# Business Understanding.

### Business Objective:

The objective of the business, a car dealership/showroom, is to make the process of selling cars to potential clients more efficient by streamlining the allocation of resources.

The primary objective here would be to analyse the dataset collected and find areas in which we can optimise resource allocation. By using this dataset and training an effective model, we can hope to classify customers into potential clients who would buy a new car and those who would not. This would thereby save money for the business, namely by reducing the requirement to send salesmen to every single possible client.

### Situation Assessment:

The resources allocated to us the data analyst include the dataset which contains information for the various potential clients and the relevant parameters of measurement for each field within the dataset, as well as a description for each field.

The project requires us to study the dataset and leverage the data to train a model that can help classify potential clients who would likely buy a new car and those who would not, i.e. the target variable (Target) by training a model using the other independent variables.

If successful in our training and modelling, we can use these predictions to better manage resources by focusing on clients that have a higher propensity to buy a new car than those who do not.

### Technical Assessment:

As far as the technical goals are concerned, we shall primarily focus on training various classification models to ensure the best prediction accuracy given the dataset. Secondary goals will include ensuring a fair and understandable scoring criteria as well as avoiding overfitting and bias.

### Project Plan:

Split into 2 major stages, this project will first explore the dataset and try to ascertain the best approach. Depending on the primary findings, we shall then train various classification models to separate the dataset into potential clients and not. The final stages will involve reduction of overfitting, tuning of hyper-parameters and evaluation of performance.

# 

# Data Understanding.

### Initial Data Collection:

As the dataset was provided by the business itself, there was no collection phase required. At most, the only step required was to download the folder and extract it to reveal the training dataset.

### 

### Data Description:

The dataset is a monolithic aggregate of data points, ranging in the 130,000s. This gives us an ample dataset to explore and train on, all the while effectively splitting a section on which we would perform testing and evaluation.

Due to the large nature of the dataset, for most of the models trained, the splitting of the data was performed using scikit\_learn’s train\_test\_split library into a training data frame which constituted 80% of the dataset, and the rest was used for testing purposes.

The raw dataset contains 17 columns and 131337 corresponding entries. The columns cover the various fields of age, gender, car models and segments, servicing time and costs and mileage information for the same. The descriptions for each field will be attached in the appendix of this report.

