CSE 587 Phase II

2 Algorithms / person (at least 1 from outside class)

Algorithms discussed in class are: Linear Regression, k-Means, k-NN, Naive Bayes, Logistic Regression and Decision Tree.

Jiabao Yao: SelectKBest, GridSearchCV, RandomForestRegressor, LinearRegression, CatBoostClassifier

Refined Questions

During phase II, some of the initial questions from phase I were refined to reveal deeper insights and more underlying relationships between different features. This refinement process provided a clearer understanding of the problem statement, supporting the development of an accurate price predictor.

Refined Question 1: What features could be affected by mileage? (Te Shi)

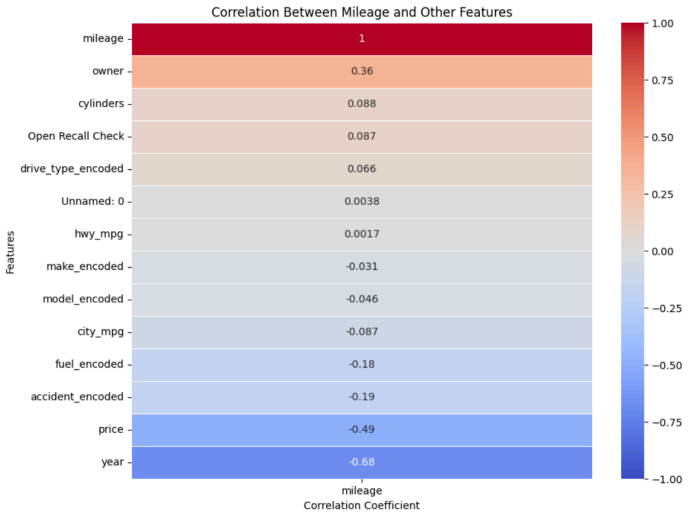
Mileage is a crucial determinant of a car's resale value. In Phase I, the analysis focused on how mileage affects resale value across different car brands, concluding that, in general, resale value negatively correlates with mileage and that this trend varies across brands.

To gain further insights about this important feature, it is essential to examine the relationship between mileage and additional features. The correlations found can serve as indicators of how mileage influences a car's overall condition, potentially explaining why increased mileage leads to decreased prices. Additionally, understanding these correlations could enable the model to estimate mileage based on other related features when exact mileage data is unavailable.

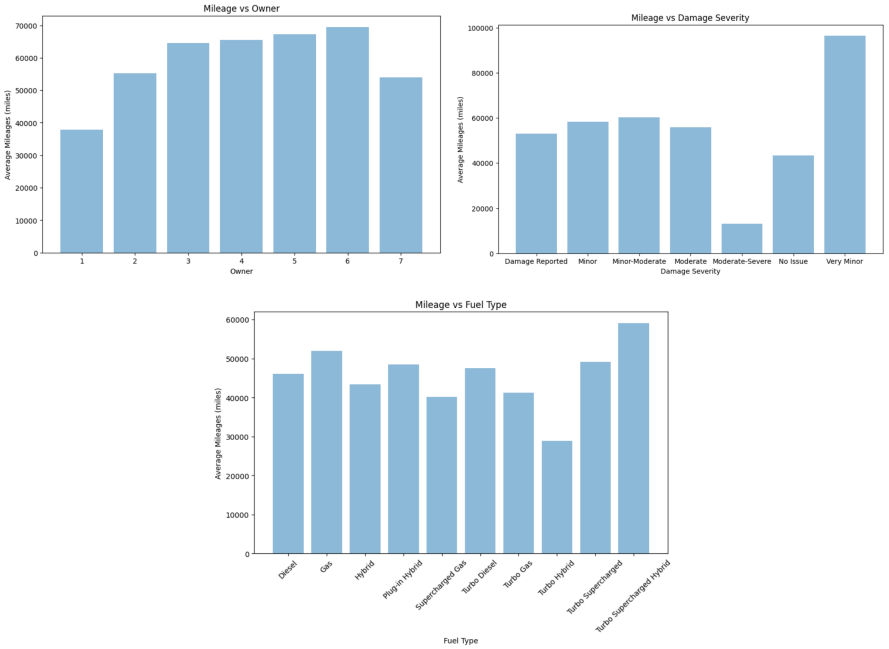
Hypothesis 1 & 2

The first two hypotheses remains the same as phase I, study the relationship between mileage and resale price across different brands

Hypothesis 3: other features such as mileage per gallon and owners may also have a correlation with mileage.



The graph above shows the correlation between mileage and other features. Features like VIN and color, assumed to be less relevant, were excluded based on intuition. The results indicate that, aside from price, other features with a relatively significant impact on mileage include year, owner, accident history, and fuel type.



The graphs above illustrate the relationships between mileage and the factors of ownership count, damage severity, and fuel type. The data suggests that mileage generally increases with the number of owners, possibly due to more usage across multiple drivers. In terms of damage severity, cars with very minor damage show significantly higher average mileage compared to those with more severe damage, indicating that extensive damage might reduce a car's lifespan or usage. For fuel type, average mileage is relatively consistent among common types like diesel and gas, while less common fuel types show more variation. This relationship warrants further exploration in phase II.

Refined Question 2: Do different car classes exhibit significant and distinct characteristics across features? (Te Shi)

Original Question 2 focused on exploring the relationship between different car classes and their resale value. In Phase II, I feel it is necessary to investigate additional significant characteristics present in different car classes. Phase I's EDA revealed that the average resale price varies among car classes. By gaining a deeper understanding of other characteristics associated with each car class, we can achieve more insightful analysis and better identify car classes. This understanding could be crucial for the final price prediction model, as car class has a noticeable correlation with price.

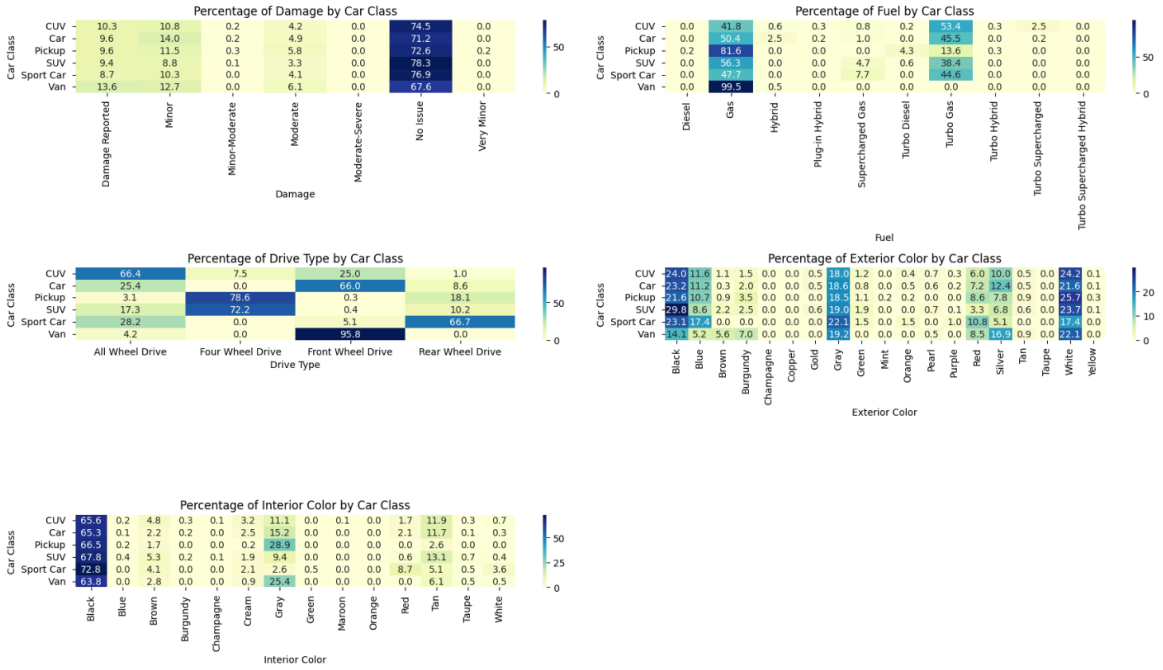
Hypothesis 1&2: General Resale Price is different across different car class

This part is same as EDA for original question 2

Hypothesis 3: Other features like mileage and fuel efficiency could be significant different across different classes

car_class	year	price	mileage	city_mpg	hwy_mpg	cylinders	owner	Open Recall Check
CUV	2020.0	24998.0	39000.0	22.0	28.0	4.0	1.0	0.0
Car	2019.0	19998.0	46000.0	26.0	35.0	4.0	2.0	0.0
Pickup	2020.0	33998.0	42000.0	17.0	22.0	8.0	1.0	0.0
SUV	2020.0	36998.0	40000.0	18.0	23.5	6.0	1.0	0.0
Sport Car	2019.0	37998.0	31000.0	18.0	27.0	6.0	2.0	0.0
Van	2018.0	22998.0	62000.0	19.0	28.0	6.0	2.0	0.0

Quantitative features were examined firstly. The table above displayed the median values of these quantitative features across different car classes. Some features, like the *owner* and *open recall check*, appear consistent across car classes without significant variation. However, some features differ significantly. For example, a class type car, which means sedan type vehicle, has the lowest median price and a notable price gap compared to sport cars. Furthermore, in terms of mileage per gallon, CUV and sedans show higher values reflecting their cost-effectiveness. In addition, the number of cylinders for pickup is higher as these vehicles require more horsepower for carrying heavier loads. generally more than others since it requires more horsepower to carry load. SUVs, sports cars, and vans tend to have more cylinders than CUVs and sedans, likely due to performance and utility demands.



The heat maps above illustrate the percentage distribution of various feature values across different car classes, specifically focusing on damage, fuel type, drive type, exterior color and interior color. The results reveal that certain features display similar distributions across car classes, suggesting they may not be critical in distinguishing car types. However, other features exhibit clear variations, making them more valuable for classification.

