CRoss **I**ndustry **S**tandard **P**rocess for **D**ata **M**ining (*CRISP-DM*)

The Cross-Industry-Standard-Process for Data Mining is very important for data science projects. This is a widely used technique in the industry for data mining. It has a well-structured methodology and systematic approach to tackling a business problem. It is a flexible and iterative process that allows adjustments and changes as new data is available. It provides a valuable framework for managing and executing data-driven projects.
For the same reason, I have used this methodology for my project that predicts the customers that already have a car and are likely to buy a new one. The reports discuss how the results will help the marketing team for their marketing campaign.

Business Understanding:

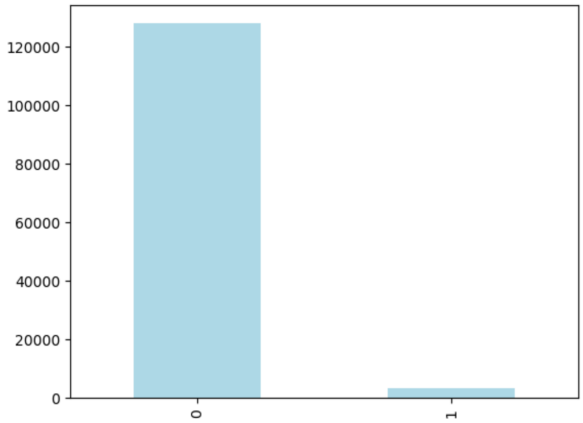
1. **Business-objective:**
   The current car market industry is competitive and innovative in its own ways. Today, a car is not only a luxury item but also a necessity because of the increased population, causing more crowds in public transportation. With car companies having different car models according to different price caps, buying a car becomes much easier in today's time. Selling a car to first-time buyers is relatively easier as they are more easily convinced, but targeting customers that already have a car and making them buy another car is a bit challenging as these people have a better understanding of the industry because of the experience of using the first car. Hence the main challenges for the marketing team are how to convince these buyers and what is the key area that they look at before buying a new car. By understanding what the main features are that customers look for before buying their second car and helping the marketing team to understand the car models, they are trying to sell and explain how it can be beneficial for them. Also, having the data of customers who are more likely to buy a car will help them to narrow down their targeted customers and provide better promotional offers according to their needs. This will, in turn, help them in reducing their marketing cost and using fewer resources, reducing the cost of the company and by doing more sales, they will add up to the revenue of the company.

2. **Data Understanding:**
   The given dataset had 131337 rows and 17 columns, indicating that the data is sufficiently large. The number of features is comparatively less but covers all the necessary information that is looked at when buying a car. Majority of the data was of numeric type and there were three columns (gender, car_segment, car_model) that had categorical values. These values were converted to numeric features as machine learning models do not run on numeric values. The encoding used to convert the numeric values was one hot encoding. The advantage of using this was it does not classify data in a specific order, thereby reducing the bias. Two features (age_band and gender) have a very high number of missing values. The age_band column was dropped as the number of missing values was very high. There were no duplicates in the data. Overall, all the features in the data did not provide any sensitive information regarding a particular customer, indicating that the data was safe and had no privacy concerns. The only column which could leak any kind of private information

was the ID column and all the other columns talked about the car. Hence the column was dropped.