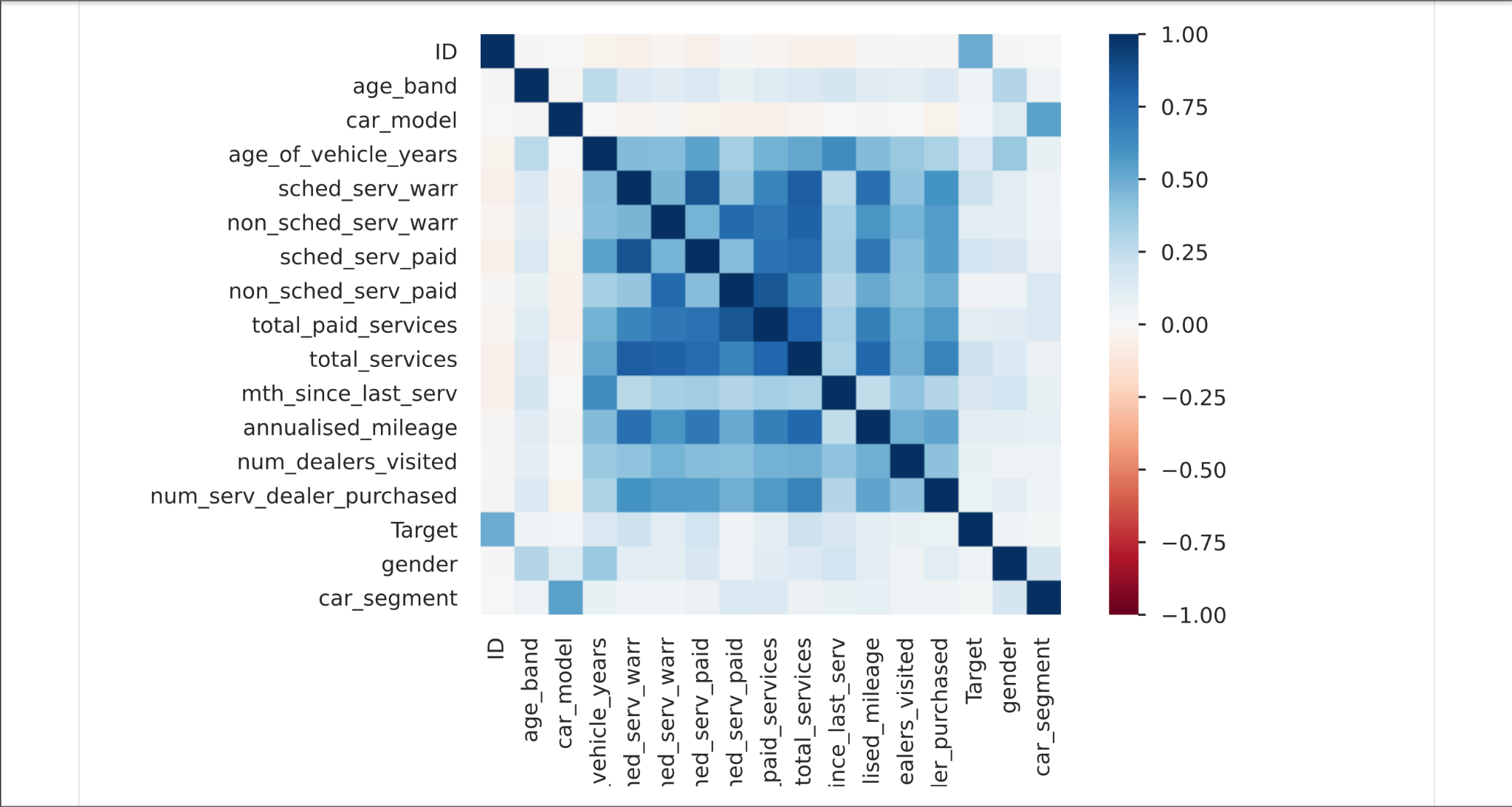
### 

### 

### Data Exploration:

Due to the large nature of the dataset, visual exploration and scatter graph plotting proved to be extremely difficult to perform and replicate. As such, it was imperative to use a more numerical and statistically descriptive approach.