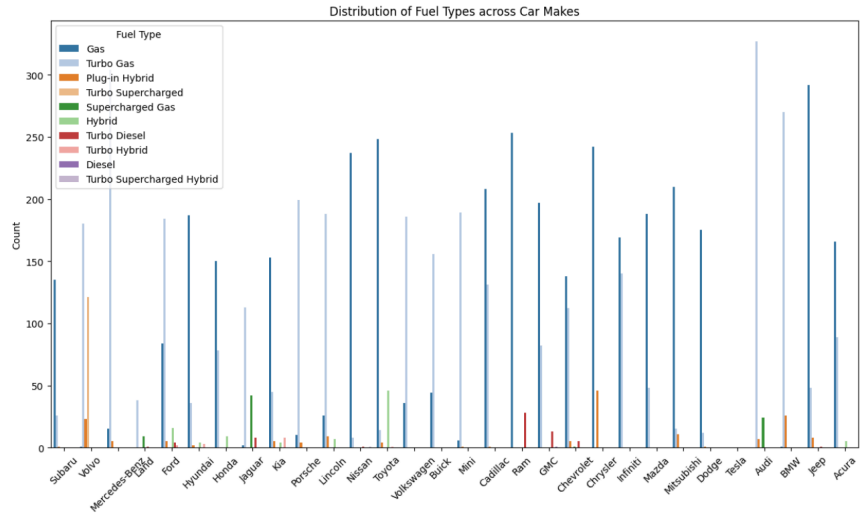
For example, front-wheel drive is predominant in vans but rare in pickup, SUV or sport cars, while rear wheel drive is only common in sport cars, probably due to the performance benefits it offers. Additionally, color preferences show interesting patterns across classes. While black, blue, white and silver are the most popular choices regardless of car class, the exact popularity could differ a lot. These insights could be used to determine the importance of different features when applying algorithms in phase II.

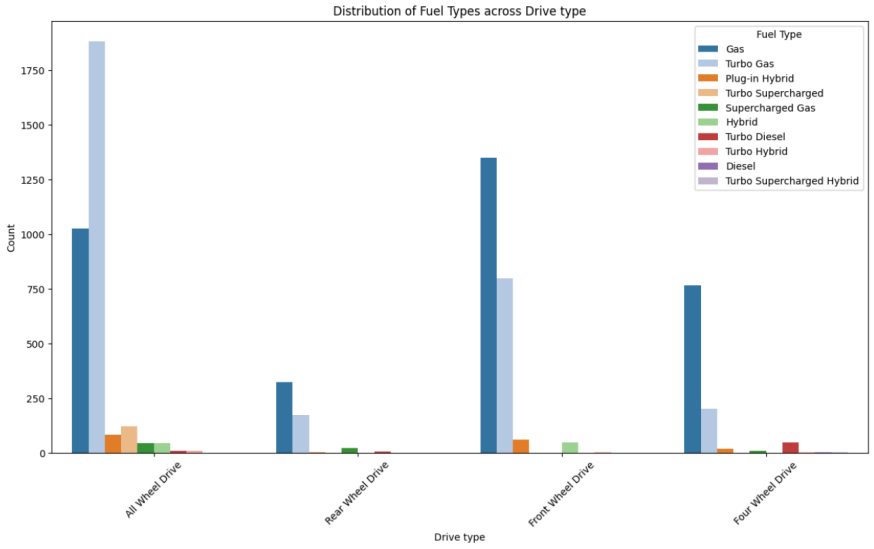
Refined Question 4: What features could be affected by mileage? What attributes are associated with fuel for used cars? (Jiabao Yao)

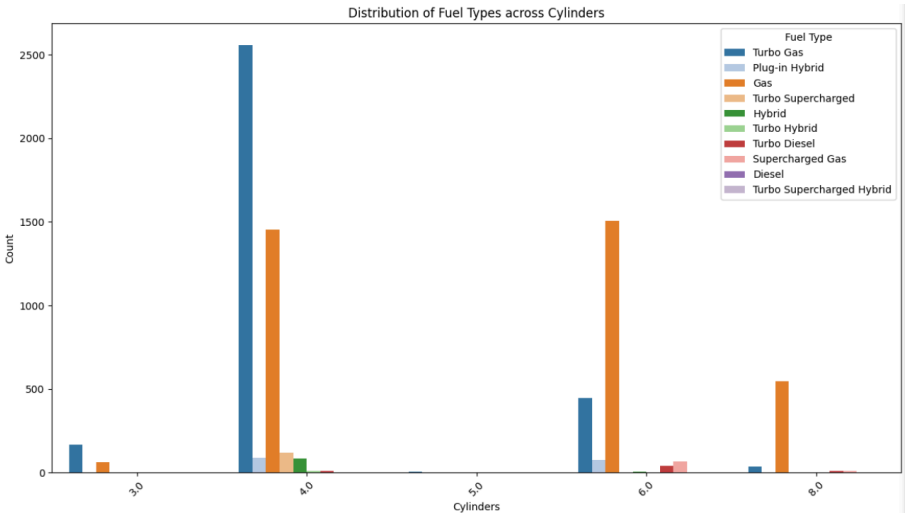
Fuel type is a key factor in determining a car's resale value. In Phase I, our analysis focused on how fuel type impacts resale value across different car brands. To deepen our insights, it's crucial to examine the relationship between fuel type and other vehicle attributes. By understanding these correlations, we can enhance the model's ability to estimate fuel type based on related features, even when direct fuel type data is missing.

Hypothesis 1: The factory configuration of a car functions like a drive type template, suggesting a strong relationship between the car's class and fuel type.

This part is the same as EDA for original question 4. This section supplements the initial EDA for Question 4. Here, I focus on a categorical analysis by selecting 2 non-numeric and 3 numeric parameters.

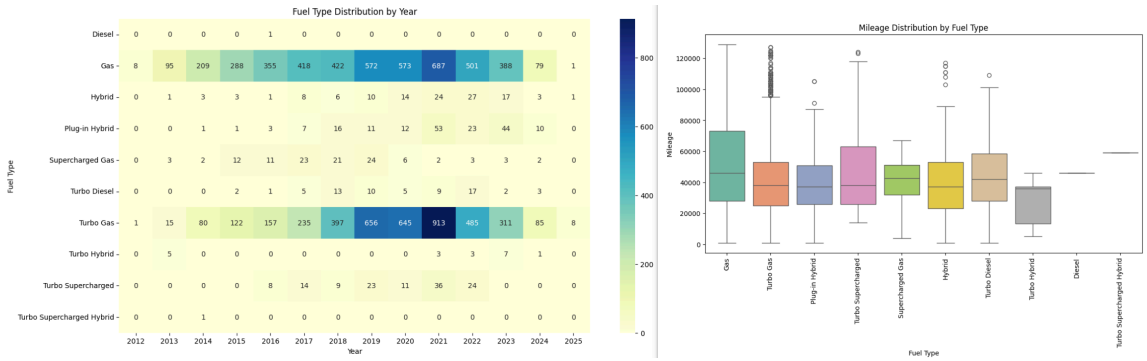






Fuel Type and Make, Drive Type, and Cylinders

Gas and Turbo Gas are the primary fuel types across all brands and drive types. However, the distribution of all four fuel types varies significantly by drive type, and certain brands, like Volvo, show a preference for specific fuel types. While there is no clear trend between fuel type and the number of cylinders, most cars fall within the 3-4 or 5-6 cylinder range.



Fuel, Year, and Mileage

Gas and Turbo Gas have consistently been the dominant fuel types in the used car market over time, indicating minimal correlation between vehicle year and fuel type. Additionally, the median mileage is similar across all fuel types, suggesting that mileage is not significantly related to fuel type.

Hypothesis 2: Human or temporal factors, such as the vehicle's production year, do not significantly influence changes in fuel type.

	Feature	p-value
0	year	NaN
1	make	0.000000e+00
2	model	0.000000e+00
3	price	NaN
4	mileage	NaN
5	Miles per gallon	0.000000e+00
6	Transmission	4.486237e-01
7	owner	NaN
8	VIN	4.893768e-01
9	class	0.000000e+00
10	Auction Brand / Issues	9.999721e-01
11	Accident / Damage	1.501314e-01
12	Open Recall Check	1.293102e-02
13	Odometer Check	7.175124e-01
14	Certified Pre-Owned	5.546073e-02
15	cylinders	NaN
16	Drive type	9.803720e-238
17	Miles per gallon equivalent (MPGe)	1.000000e+00
18	Range (when new)	1.000000e+00
19	Time to fully charge battery (240V)	1.000000e+00
20	Motor	1.000000e+00
21	Bed Length	1.491441e-22
22	exterior_color	1.154439e-01
23	interior_color	5.632623e-39

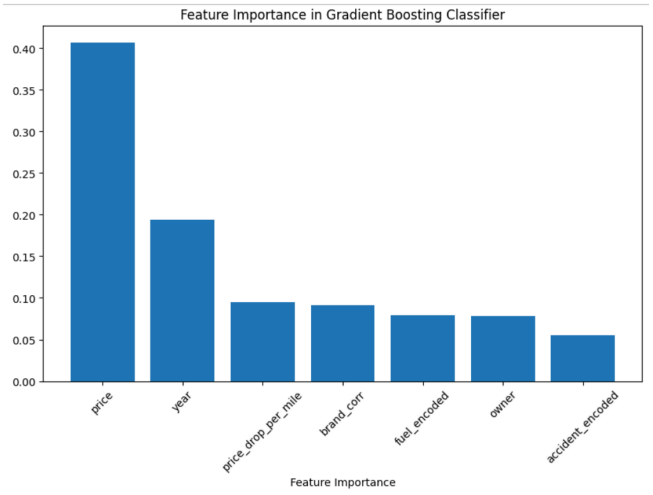
In this section, I calculate the correlation between fuel type and all other features. For categorical features, I use the Chi-Square test, and for numeric features, I apply ANOVA. Features with a p-value less than 0.05 are considered significantly related to fuel type, resulting in a list of significant features. The significant features for fuel type are ['make', 'model', 'Miles per gallon', 'class', 'Open Recall Check', 'Drive type', 'Bed Length', 'interior_color'].

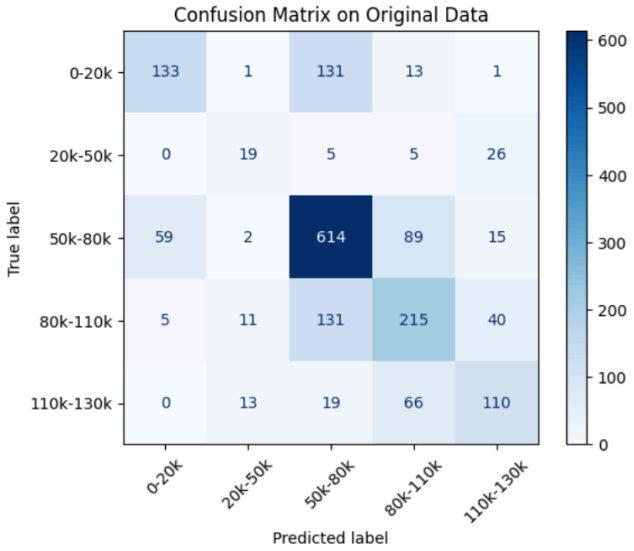
Phase II

Question 1: What features could be affected by mileage? (Te Shi)

1. **Algorithms/Visualizations**

   The Gradient Boosting Classifier was used in conjunction with a customized sampling algorithm to predict mileage ranges [1]. The customized sampling algorithm generated additional synthetic data for underrepresented but potentially important feature values. This approach aimed to give the model a better understanding of the overall data distribution and to mitigate overfitting.



