.00	8903.50	5600.00	15980.00
hyundai	742.00	8839.88	5080.00	34510.00
opel	74.00	8822.84	5460.00	14860.00
chevrolet	55.00	7915.45	4750.00	10430.00
suzuki	252.00	7791.71	3410.00	14530.00

Fig. 15. Mean, Min, and Max Depreciation By Car Make

VIII. Appendix C

A. Submission Folder Structure

```
Group Submission Folder
 ├─ Data
 │   ├─ data_given
 │   │   ├─ train.csv
 │   │   └─ test.csv
 │   └─ data_additional
 │       ├─ (19 files)
 │       └─ ...
 ├─ Report.pdf
 ├─ Task_1_gridsearch.ipynb
 ├─ Task_1_ensemble_randomsearch.ipynb
 ├─ Task_2_knn.ipynb
 ├─ Task_3_A.ipynb
 ├─ Task_3_B.ipynb
 └─ Task_3_C.ipynb
```

Note: Our submission folder includes the final report, six independent scripts for the three tasks in this final project, and a data folder that includes the given dataset and additional data we used in Task 2 and Task 3.