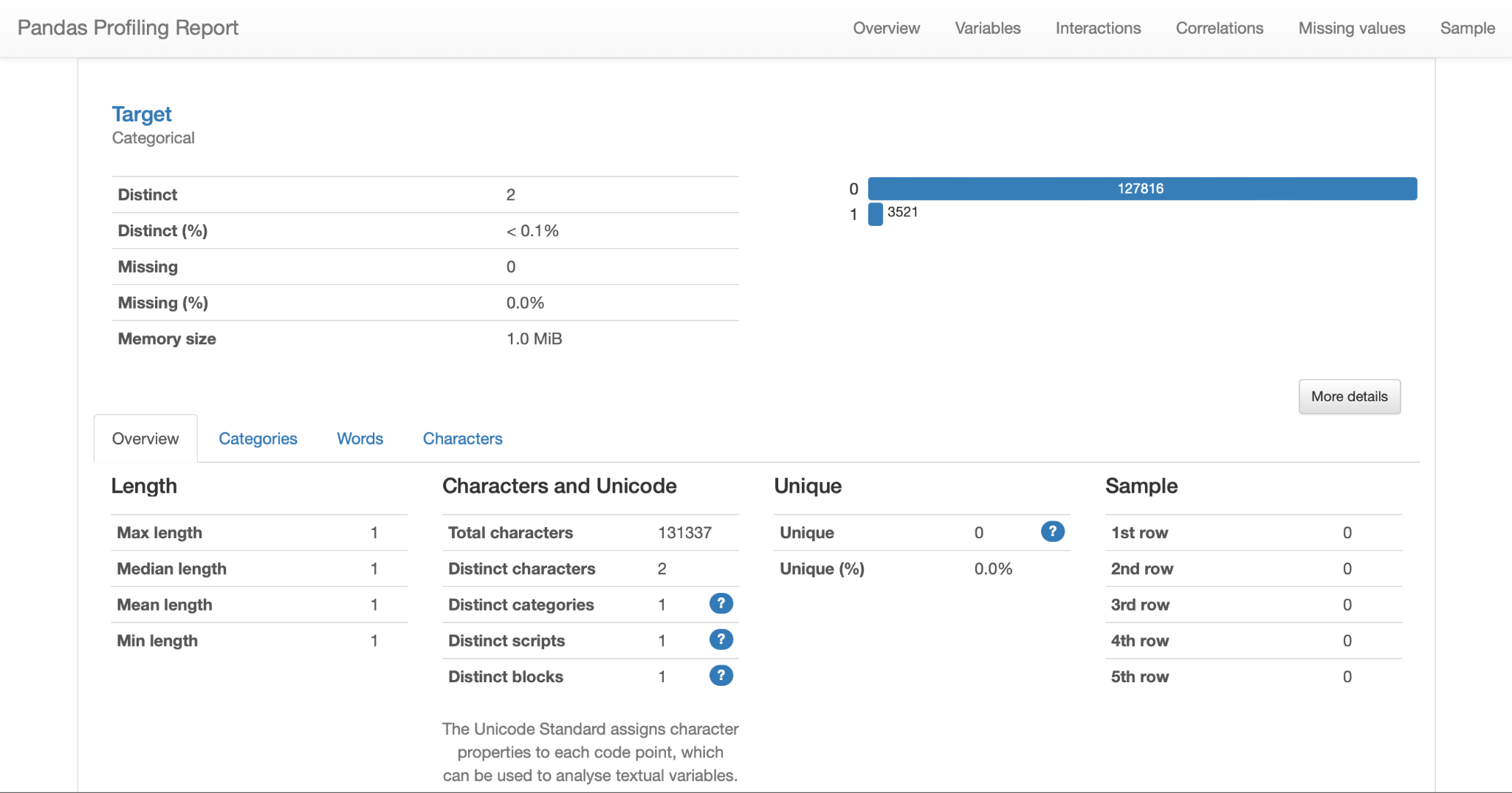
Building a heatmap for the same shows us the various variable interactions, allowing us to extract the columns with the most impact to ensure better results and removing unnecessary columns to avoid overfitting or bias.



Namely, the following fields hold the most correlation and maintain a relationship with the target variable:

["age\_of\_vehicle\_years", "sched\_serv\_warr", "non\_sched\_serv\_warr", "sched\_serv\_paid", "total\_paid\_services", "total\_services", "mth\_since\_last\_serv", "annualised\_mileage"].

The visual and descriptive statistics gathered are as follows :







### 

### Data Quality Verification:

Performing a visual inspection of the dataset, we notice quite a few values that do not have an actual value, or seem to be not filled. Of note, the columns for “age\_band” and “gender” seemed to be primarily filled with NaN values.

Apart from this, the dataset appears to be mostly clean and there is very little in terms of actual cleaning that would need to be done to use this dataset, however, this is taken care of in the next stage.

The dimensions of the dataset also might prove to be an issue when training, since the class weights are heavily imbalanced. Two approaches could be used to take care of this. The first approach would involve dropping the data points with NaN values, which would reduce the class weight disparity, but decrease the size of the testing data. The second approach would be to ensure hyper-parameters for class weight are implemented, and using a data dictionary to replace NaN values with suitable values.