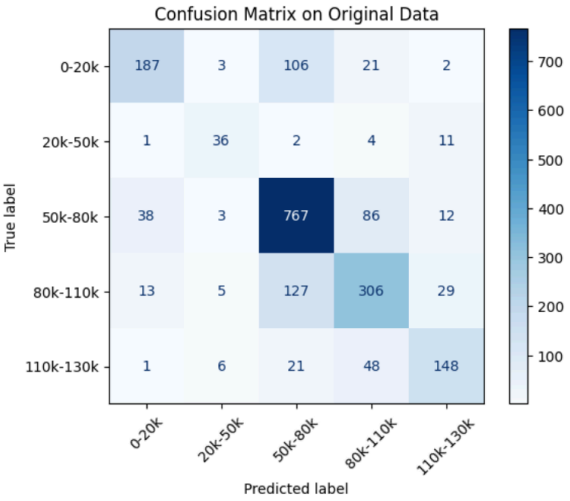



```
Classification Report:
              precision    recall  f1-score   support

           0       0.68      0.48      0.56       279
           1       0.41      0.35      0.38        55
           2       0.68      0.79      0.73       779
           3       0.55      0.53      0.54       402
           4       0.57      0.53      0.55       208

    accuracy                           0.63      1723
   macro avg       0.58      0.53      0.55      1723
weighted avg       0.63      0.63      0.63      1723

1723
Confusion Matrix:
[[133   1 131  13   1]
 [  0  19   5   5  26]
 [ 59   2 614  89  15]
 [  5  11 131 215  40]
 [  0  13  19  66 110]]
{'learning_rate': 0.1, 'max_depth': 6, 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 200}
```




```
Classification Report:
              precision    recall  f1-score   support

           0       0.78      0.59      0.67       319
           1       0.68      0.67      0.67        54
           2       0.75      0.85      0.80       906
           3       0.66      0.64      0.65       480
           4       0.73      0.66      0.69       224

    accuracy                           0.73      1983
   macro avg       0.72      0.68      0.70      1983
weighted avg       0.73      0.73      0.72      1983

1983
Confusion Matrix:
[[187   3 106  21   2]
 [  1  36   2   4  11]
 [ 38   3 767  86  12]
 [ 13   5 127 306  29]
 [  1   6  21  48 148]]
{'learning_rate': 0.1, 'max_depth': 6, 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 200}
```

The final model has a 63% accuracy on original data and 73% with synthetic data. The above graphs will be analyzed in next section

2. Explanation and Analysis

- **Choice of Gradient Boosting Classifier(GBC)**

  Gradient Boosting Classifier (GBC) was selected for this task based on the insights obtained from EDA in phase I. The EDA revealed that certain, including *year*, *price*, *make*, *owner*, *Damage/Accident*, *fuel type* and *owners* have varying degrees of correlation with mileage. Among these, although *year* and *price* showed the strongest correlations, other features only exhibit moderate to weak correlations (0.1 to 0.4).

  Given the complex, non-linear relationships among these multiple features,simpler classification algorithms or models such as decision trees or logistic regression might struggle to capture the underlying pattern effectively. GBC, an ensemble method that combines multiple weak learners to form a robust model, was therefore deemed as a good choice for this task [1].

- **Customized Synthetic Sampling Algorithm**

  To enhance the model's performance and let it better reflect a variety of data records, a customized synthetic sampling algorithm called generate_uncommon was implemented to address underrepresented feature values in the dataset. The reason is that some feature values are very limited in data, however, could also be crucial for the final predictor model. Therefore, instead of just treating them as outliers and dropping, it is better to increase the number of them.

  This algorithm generated additional samples by setting specific underrepresented values while using statistical rules to generate realistic values for other features. The features were divided into two categories: categorical and numerical.

  - For Categorical features such as make, the percentage of each feature value in the original dataset was calculated. The algorithm then generated synthetic rows based on these proportions, for example, if 10% of cars are Honda in the original dataset, then in the newly generated data 10% of car make will also be set to Honda.

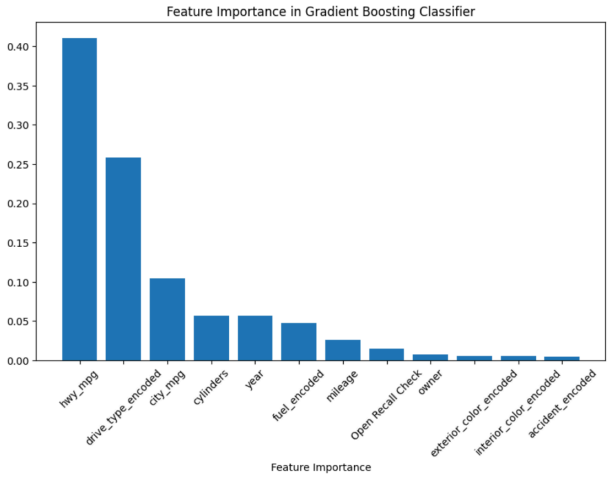
        - For numerical data such as mileage and price, I assumed a normal distribution for each feature, and their mean and standard deviation is calculated. Synthetic values were then generated by sampling from this distribution to approximate real-world patterns.
    - **Feature Encoding and Calculated Data Generation**
      From the EDA, one observation is that unlike other features with weak to moderate correlations to mileage, different car makes exhibit a significant relationship with price and mileage. Additionally, there is also a strong negative correlation between price and mileage. To quantify this effect and capture make-specific trends, a simple linear regression model was used to calculate the *price_drop_per_mile* value for each car make. This metric, which represents the rate of price decrease per mile for different makes, was then added to the dataset to enrich the feature set with a targeted, calculated representation of car make effects.
      For other text-based features, such as *Damage/Accident*, label encoding was applied to convert these categorical values into numerical representations suitable for the model. Additionally, mileage was grouped into five ranges—'0-20k', '20k-50k', '50k-80k', '80k-110k', and '110k-130k'—to create discrete classes for this classification task.
- **Model Tuning**
  The model's performance was further optimized by tuning key hyperparameters including *n_estimators*, *learning rate*, *max_depth*, *min_samples_split* and *min_samples_leaf*. GridSearchCV is used to systematically test various combinations of these parameters to identify optimal configuration. This tool can help create the best possible model within the parameter search space in a relatively efficient way.
- **Results Analaysis**
  The above feature importance graph reaffirmed the finding in the EDA stage. The ranking of importance closely matches the correlation values of each feature. A noticeable difference is that *year* in EDA shows a higher correlation with mileage than price, however the GBC considers price as the most influential feature, this could mean that year might not be that representative on its own. The age of a car could be more like an indirect measurement about a car's condition. It shows that mileage has a strong relationship with price, so confirmed its importance when predicting the resale value.
  From the confusion matrix and classification report, the 63% accuracy reflects a moderate performance of this model in capturing the complex nature of this dataset. However, the overall recall value and F1 score in the macro average indicates the performance of this model varies across different categories.
  Specifically, the low precision on class 0 (0k-20k) and class 1 (20k-50k) indicates it could not capture the characteristics of these two categories correctly. This is also reflected in the confusion matrix, where this model often confused class 1 and class 2 with other classes.
  The best performance is from class 2 (50K-80K) which has best precision and recall value. And from the confusion matrix it is clear that there is more data in this range, could be an explanation why the dataset with synthesized data has a better performance using the same model.
  Contrary to the original dataset, the performance of this model is better for all the 5 categories with synthesized data included. This indicated that more data could be helpful in

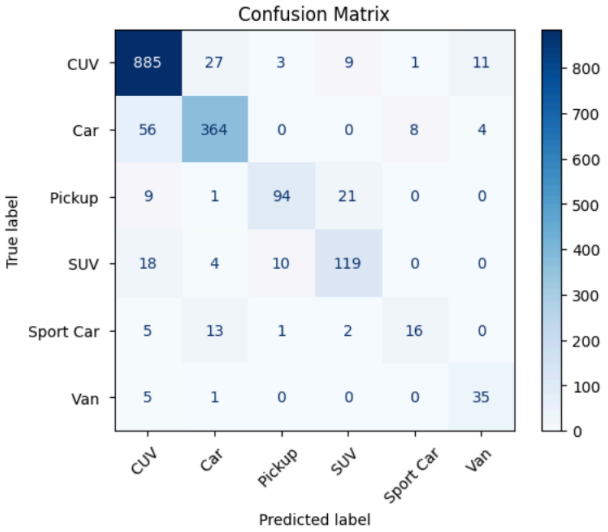
terms of performance improvement in general, specifically the sampling generation algorithm could be reinforcing important patterns without confusing models. However, since the new data is generated based on distribution of the original data, so the improvement of accuracy could also be from overfitting and capturing more information, more investigation should be done in the next step to confirm the validity of this algorithm.

- **Reference**
    1. **Gradient Boosting Classifier:**
       A. Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2017. [Online]. Available: http://cds.cern.ch/record/2699693
    2. **GridSearch CV**:
       https://scikit-learn.org/dev/modules/generated/sklearn.model_selection.GridSearchCV.html

Question 2: Do different car classes exhibit significant and distinct characteristics across features? (Te Shi)

1. **Algorithms/Visualizations**
   Decision Tree and KNN were used to classify car classes given a bunch of features. The performance of these two classification was compared to select an optimal solution for this problem.

