Student Name	*Nathan Collins*
Project Name	*MLAA Assignment 2*
Date	*Apr 28 by 23:59*
Deliverables	*<MLAA_Assignment_2_1>* *<Experiment_Report_1>*

EXPERIMENT REPORT **[A]**

EXPERIMENT BACKGROUND

a. Business Objective

The overarching aim of the project is to utilise analytical and statistical methods to determine the likelihood that an existing customer of an automotive manufacturer will purchase a new vehicle. To achieve this, several experiments will be conducted using a provided historical dataset. The success of the model will empower business stakeholders to implement a cost-effective re-purchase campaign targeting existing customers who are prospective in purchasing a new or secondary vehicle.

The accuracy of the results will determine the value of the leads attained following the marketing campaign. Precise results may yield a positive return on investment by identifying these fulfilling customer leads, while incorrect results could lead to revenue loss through an unsuccessful marketing campaign, overstocking incorrect car varieties (models and segments) in anticipation of specific buyers, or granting unnecessary warranty and servicing inclusions with purchases, enacting a toll on the business, and damaging its longevity.

b. Hypothesis

Alternative hypothesis:

There is a relationship between some features in the car sales dataset and the likelihood that existing customers of an automotive manufacturer will purchase a new vehicle.

Null hypothesis:

There is no relationship between the features in the car sales dataset and the likelihood that existing customers of an automotive manufacturer will purchase a new vehicle.

The null hypothesis assumes no relationship is present between the dataset features, while the alternative hypothesis suggests that there is a relationship. By comparing the model’s performance against the null hypothesis, it determines whether the model is significantly better than random chance at predicting customer purchase behaviour.

Investigating these questions will assist with accomplishing the business goal by determining which existing customers who have purchased a second vehicle share certain features.

c. Experiment Objective

Experiment 1. will consist of a simple **univariate logistic regression** and **KNeighbour Classification model**, examining the **Target** variable (which determines if a customer has purchased more than 1 vehicle) against all other variables independently, each found in two separate data frames:

Table 1 Data frames applied for analysis and modelling.

Data Frame	Feature
cars_ALL	"*all features*" 18289 entries, 42 features
cars_NAG	"*no age-gender*" 128611 entries, 34 features

Working with two data frames facilitates a broader scope of outcome possibilities while retaining as many variables as possible without resorting to oversampling. As features such as **age_band** and **gender** offer decisive analytical data and insights into the classification of certain population groups, they will be retained where possible.

These features will undergo further classification through evaluation with a **KNeighbour classification model**, where integrating machine learning with a non-parametric, non-assumptive algorithm against a training dataset may classify new instances not recognised with a univariate logistic regression.

For this experiment to be deemed a success, it is anticipated that precision and recall values will result in high coefficients, close to a value of 1. If a low or no depicted coefficient value is produced, it will pave the way for further experimentation with more complex algorithms.

EXPERIMENT DETAILS

a. Data Preparation

Data Understanding

The data frame was explored to derive foundational insights, indicating the initial seventeen features, two of which contain NaN values (**age_band** and **gender**) and four consisting of object variables (**age_band**, **gender**, **car_model** and **car_segment**) which will need to be transformed. All remaining features are full and are integer values.

Data Cleaning

In order to prepare the data for analysis and modelling, the ID column was removed to reveal 2726 duplicate entries. As duplicate entries can weigh certain outcomes and impede data accuracy and consistency, these were dropped.

Next, the NaN values were totalled: 109668 in **age_band** (85.2% of total rows) and 67455 in **gender** (52.4% of total rows) and visualised with **missingno**. As age and gender offer alternative means of classification over the dataset, aside from car-related data, it was decided to retain these features in a separate yet more condensed dataset.