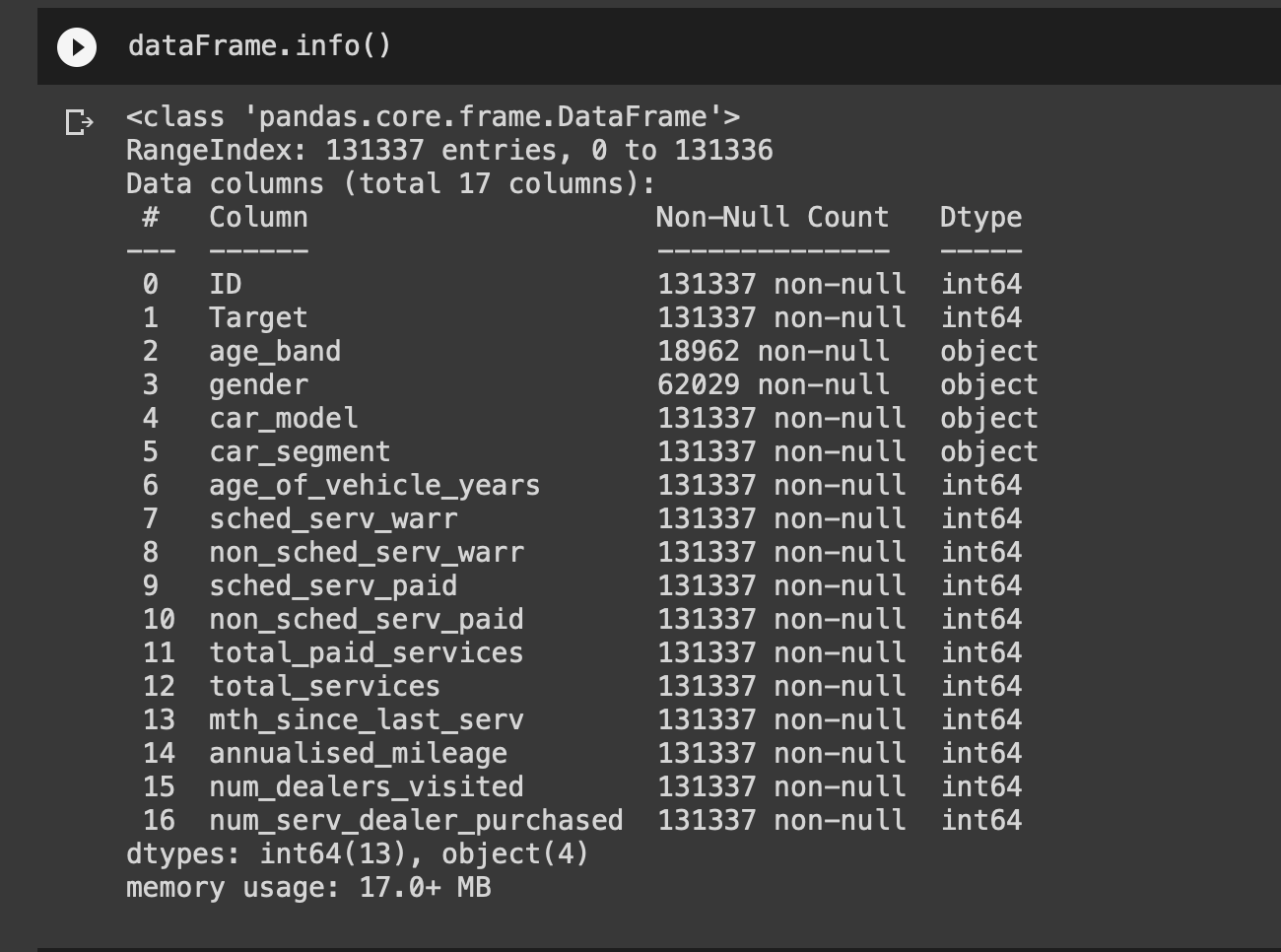
# Data Preparation:

In this stage of the project, we refer back to the fields we noticed as having outliers or inaccurate data. While the standard approach is to drop entries that contain NaN values, it would result in losing roughly half our dataset, which would not be a favourable approach.

Dropping rows works on the assumption that the missing data is unintentional or corrupted, however, in this specific business use case, it could simply have been intentionally left out by the customer to remain anonymous. As such I decided to use a pragmatic approach and created data dictionaries to replace values in these columns with new categorical fields to allow for a more apt use of the data points available.

Once this is done, we can verify the fields once again to ensure no null values or any other data quality issues and proceed with the next stages of this project. This stage of the data preparation and exploration is common across all experiments, and yields us the most data entries to train and test our models on.

The aforementioned data splitting in the ratio of training to testing data as 80:20 will also be performed herein.

Before cleaning.