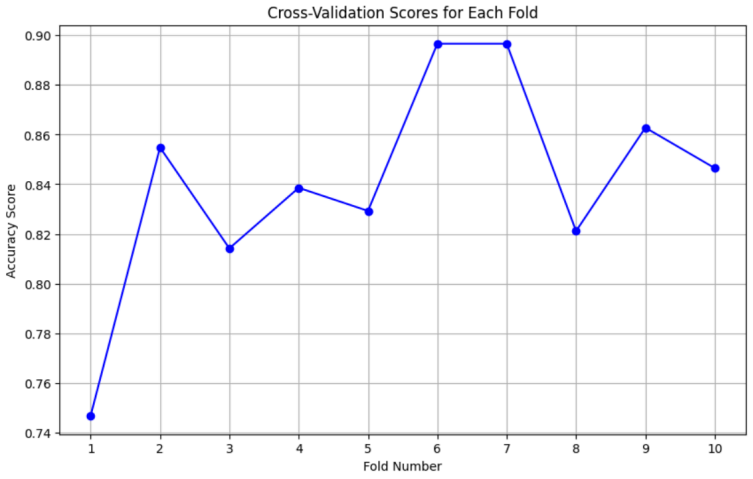


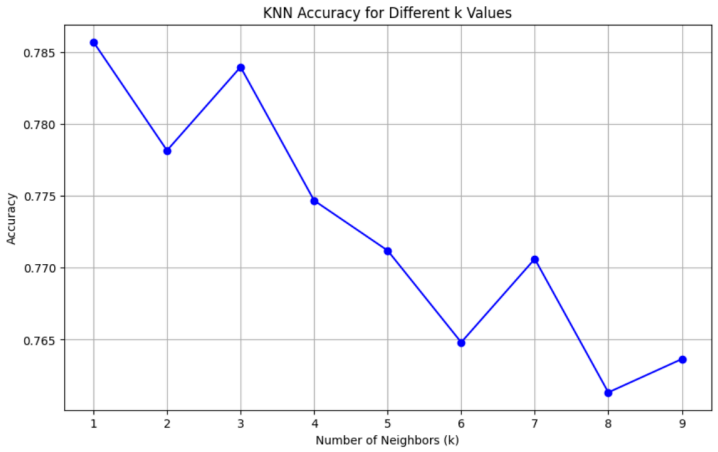



```
Fitting 5 folds for each of 162 candidates, totalling 810 fits
optimal hyperparameters: {'criterion': 'entropy', 'max_depth': 10, 'max_features': None, 'min_samples_leaf': 1, 'min_samples_split': 5}
Accuracy: 0.8786295005807201
Classification Report:
              precision    recall  f1-score   support

         CUV       0.90      0.95      0.92       936
         Car       0.89      0.84      0.86       432
      Pickup       0.87      0.75      0.81       125
         SUV       0.79      0.79      0.79       151
   Sport Car       0.64      0.43      0.52        37
         Van       0.70      0.85      0.77        41

    accuracy                           0.88      1722
   macro avg       0.80      0.77      0.78      1722
weighted avg       0.88      0.88      0.88      1722
```





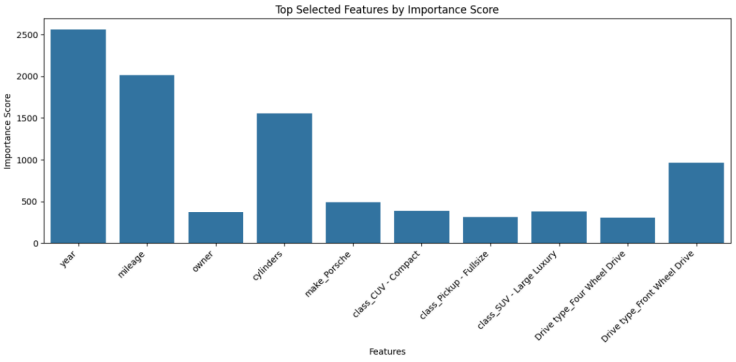
2. Explanation and Analysis

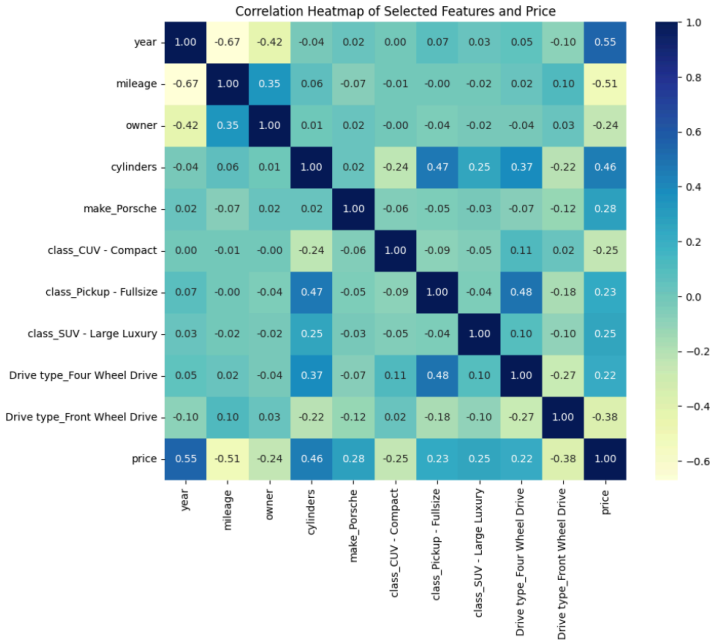
- **Use of Decision Tree and KNN**
  The EDA of question 2 revealed distinct characteristics across different car classes, suggesting that relatively simpler classification models could perform effectively. Therefore, decision trees and KNN are used to solve this question. The performance of these two will be compared.
- **Model Tuning**
  The decision tree model underwent an extensive hyperparameter tuning using GridSearchCV. The parameters tested included *criterion*, *max_depth*, *min_samples_split*, *min_samples_leaf*, and *max_features*. The best-performing model achieved an accuracy of 87.8%, using "entropy" as the criterion to measure data splitting. This model had a relatively high depth, which is well-suited to handle the complexity of the feature-rich dataset.
- **Result Analysis**
  Both classifiers yield a solid result but the decision tree is relatively better then KNN. For KNN, the accuracy of classification is dropping with the increase of k, showing that some classes may overlap with the increase of number which lead to more misclassification. The performance disparity can be also explained as KNN tends to perform badly for high dimensional data.
  In the decision tree, the feature importance maps clearly showed that *hwy_mpg* and *drive_type* are the 2 most important features to determine car class and the obvious difference among different car classes can also be proved by the EDA graph for these two features.
  The accuracy is used as a measurement to determine the best model among trees with different combinations and KNN. The confusion matrix, precision, recall, f1-score and cross-validation provide more information to further confirm the performance.
  Specifically, this decision tree model performed well in all of these criterions. Except high accuracy, high precision and recall value indicates that this model not only effectively identifies car classes but also minimizes misclassifications. In the end the cross validation further reinforced the model's robustness.
  Overall, the Decision Tree model provides a reliable and efficient approach to predicting car classes. Its simplicity and strong performance make it a valuable tool for cases where users may have limited or incomplete information on car attributes.

Question 3: Does color matter for used car prices and how does it affect them? (Jiabao Yao)

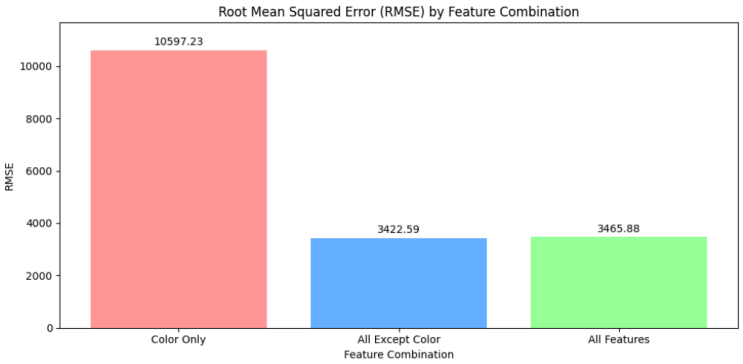
1. **Algorithms/Visualizations**
   The **SelectKBest** method was applied to identify the most important features for the price prediction model. SelectKBest uses statistical tests like the Chi-Square test, ANOVA F-test, or mutual information score to score and rank features based on their relationship with the target price. If color is among the selected features, it suggests

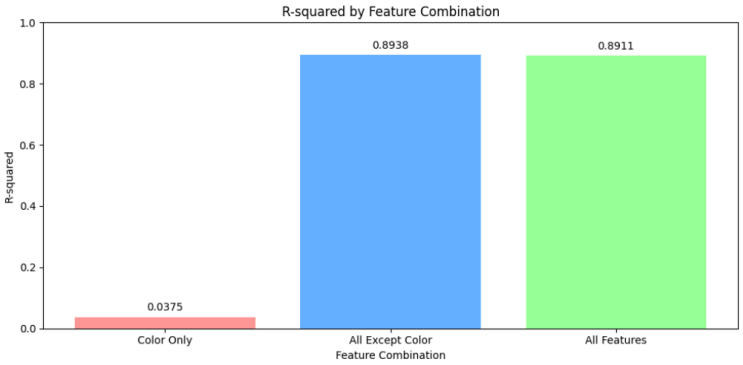
that color (exterior or interior) plays a key role in predicting price; otherwise, color is not a primary factor.

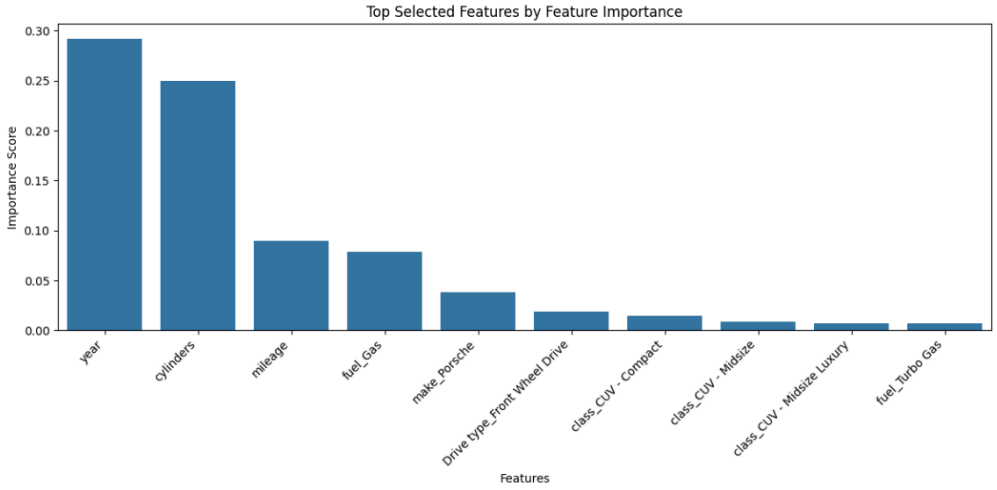




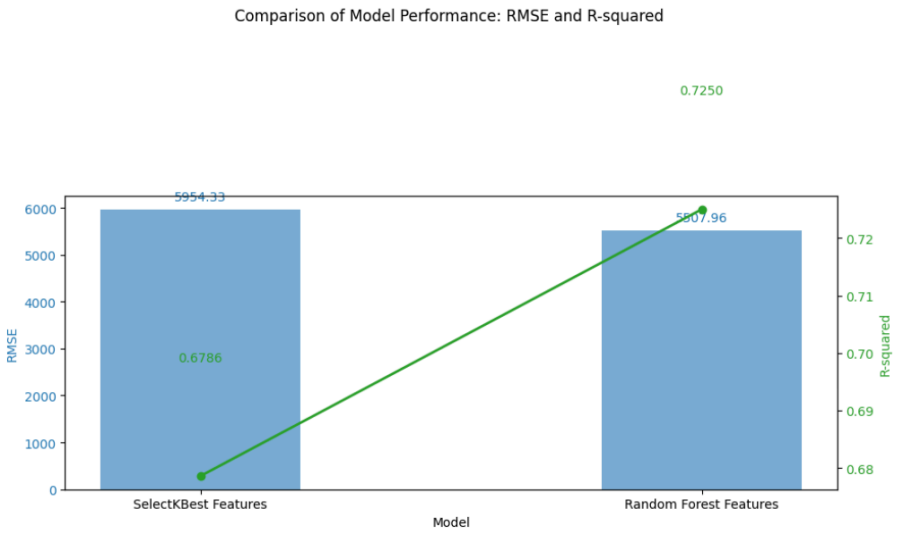
Random Forest Classification was used to make predictions based on various feature combinations, including only color, all features, or other subsets, to predict the price of used cars. Since Random Forest is built on decision trees, adjusting the number of trees can improve prediction accuracy. The algorithm also calculates feature importance, providing insights into the top features influencing price prediction.