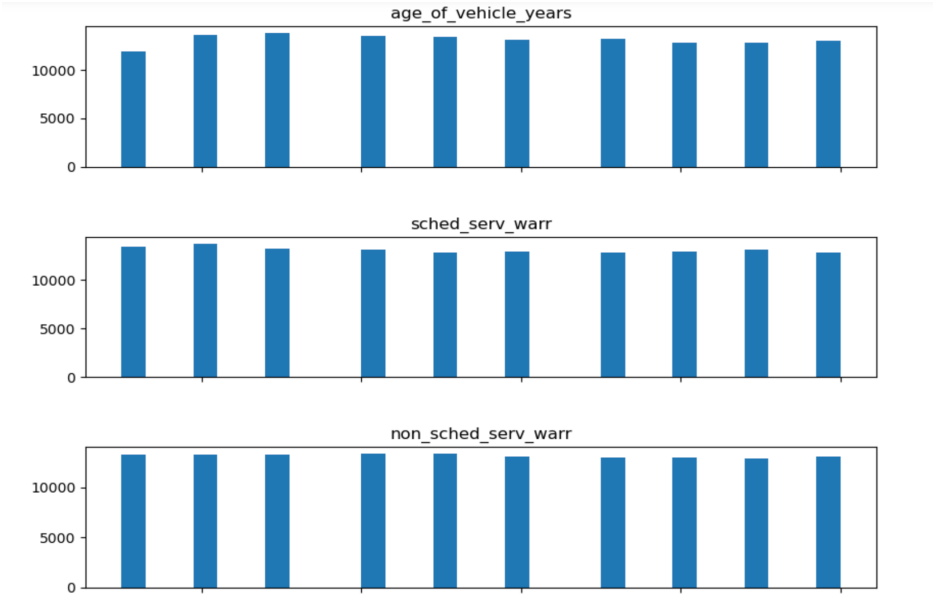
3. **Data Preparation:**

Using different data processing and preparation techniques, some key observations were made about the dataset. Firstly, the data in the dataset was very well normalised, and all the features had similar min, max and standard deviation values. This meant that the data did not need any scaling for better normalisation. Secondly, there was a class imbalance on the Target variable, with Target = 0 (Customers not likely to buy a car) being the majority class. Using different classifiers, we can understand how the model tackles the class imbalance and how well it can distinguish between the classes even after having low values in minority class (which is the case for all the real-world datasets)



Checking the values for class weight

Thirdly, the unique values inside the features indicate that the values are well-distributed and do not require any further cleaning.



Checking for the distribution of unique values in the selected features

4. **Modelling:**

A total of 4 different models were used to compare the performance. The comparison was made based on the performance metrics and the values in the confusion matrix. The performance metrics used were:

1. Accuracy – Measures the overall correctly predicted values.
2. Precision – Measures how many actual positive values were predicted.
3. Recall – Measures the actual positive sample to the actual positive values.
4. F1 Score – Provides the balance score between the Precision and Recall
5. AUC-ROC Score – Measures the overall performance of the model across all possible threshold values.

The hypothesis and predictions were made based on the cumulative scores from the metrics, and the primary focus was on the F1 scores. Since all the models were accurately predicting the majority class (Target = 0), my focus was to check how accurately are they predicting the minority class (Target = 1) as the marketing team wanted the customers that are more likely to buy a new car.

Classification report:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	25608
1	0.80	0.21	0.33	660
accuracy			0.98	26268
macro avg	0.89	0.60	0.66	26268
weighted avg	0.98	0.98	0.97	26268

Linear Regression Classification Report

Classification report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	25608
1	0.95	0.72	0.82	660
accuracy			0.99	26268
macro avg	0.97	0.86	0.91	26268
weighted avg	0.99	0.99	0.99	26268

SVM Classification Report

Classification report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	25608
1	0.84	0.79	0.81	660
accuracy			0.99	26268
macro avg	0.92	0.89	0.90	26268
weighted avg	0.99	0.99	0.99	26268

Decision Tree Classification Report

Classification report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	25608
1	0.96	0.77	0.85	660
accuracy			0.99	26268
macro avg	0.98	0.88	0.92	26268
weighted avg	0.99	0.99	0.99	26268

Random Forest Classification Report

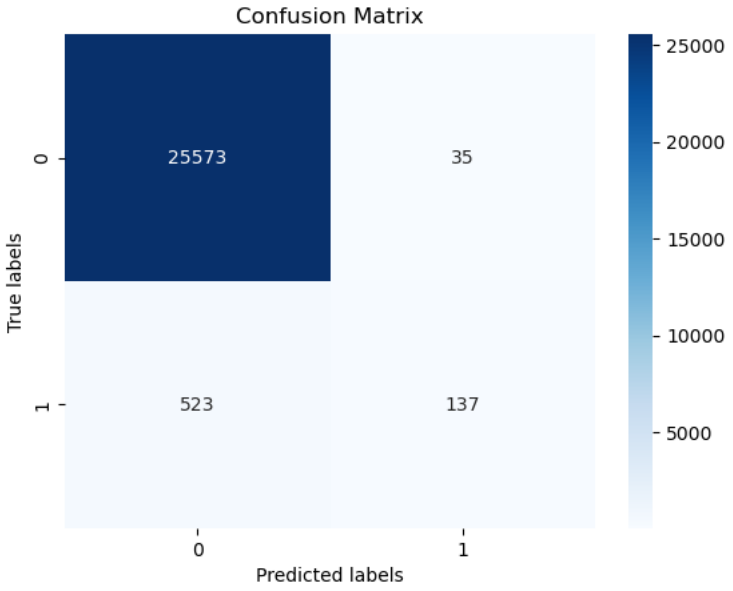
Classification report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	25608
1	0.93	0.79	0.85	660
accuracy			0.99	26268
macro avg	0.96	0.89	0.93	26268
weighted avg	0.99	0.99	0.99	26268