# 

After cleaning.

genderDictionary = {"Male" : 1,

"Female" : 2,

numpy.nan : 3}

ageBandDictionary = {"1. <25" : 1,

"2. 25 to 34" : 2,

"3. 35 to 44" : 3,

"4. 45 to 54" : 4,

"5. 55 to 64" : 5,

"6. 65 to 74" : 6,

"7. 75+": 7,

numpy.nan : 0}

carSegmentDictionary = {"Small/Medium" : 1,

"Large/SUV" : 2,

"LCV" : 3,

"Other" : 4}

carModelDictionary = {"model\_1" : 1,

"model\_2" : 2,

"model\_3" : 3,

"model\_4" : 4,

"model\_5" : 5,

"model\_6" : 6,

"model\_7" : 7,

"model\_8" : 8,

"model\_9" : 9,

"model\_10" : 10,

"model\_11" : 11,

"model\_12" : 12,

"model\_13" : 13,

"model\_14" : 14,

"model\_15" : 15,

"model\_16" : 16,

"model\_17" : 17,

"model\_18" : 18,

"model\_19" : 19}

The Data Dictionary

# 

# Modelling:

As part of the modelling process, a myriad of models were trained as per the requirements, of which 6 models showed the best results and reproducibility. All of the models were classification models, building in some way and form upon each other, with a large aspect of tuning and evaluation at each stage.

The first model trained was a Logistic Regression Classification model. Being a straightforward model with a clear understanding of the process and outcome, I was more confident in starting out with this and branching out to the Tree based models later. The modelling process was easy enough, as demonstrated in the code, and I proceeded with the hyper-parameters for the “solver” and “class\_weight”, as they proved to modify the results ever so slightly in the direction of lower error but avoiding overfitting.

The second model in our experiment utilised a Support Vector Machine Classification model. While this was new territory for me, I presumed that given the large size of the dataset, we would obtain very effective results as visually separating the dataset could prove easier. Tuning and evaluating with the hyper-parameters for the “kernel”, “class\_weight”, “C” and “gamma”, we eventually obtained a model with a lower margin of false positives and negatives. However, a massive issue faced here was with respect to the large training time, with over 30 minutes required to run the code for tuning 4 hyper-parameters, which will be seen in the code.

Part C, D and E of our experiment utilised the three types of tree based classification models; a Decision Tree Classification, a Random Forest Classification model and a Extra Random Trees Classification model, with unique hyper-parameters for each. This section of the experiment caused the most stress in terms of overfitting and reducing this trend, causing us to carefully tune the hyper-parameters to ensure a better fit, all the while maintaining a high accuracy rate.

Part F of our experiment utilised a K Nearest Neighbours Classification model, which unfortunately did not perform well compared to the rest of the models. It proved to be very finicky to train, with the varying number of neighbours not shedding any new light or making it easier to work with. Additionally, given the size of the dataset, we had to invest quite a bit of time to get meagre improvements, and as such do not recommend this model for this dataset.

# 

# Evaluation:

With classification models in mind, there are a few metrics that can be considered for evaluation. The accuracy scores, precision score, F1 score, mean square error and mean absolute error are just a few to mention.

Each of these scores has a particular strength at identifying the overall metric of performance, however, for the businesses current use case, it made a lot more sense to use a confusion matrix wherever possible to identify the strength of the model.

As the primary goal is to predict which clients would buy a second car, we cannot rely on a single metric. This is in part to the fact that the dataset is massively imbalanced, and any testing would yield very high percentages of accuracy. Another aspect to consider is false positives v/s false negatives.

In an industry where you would rather send a salesman to a client who wouldn’t buy a car (false positive), than not send a salesman to a client who would buy a car (false negative), it is imperative to focus on the classification model’s performance specifically in these regions. While true positives and true negatives are important factors, it is the edge cases here that could potentially cost the business a large number of missed opportunities for a sale, as well as a large cost of resources in sending salesmen to unwilling purchasers.

The primary metric of comparison here was the use of a confusion matrix, as it best displayed the number of false positives and negatives. The secondary reason behind its use was the fact that the imbalance in the dataset proved to be a great hindrance to the accuracy score’s reliability.