Linear Regression was used to determine which algorithm finds a more accurate feature set.



2. **Explanation and Analysis**
   - **Use of** Statistic algorithm **SelectKBest** and **Random Forest,** and **Linear regression.**
     SelectKBest finds k most important features by the scores of features (Process 1) while Random Forest finds k most important features by the importance of features (Process 5). However, the result of these two algorithms are different. For further understanding and confirming the real important features, I use Linear Regression to figure out which algorithms perform better for selecting key important features (Process 7). The accuracy of the model is measured by r2 and RMSE.
   - **Model Tuning**
     The primary parameter to adjust in a Random Forest Classifier is the number of decision trees (n_estimators). Increasing the number of trees generally improves accuracy, but the model's accuracy will stabilize beyond a certain point. The random_state parameter controls the randomness of bootstrapping the samples for tree-building and the sampling of features when finding the best split at each node. By using a fixed random_state across all training runs, I can consistently compare how different inputs (color only, all except color, all features) affect model accuracy. Each input may have an optimal number of trees (n_estimators) that maximizes accuracy. Given a specific range, GridSearchCV helps find out the best n_estimators by evaluating the best score.

     Additionally, given that SelectKBest and Random Forest yield different lists of important features, Linear Regression was used to identify the best set of k features for price prediction.
   - **Result Analysis**
     Linear regression comparisons show that Random Forest provides the most accurate selection of the top k features (Process 7). When choosing the top 20 important features, neither exterior or interior color appears on the list,

indicating that color is not a significant predictor of price. Moreover, when training data includes only color, model accuracy is very low (Process 2). However, interior colors like Red and Brown have slightly higher importance, aligning with the median price distribution for these colors.

For training data that includes all features or all except color, the model performs well and even slightly better without color, suggesting that color can be disregarded in a price prediction model (Process 3-4 &6).

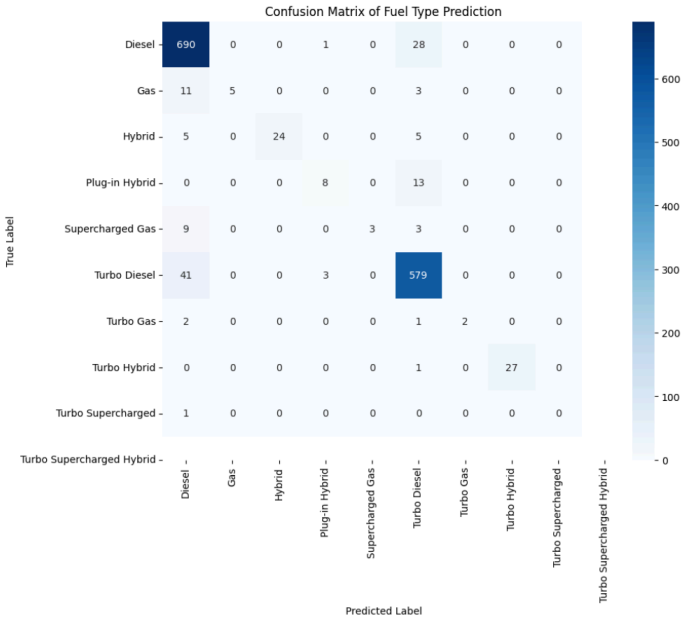
Random Forest is a strong candidate for the price prediction model, achieving a low root mean squared error of 3427.9 and an R-squared of 0.893.

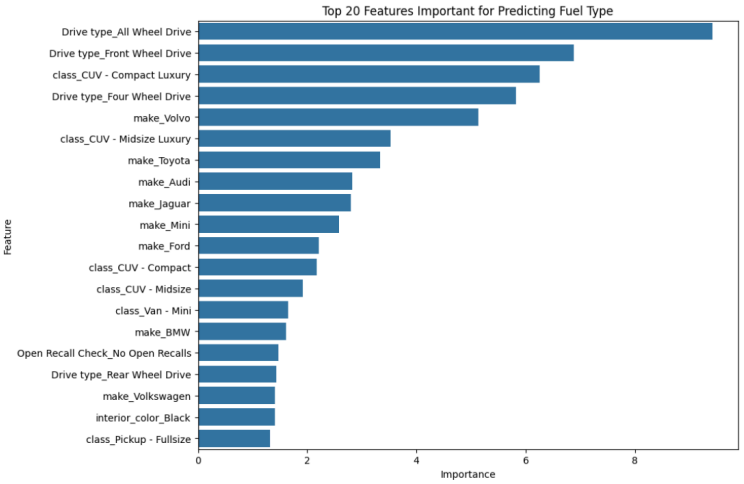
Question 4: What attributes are associated with fuel for used cars? (Jiabao Yao)

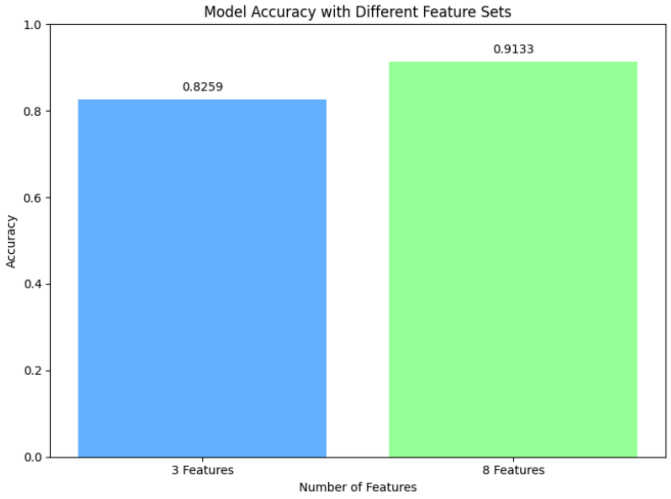
1. **Algorithms/Visualizations**

GridSearchCV was utilized to fine-tune hyperparameters, optimizing 'iterations,' 'learning_rate,' 'depth,' 'random_seed,' and 'verbose' for training the CatBoostClassifier.

The **CatBoostClassifier** was used to analyze the relationship between fuel type and other features in the dataset, enabling both fuel type prediction and insights into feature importance.







2. Explanation and Analysis

- **Use of GridSearchCV and CatBoostClassifier**

  GridSearchCV automates the process of tuning the CatBoostClassifier by exploring various parameter combinations within specified ranges. It identifies the optimal parameters for the model to maximize accuracy.

  The **CatBoostClassifier** not only predicts the target fuel type but also provides insights into feature importance, highlighting the influence of each feature on fuel type predictions.

- **Model Tuning**

  Five key parameters are essential for training an effective CatBoostClassifier: iterations, depth, learning_rate, random_seed, and verbose. CatBoostClassifier, being a tree-based model, relies on depth to control the complexity of each tree. Typically, random_seed and verbose are set to fixed values—random_seed ensures reproducibility, while verbose controls logging frequency during training.

The main parameters to tune for optimal model accuracy and efficiency are iterations, depth, and learning_rate. These directly affect model performance and computation time and can be optimized using GridSearchCV over a specified range. Model tuning throughout all model training.

- **Result Analysis**

  From the analysis in Question 1, we know that fuel type is a crucial feature for predicting price. Therefore, accurately predicting fuel type is essential when it is missing.

  Using eight features with low p-values identified in the EDA (make, model, Miles per gallon, class, Open Recall Check, Drive type, Bed Length, interior_color), the model based on CatBoostClassifier achieved an accuracy of 0.913, demonstrating that CatBoostClassifier is effective in predicting fuel type based on highly related features (Process 1).

  Further, by examining feature importance from CatBoostClassifier, we can narrow down the list of relevant features (Process 2). The top 3 features—'Drive type', 'class', and 'make'—play a significant role in predicting fuel type. Training the model with only these three features yields an accuracy of 0.826, indicating their strong predictive power (Process 4).

  From the error distribution (Process 3), the model performs well with larger classes, such as Diesel and Turbo Diesel, suggesting it might rely heavily on the frequency of instances in the training data. Smaller classes, especially Gas and Supercharged Gas, have poorer performance, indicating the model might need more instances or better feature differentiation for these types. There's confusion between similar fuel types (Diesel and Turbo Diesel), suggesting that additional features or refined feature engineering might help improve separation between these classes.

Question 5 and 6 are refined slightly based on phase 1 EDA to understand the problem statement deeper. The experiments are conducted on one CCR private-owned node with an A100 to train the LSTM network. (Run brand_price_q5.py and smote_owner_price_q6.py for each question)

Q5(refined): Does the resale price of a particular brand get influenced by the resale prices of competing brands, and if so, is there a lag effect in this influence? (Chao Wu)

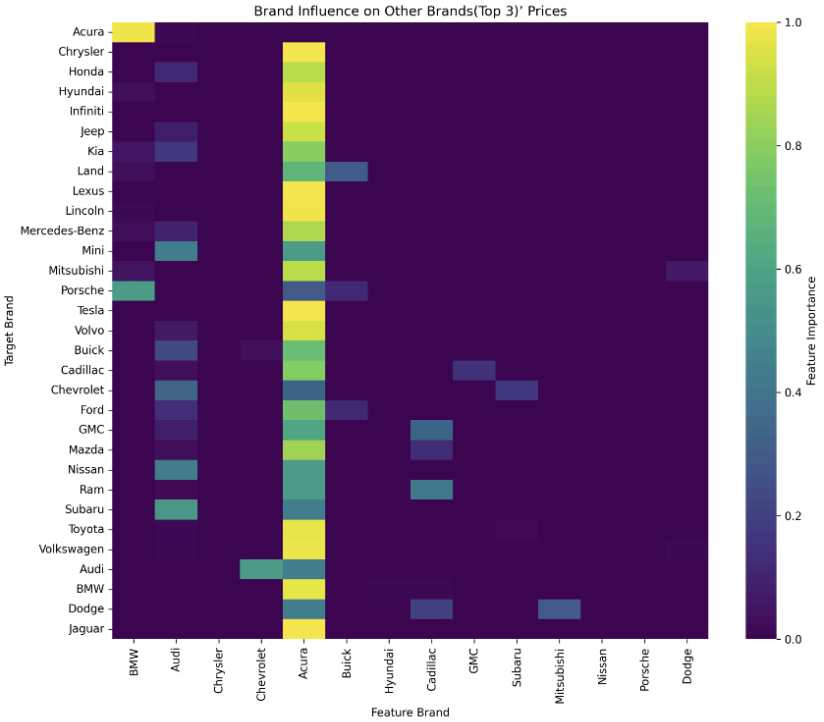
1.Algorithms/Visualizations

The experiment employs an XGBoost model and an LSTM neural network to understand whether a brand's resale price is influenced by the prices of other brands.