Random Forest with Feature Engineering Classification Report

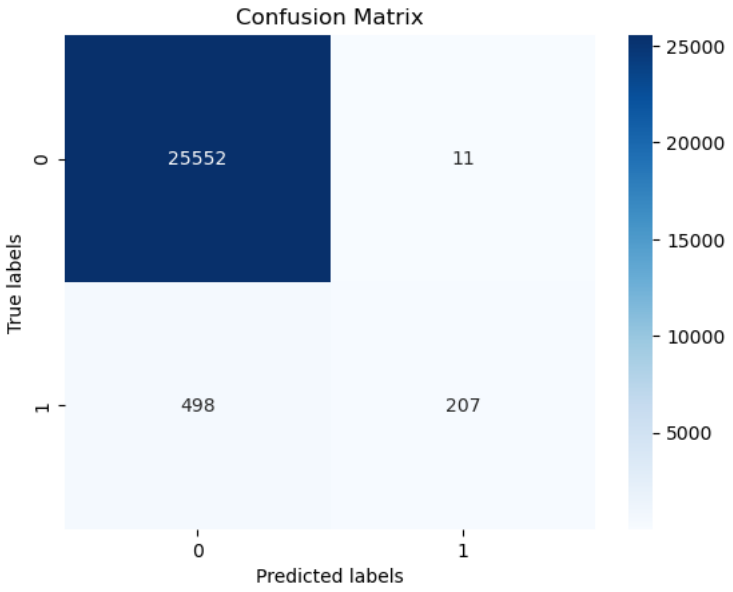
5. **Evaluation:**

Task 1: Using the logistic regression Classifier the model had high accuracy and precision but could not predict the positive cases very well and hence had a low f1 score.



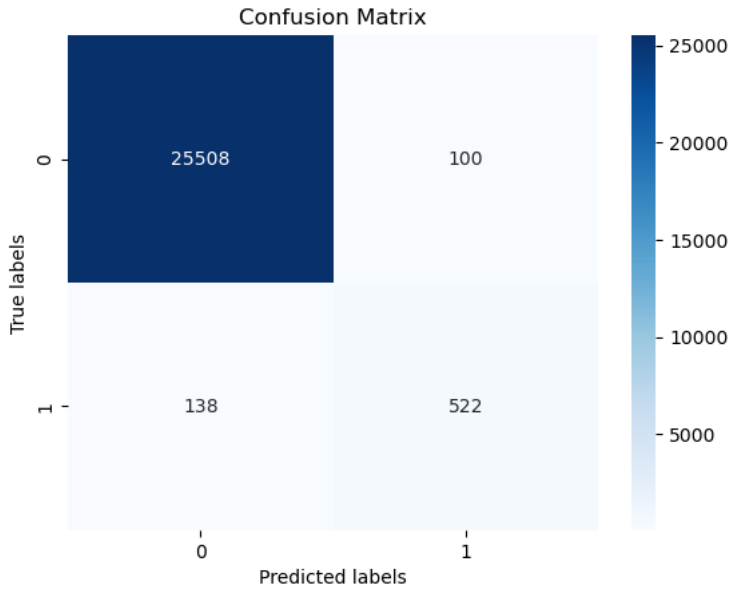
Confusion Matrix – Logistic Regression Classifier

Task 2: As SMV is a more complex model and understands the not linear data very well, the model had a high accuracy and precision score. The recall and f1 score had a drastic increase stating that the classifier could distinguish between the classes, but the AUC-ROC score suggests that the model is not fitting the data very well. SVM had the least possible false positive values, but very high false negative values.