The various models' performance metrics are as follows:

| Model Trained | Training Data Scores | Testing Data Scores |
| --- | --- | --- |
| Logistic Regression | [[76785 25488]  [ 373 2423]] | [[19098 6445]  [ 88 637]] |
| Support Vector Machine | [[98281 3927]  [ 33 2828]] | [[24575 1033]  [ 101 559]] |
| Decision Tree | [[95210 6998]  [ 98 2763]] | [[23896 1712]  [ 45 615]] |
| Random Forest | [[97052 5156]  [ 92 2769]] | [[24295 1313]  [ 48 612]] |
| Extra Random Trees | 0.06608038527063168 | 0.0678391959798995 |
| K Nearest Neighbours | [[102081 127]  [ 867 1994]] | [[25544 64]  [ 296 364]] |

With the exception of the K Nearest Neighbours Classification model, the other models seem to have performed very well on the training and testing data, showing a fair distribution for False Positives and Negatives, which indicates an avoidance of overfitting. The Extra Random Trees however do not allow us to visualise our results in the same manner, and so need to be analysed in a further experiment to better understand the extent of overfitting in them.

# Conclusion:

Of all the models trained, the Tree Based Classification models performed the best, while the K Nearest Neighbour model performed the worst. A large portion of the error and difficulty in ascertaining overfitting stems from the imbalance the dataset’s classes impose on the training and testing data’s evaluation metrics, however, given the confusion matrices, we can extract a satisfactory result.

Implementing our findings would not pose a difficulty as most of the error lies on the side of caution, i.e. we are not going to lose out on customers as much as we would send extra salesmen. However, this could be a silver lining as we are not excluding any potential customer as far as possible. Hence, the positives outweigh the negatives.

While age and gender information could bias our dataset, after performing our analysis we can conclude that the values show little correlation with the target variable and were removed altogether at the time of training, and so have zero bearing on the matter of actual deployment. As for the data privacy aspect, since most of the data points that would indicate personal information have a large amount of unfilled data, it poses no risk to the clients, and since most of such data has been dropped while training, it poses no risk of bias either.

By implementing the Logistic Regression model, or any of the classification models with results that satisfy the team in charge of changing policies, we can start predicting the likelihood of customer purchases, and suggest that the team in charge of sales to reallocate the distribution of salespeople and advertising material towards these individuals. Additionally, we can further analyse the dataset and attempt to figure out where the interest of those customers who wont buy second cars lies, and how to best cater to their needs.

It is the recommendation of us, the data analyst, to move forward with the results listed and shown above to positively influence the end goals of the business, and garner better returns. As mentioned above, there could be multiple different ways in which the different teams at the business could utilise the prediction data, namely by working with other complementary businesses to facilitate greater outreach and sales.

However, future suggestions to the business would include performing a study to better balance the dataset to ensure that future training yields results that are not overfitted or biased, as well as new methods to collect data revolving round the parameters that indicate high corelation with the target variable.

# Appendix.

**ID:** Unique ID of the customer

**target:** Model target. 1 if the customer has purchased more than 1 vehicle, 0 if they have only purchased 1.

**age\_band:** Age banded into categories

**gender:** Male, Female or Missing

**car\_model:** The model of vehicle, 18 models in total

**car\_segment:** The type of vehicle

**age\_of\_vehicle\_years:** Age of their last vehicle, in deciles

**sched\_serv\_warr:** Number of scheduled services (e.g. regular check-ups) used under warranty, in deciles

**non\_sched\_serv\_warr:** Number of non-scheduled services (e.g. something broke out of the service cycle) used under warranty, in deciles

**sched\_serv\_paid:** Amount paid for scheduled services, in deciles

**non\_sched\_serv\_paid:** Amount paid for non scheduled services, in deciles

**total\_paid\_services:** Amount paid in total for services, in deciles

**total\_services:** Total number of services, in deciles

**mth\_since\_last\_serv:** The number of months since the last service, in deciles

**annualised\_mileage:** Annualised vehicle mileage, in deciles

**num\_dealers\_visited:** Number of different dealers visited for servicing, in deciles

**num\_serv\_dealer\_purchased:** Number of services had at the same dealer where the vehicle was purchased, in deciles