Data Preparation and Feature Selection:

The average resale price for each brand by year is calculated. An XGBoost model is trained to predict each target brand's resale price based on the resale prices of all other brands. Using

the trained XGBoost model, feature importance scores are extracted to identify the top 3 brands that have the most significant influence on the target brand's resale price. The results are stored in top_features_dict, which lists the top 3 influencing brands for each target brand.

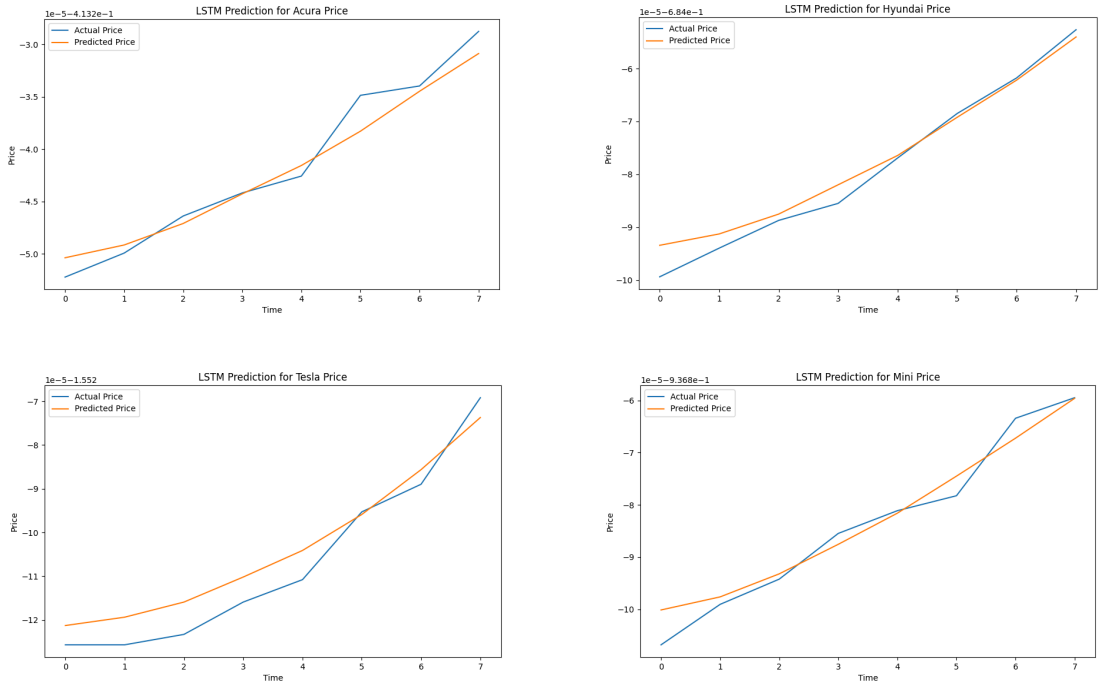


LSTM Model for Time Series Prediction:

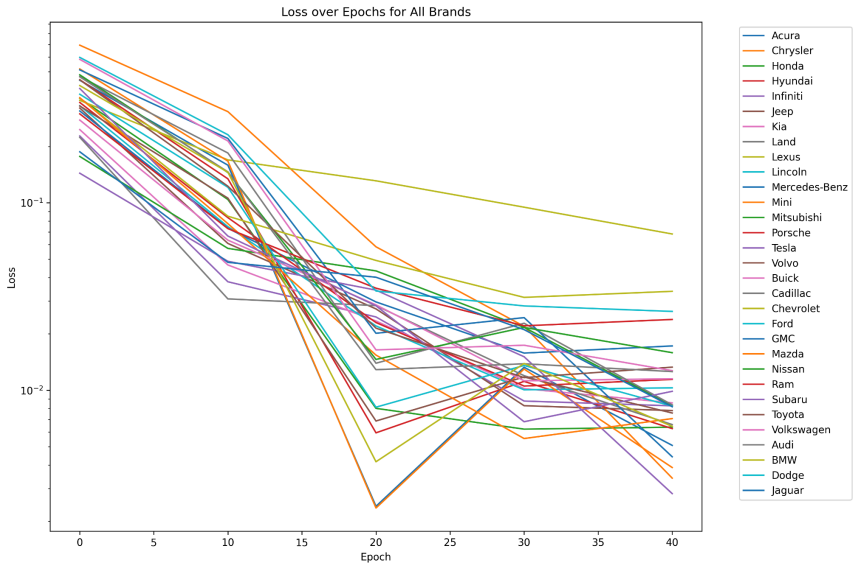
The LSTM model is trained on a time series data structure created using the top 3 influencing brands for each target brand. Each LSTM model attempts to predict the future prices of a target brand based on the historical prices of its top influencing brands. We do not use BiLSTM because the data cannot be backward (it's time series data).

Sample brands LSTM prediction over years(log scale with x-axis as time step-not years):

(See pdf for all brands price predictions)



A MinMaxScaler is used to scale the data, ensuring normalized values for the LSTM.
The loss history is recorded for each brand, which shows the progress of the model's learning ability over epochs.



2. Explanation and Analysis

Using XGBoost: XGBoost is an algorithm especially good at structured data and implements machine learning algorithms under the Gradient Boosting framework. Here we use this powerful tool to extract the feature importance of competing brands to see which competing brands are likely to influence the target brand's resale price. By fitting the XGBRegressor

using other brands' average price data to model the target brand price, we consider the top 3 brands with the highest importance as most relevant to the target brand's resale price.

Using LSTM: Introducing over the years average price is coarse and the question needs to be looked up deeply considering the year-wise lag effect. Once we identified the top 3 influencing brands over the years, LSTM was selected for time series prediction. LSTM is suitable for sequential data and can get temporal dependencies, making it ideal for analyzing price trends to predict future prices. Given that resale prices likely depend on past prices of both the target brand and competing brands, LSTM's memory capabilities make it a rational choice. Here we still use the above top 3 important brands to predict the target brand price.
We constructed a data sequence in a sliding window so that the LSTM model can learn from data from past years. For each year of data, the top_brands data from the previous years is used as input feature X. The target_brand price of the current year is used as the target value y.

Results Analysis:

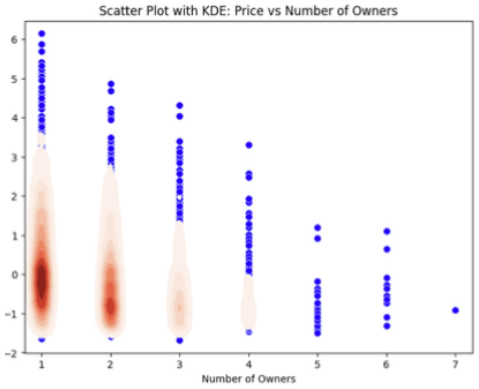
As shown in the heatmap except Acura which nearly has the most significant impact to all other brands, the feature brands and the target brands show rational correlations in most scenarios, such as Porsche is most influenced by the resale price of BMW, Mercedes-Benz is most influenced by Acura and Audi. For the Acura case, it could be the reason that we have many samples of Acura far more than other brands which have a significant impact on other brands' resale price.
As shown in the by brand prediction, LSTM captures the basic trends of resale price change over years, making it a robust tool to predict future resale price. The loss of LSTM significantly dropped after 50 epochs during the training process. To evaluate the efficiency of the algorithm, we use the trained model to predict the resale price(rescaled). We got the R^2 value around 0.97 which indicates the model efficiency. The MinMaxScaler() here is used to let the model converge but we still need to rescale it. After the rescale process, we get the by brand MAE and RMSE (in output file brand_metrics_results.csv). We can see the MAE and RMSE are relatively small compared to merely applying one single model. In this case, using top 3 important brands to forecast target brand price is feasible.

Brand	MAE	RMSE
Acura	1620.74	1948.41
Chrysler	1445.5	1691.58
Honda	1050.17	1333.61
Hyundai	828.48	1148.24
Infiniti	2902.72	3528.92
Jeep	1086.83	1438.76
Kia	1276.45	1461.18
Land	3037.3	3621.89
Lexus	3766.65	4877.56
Lincoln	2599.85	3390.31
Mercedes-Be	5798.35	7143.67
Mini	790.79	874.84
Mitsubishi	785.82	1527.29
Porsche	2164.37	2757.57
Tesla	800.78	883.17
Volvo	1351.58	1846.9
Buick	934.99	1107.06
Cadillac	2523.17	2954.46
Chevrolet	3062.27	3557.63
Ford	1664.83	1774.86
GMC	1320.7	1695.63
Mazda	1178.65	1516.49
Nissan	1505.1	1875.81
Ram	3771.87	5133.87
Subaru	1472.91	1665.37
Toyota	1612.24	2025.71
Volkswagen	1092.18	1305.8
Audi	1751.14	2042.7
BMW	2777.05	3076.12
Dodge	3156.03	3856.4
Jaguar	1649.64	2167.21

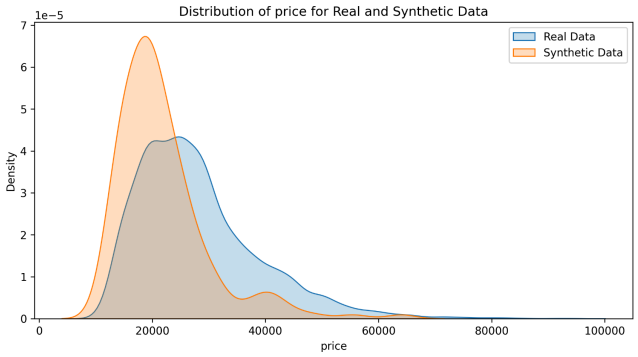
Q6(refined): Is the number of owners a significant factor influencing resale price? Given the current imbalance in the distribution of owner counts, could data augmentation help create a more balanced distribution and enhance the predictive power of this feature? (Chao Wu)

1. Algorithms/Visualization:

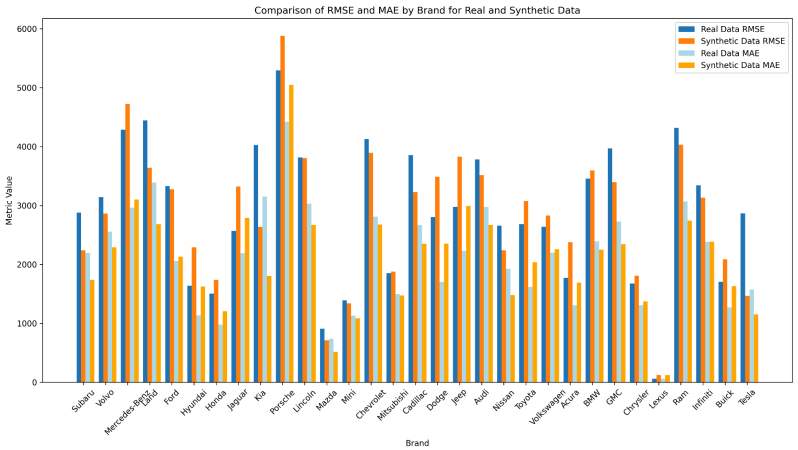
In previous EDA we found #owners is not a very significant impactor for predicting price. And we also see the distribution of #owners are imbalanced with many owner counts = 1,2,3 and less in 4,5.