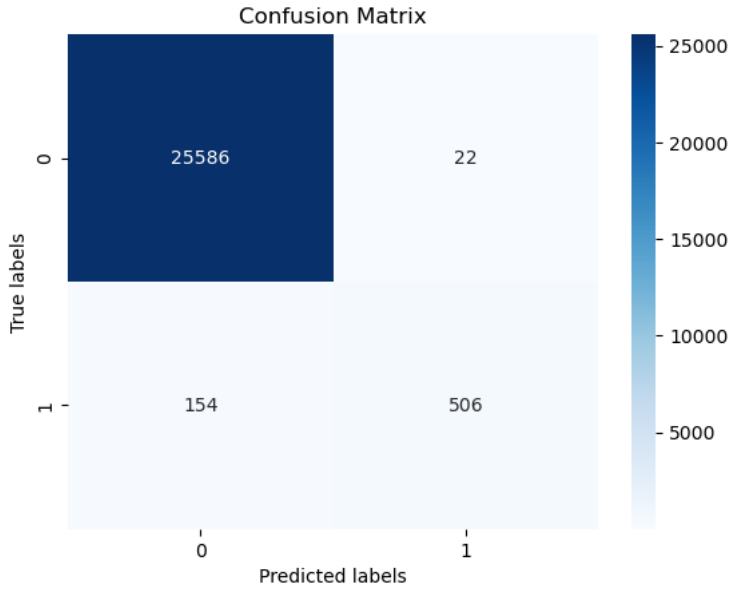
Confusion Matrix – SVM Classifier

Task 3: Decision tree was used as it splits the data into different branches and has better data penetration. The model performed well in classifying the values but was expected to perform better than SVM. It has lower Recall and F1 scores, but it balanced the false predicted values in the confusion matrix. The false positives and false negatives were in a similar range.



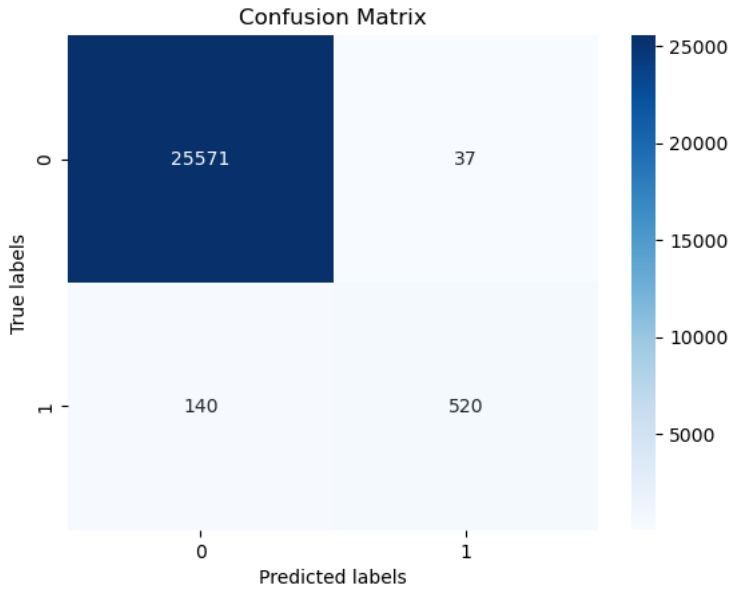
Confusion Matrix – Decision Tree Classifier

Task 4: Random Forest classifier uses the decision trees to make predictions on the data. The classifier has the best score among the previous one indicating that it best fits the model. The overall performance metrics were closer to 1, indicating that it could predict and distinguish the classes very well. The model has very low false positive values and comparatively high false negative values.



Confusion Matrix – Random Forest Classifier

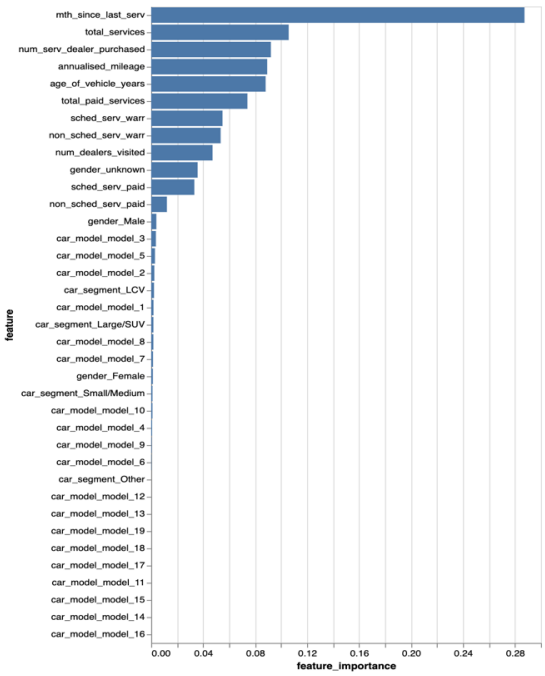
Task 5: The classifier was kept the same as the previous one. Feature engineering was performed by adding 3 new columns and dropping unimportant ones. GridSearchCV was performed to check the best parameters. The scores obtained were like the ones without performing any techniques indicating that a different approach should be taken to improvise on the false negative values.



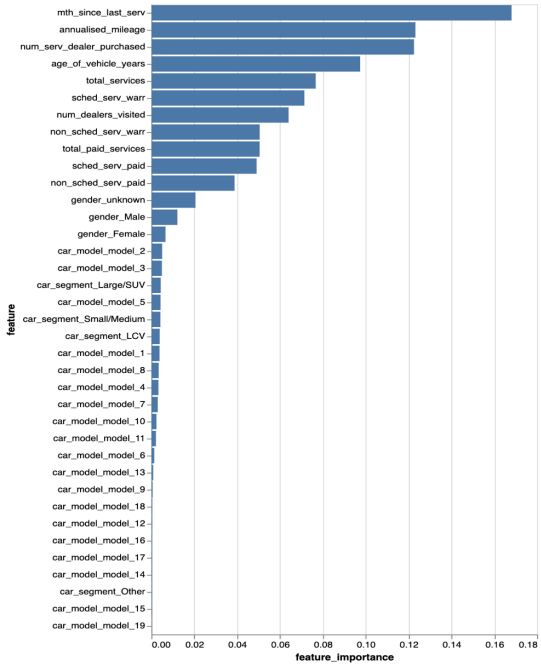
Confusion Matrix – Random Forest Classifier with Feature Engineering

6. **Deployment:**

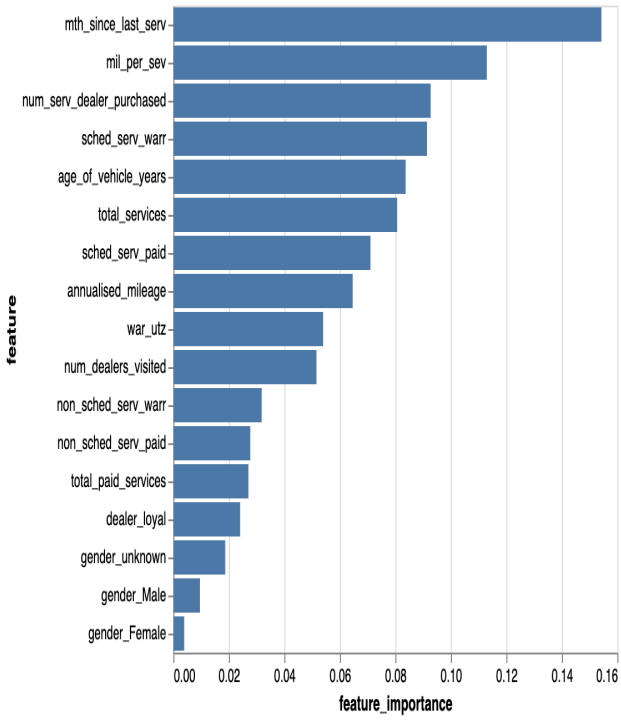
As the experiment progressed, the model kept performing better and better, indicating that the approach taken significantly impacted correctly predicting the Target value = 1. The overall performance metric scores suggest the classifiers selected could easily handle the data, except for logistic regression. Using the feature importance table in decision trees and random forests, a conclusion could be made about the most important features common in all three models. Noticeably, the features related to … are important in different models. As per the business target, the model achieved to reduce the false positive values, indicating the customers that are likely to buy a car would not be classified as customers not likely to buy. This means the company is not losing their potential customers. On the other hand, the values of false negatives were relatively high, which meant the company would waste more time convincing the customers that are predicted to buy, but actually do not want to buy. Although this is not a very significant loss to the company, they can still correctly target the customers likely to buy. Hence, I believe this model can be deployed if the marketing team focuses on selling more cars.



1. Feature Importance Decision Tree



2. Feature Importance Random Classifier



3. Feature Importance Random Classifier with feature engineering

Reference – The given data was provided by the professor.