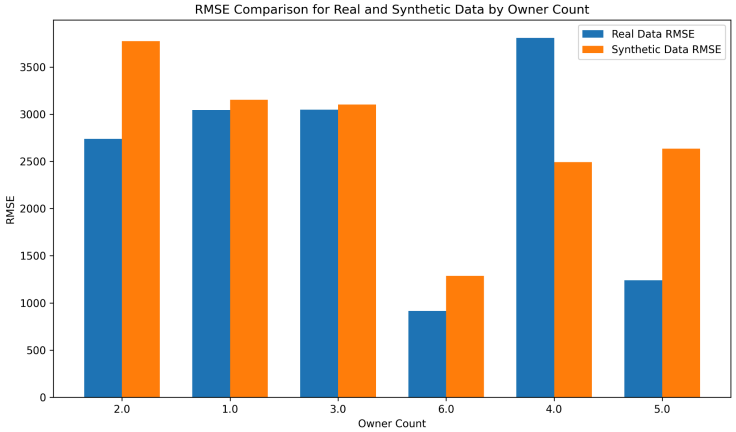


We want to know if data augmentation helps create a more balanced distribution and enhance the predictive power of this feature. We use a systematic approach involving SMOTE (Synthetic Minority Over-sampling Technique), ensemble stacking, and regression models. We first applied SMOTENC, a variant of SMOTE tailored for categorical features, to generate synthetic samples for the less common owner categories (e.g., cars with 4 or 5 previous owners). The SMOTENC algorithm is applied to the categorical features (make, model, year, and owner) and the continuous features (price, mileage), but only for #owner = 4 or 5. We didn't generate owner counts equal to 6 or 7 because it's rare in reality and scarce in the original dataset. The synthetic data is then combined with the real data to form a balanced dataset.



Then a stacked regression model combining predictions from Random Forest and Linear Regression models, followed by a meta-model (a simple neural network) to make the final prediction. This stacked model architecture aims to improve prediction accuracy by leveraging the strengths of each base model. The stacked model is trained separately on the combined (real + synthetic) data and on only the real data. The evaluation metrics, specifically RMSE and MAE, are calculated for both datasets to assess whether the augmented data enhances predictive power.





2. Explanation and Analysis

Using SMOTE: Without data augmentation, we observed an imbalance, especially with a high concentration of cars having 1–3 previous owners and fewer cars in the 4–5 owner range. SMOTE was applied specifically to create synthetic data for these less common owner categories (4 and 5). This approach aimed to balance the distribution, potentially improving the model's sensitivity to the "owner" feature in predicting prices by addressing data scarcity for higher owner counts. The overlap between the real and synthetic data distributions indicates that SMOTE data aligns reasonably well with the real data's distribution. However, the shift also implies that SMOTE might introduce a mild bias towards lower prices. Both real data and SMOTE data have long-tail effects.

Using Stacking:

Using Stacking: In the stacked model, we combined Random Forest and Linear Regression models with a neural network as a meta-model. This stack architecture leverages the strengths of each base model to better get complex relationships between the features(owner counts which are not significant using a single model) and resale price. The results show the differences in RMSE and MAE across real and synthetic datasets for each brand and for different owner groups, indicating how augmentation of scarce owner count groups can help better predict the price.

Results Analysis:

We got SMOTE data overall RMSE: 6223.46, MAE: 4184.90 and real data overall RMSE: 6632.22, MAE: 4488.35. We only use 'year', 'mileage', 'owner', 'make' to predict the price and we see a drop in both MSE and RMSE after incorporating synthetics data. The bar chart compares RMSE and MAE for real and synthetic data across various brands. Brands exhibiting lower errors for synthetic data include Subaru, Ford, Chevrolet, Porsche, Ram, Tesla. For these brands, after comparing their number both in the original dataset and the SMOTE dataset, we found that in the synthetics dataset their number is relatively small. Then we took a deep look into these brands and find the synthetic data which #owners = 4 is bigger than #owners = 5, which aligns with our assumption that the synthetic data may distort the small sample groups.

MSE values by owner count reveal the effect of data augmentation on prediction accuracy for different owner counts. The synthetic data tends to reduce RMSE for higher owner counts (like 4), indicating that balancing these classes improved the model's performance in predicting prices for cars with more previous owners. However, for lower owner counts (1–2), synthetic data has slightly higher RMSE, possibly due to slight distortions introduced by SMOTE. For 5 and 6, the possible explanation could be that scarcity in the original data distorted the data augmentation process. In the next steps, we aim to provide more data augmentation process and figure out potential ways to define how well the synthetic data is constructed.

Reference:

[1]XGBoost: https://xgboost.readthedocs.io/en/latest/
[2]LSTM: https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html
[3]SMOTE:https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html
[4]Hochreiter S. Long Short-term Memory[J]. Neural Computation MIT-Press, 1997.
[5]Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of artificial intelligence research, 2002, 16: 321-357.
[6]Breiman, L. (1996). Stacked regressions. Machine learning, 24, 49-64.
[7]Chen, Tianqi, and Carlos Guestrin. Xgboost: A scalable tree boosting system. Proceedings of the 22nd international conference on knowledge discovery and data mining. 2016.

Question 7 (Phase 2): How do the accidents or damage records of the used cars affect the resale price? (Shijie Zhou)

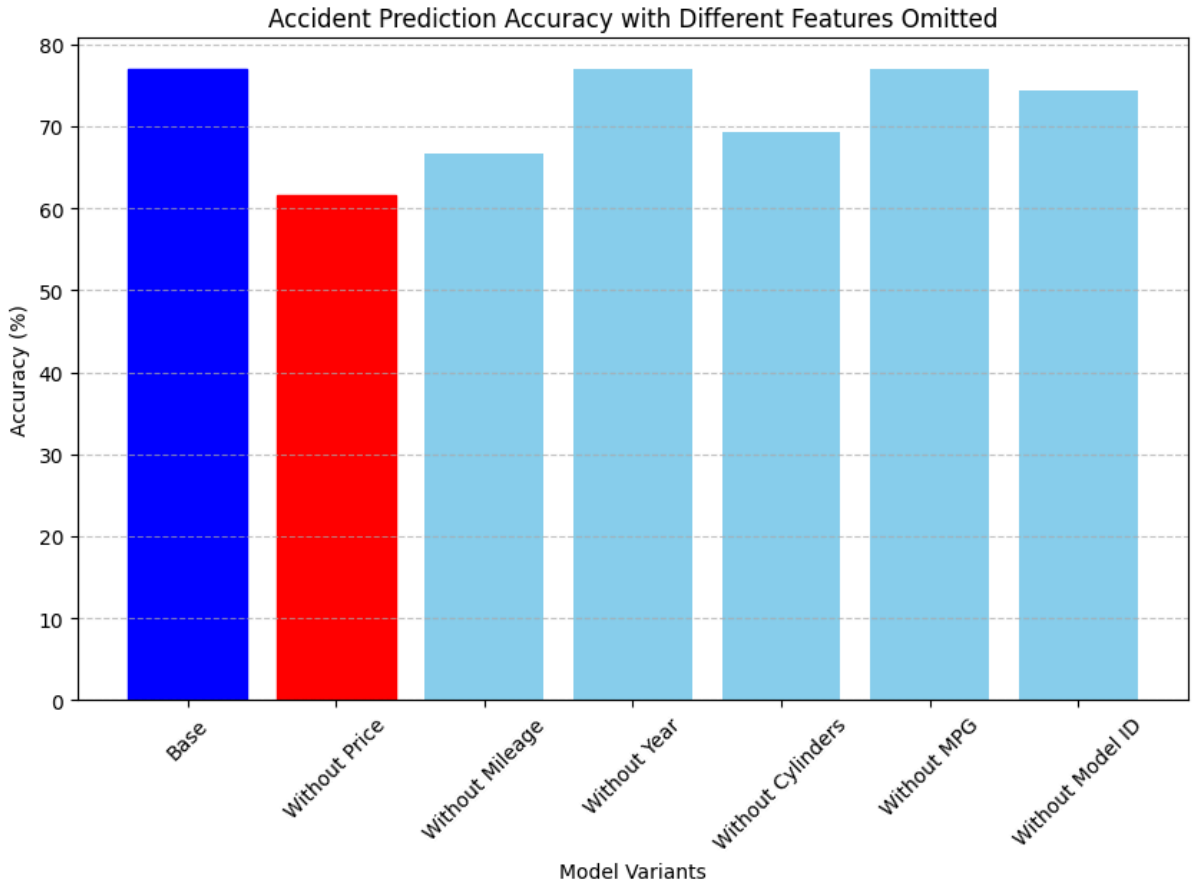
In Phase 2, we show that the used cars' price and accidents have the strongest correlation by conduction classification on predicting whether the car has at least 1 accident. We build a neural network with hidden dimension 100 and conduct experiments on Mercedes Benz's car price data.

Data processing: We convert 'Miles per gallon' string data into the average data between 'city' and 'hwy' and convert 'model' string data into the 'model_id' number value to model this attribute as different models of a car brand have different beginning prices.

Model variant: We build **neural network classifiers[1][2]** with hidden dimension 100 with following different inputs:

(1) Base experiment with all attribute 'mileage', 'year', 'cylinders', 'price', 'Average MPG' and 'model_id' as the input attribute and make the prediction on accident classification (no accident: 0, at least 1 accident: 1)
(2) Ablation experiment without 'price' attribute feeded in.
(3) Ablation experiment without 'mileage' attribute feeded in.
(4) Ablation experiment without 'year' attribute feeded in.
(5) Ablation experiment without 'cylinders' attribute feeded in.
(6) Ablation experiment without 'Average MPG' attribute feeded in.
(7) Ablation experiment without 'model_id' attribute feeded in.

The final ablation results are shown as:



We can observe that the missing 'price' feature decreases the accuracy the most and thus used cars' selling price and accidents have a higher bounded relationship than other features.

Reference

[1] Schmidhuber, Jürgen. "Deep learning in neural networks: An overview." Neural networks 61 (2015): 85-117.

[2] https://scikit-learn.org/1.5/modules/neural_networks_supervised.html

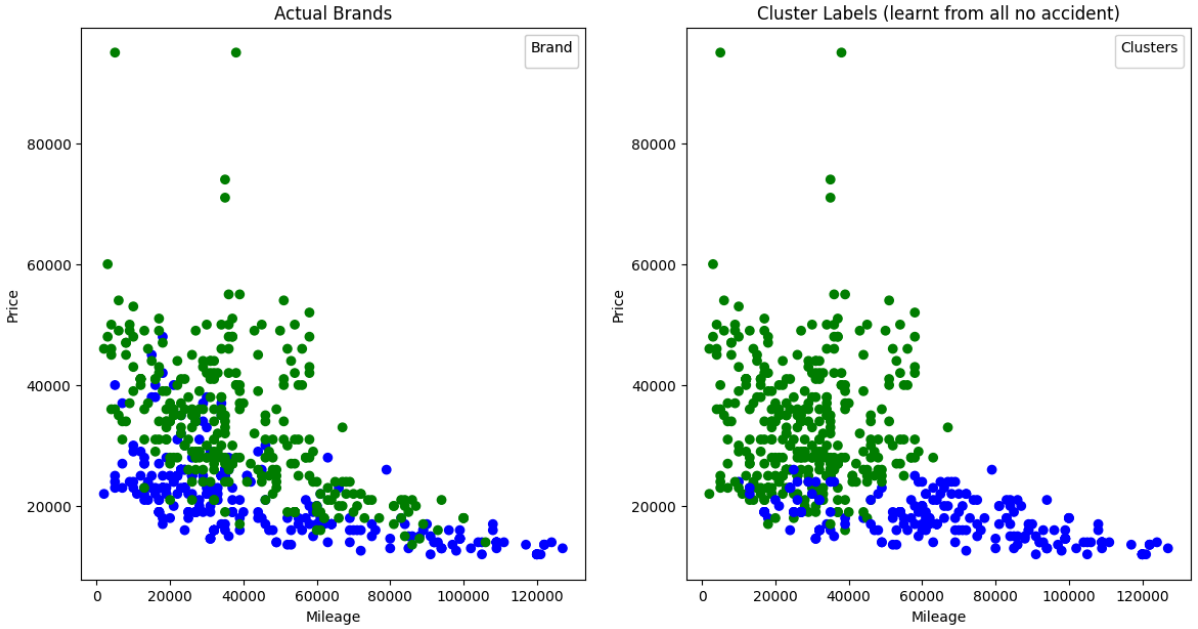
Question 8 (Phase 2): For used cars with different makes, will the accident record affect the used cars' price differently? (Shijie Zhou)

To investigate the impact difference of accident records to the used cars' price between different makes, we select two makes Hyundai and Cadillac as the representative makes for cheap car make and expensive cars make.

We conduct the unsupervised clustering using the **Gaussian Mixture Model[1][2]** only evaluating two brands' no accident cars, but with different learnt models.

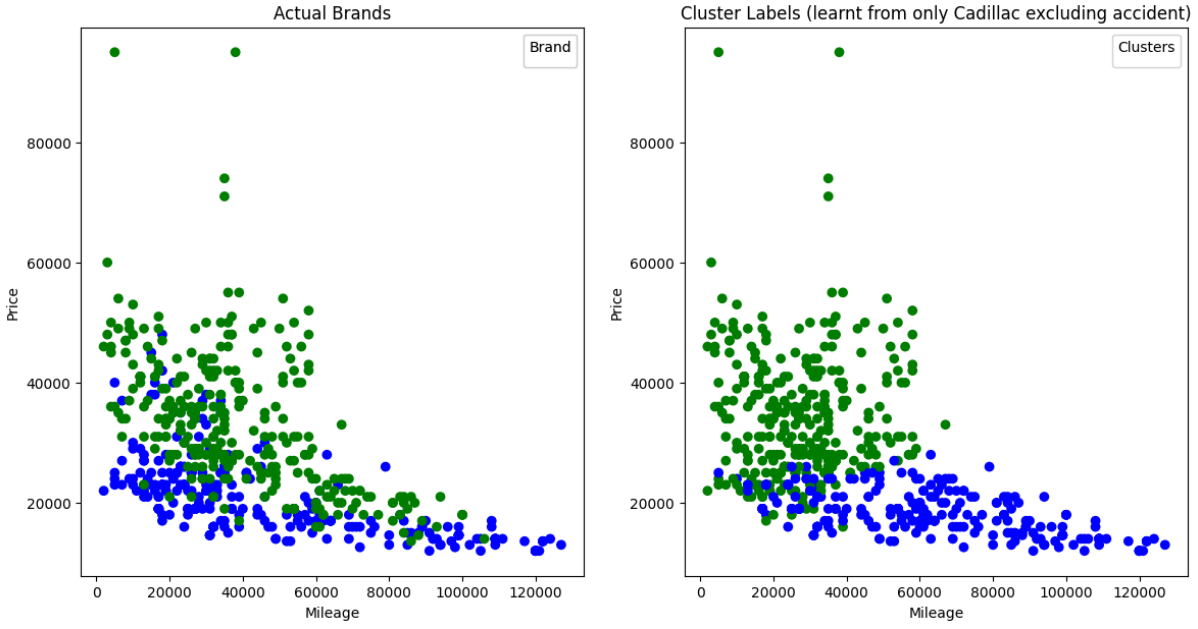
(1) First, we will learn the `Gaussian Mixture Model` on combined `Hyundai and Cadillac no accident cars`, with attributes `mileage', 'price', 'year'`.

The clustering accuracy is 0.6262975778546713 with following visualization:

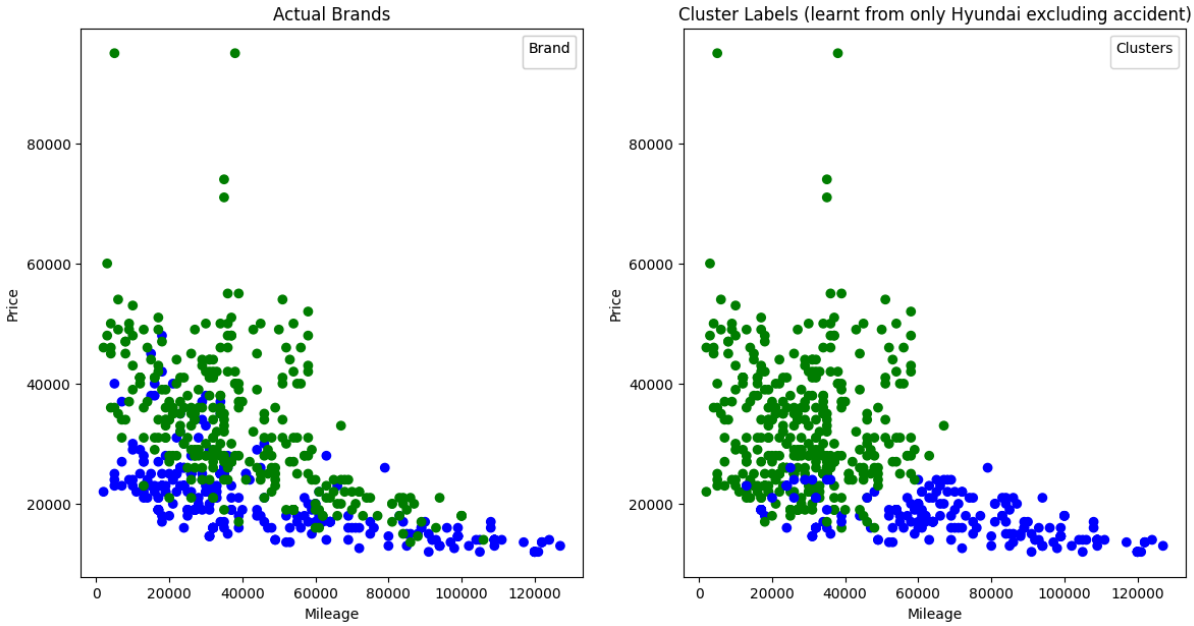


(2) Second, we will learn the `Gaussian Mixture Model` on combined `Hyundai all cars and Cadillac no accident cars`, with attributes `'mileage', 'price', 'year'`.

The clustering accuracy is 0.6505190311418685 with following visualization:



(3) Third, we will learn the `Gaussian Mixture Model` on combined `Hyundai no accident cars and Cadillac all cars`, with attributes `mileage', 'price', 'year'`.
The clustering accuracy is 0.6038062283737025 with following visualization:



Analysis on (1) (2) (3)

Evaluating on no-accident Hyundai and Cadillac cars:
(1) The clustering accuracy using Gaussian Mixture Model learnt from `all no-accident cars` is 62.63%.
(2) The clustering accuracy using Gaussian Mixture Model learnt from `all Hyundai cars and no-accident Cadillac cars` is 65.05%.
(3) The clustering accuracy using Gaussian Mixture Model learnt from `no-accident Hyundai cars and all Cadillac cars` is 60.38%.
Compared with (1), (2) increases 2.42%. It is because cars with accidents usually have lower prices and Hyundai is the cheaper brand compared with Cadillac. Thus, extra adding Hyundai cars with accident into learning will enhance the clustering performance compared with (1).
Compared with (1), (3) decreases 2.25%. It is because Cadillac is the more expensive brand compared with Hyundai and adding lower price Cadillac samples (with accident) will make the clustering boundary ambiguous.
2.42% is little larger than 2.25% which may indicate that the impact of accidents on Hyundai is larger than the impacts of accidents on Cadillac.
But it is not enough. Considering the number of sample contrasts between `Cadillac_data_with_accident` and `Hyundai_data_with_accident`: 299 vs. 120, but `Cadillac_data_with_accident` affects the Gaussian Mixture Model's performance less.
We can thus conclude that Cadillac cars' price is more robust against the accident records and Hyundai cars' price is sensitive towards the accident although it is a cheap used car brand.
We can approximate the accident record affecting Hyundai used cars' prices 2.68 times more than the influence on Cadillac used cars' prices by computing (2.42/120)/(2.25/299).

Reference

[1] Reynolds, Douglas A. "Gaussian mixture models." *Encyclopedia of biometrics* 741.659-663 (2009).
[2] https://scikit-learn.org/1.5/modules/mixture.html
[3] https://seaborn.pydata.org/tutorial.html
[4] https://plotly.com/python/