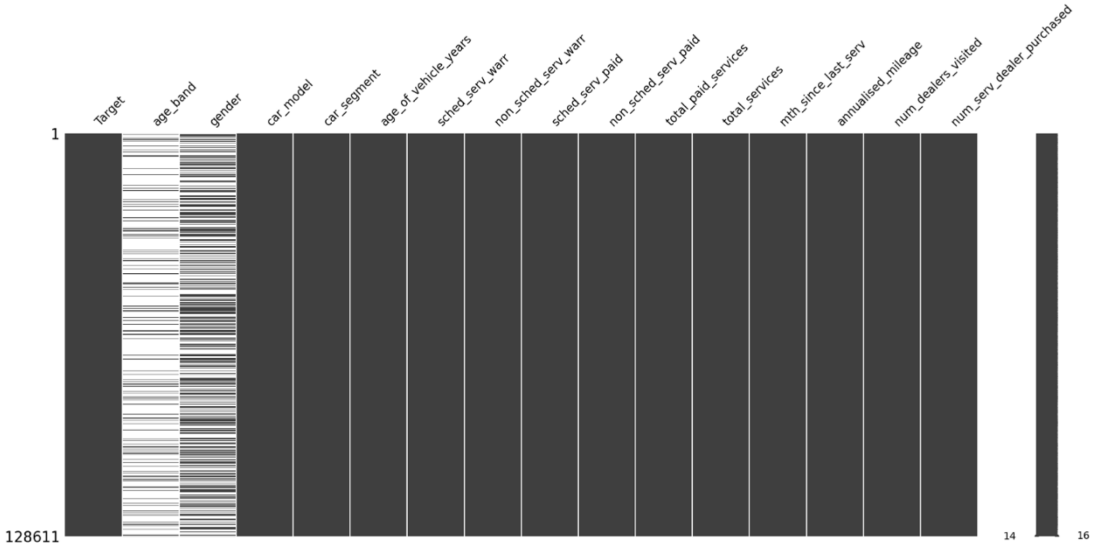


Figure 1 NaN values visualised with Missingno.

Producing Two Data Frames with All Values Converted to Integers:

cars_ALL and **cars_NAG** (see Table 1.)

For cars_ALL:

The **age_band** distribution was visualised prior to conducting **one-hot-encoding**, converting the feature into six rows with integer values of 1 and 0.

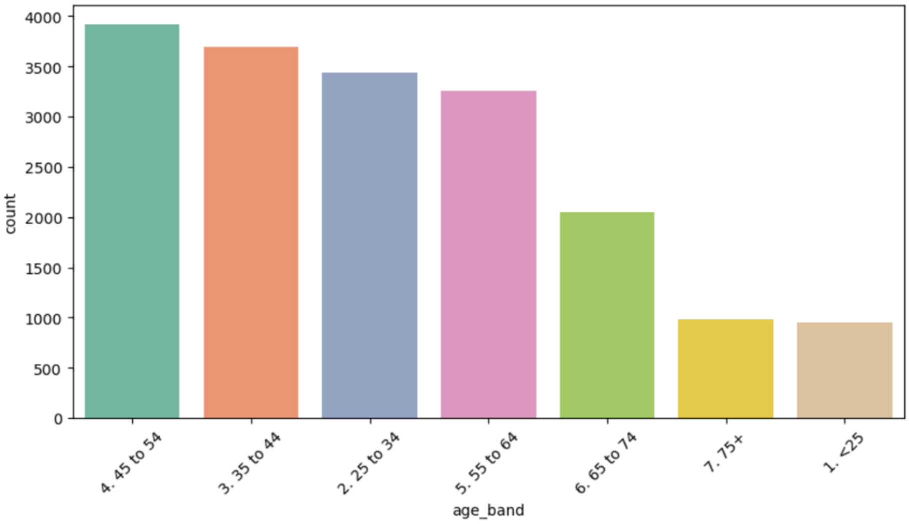


Figure 2 The age_band feature visualised.

One-hot-encoding was subsequently performed on the **gender** feature, converting the variable into two rows with integer values of 1 and 0, male and female.

For cars_ALL and cars_NAG:

The **car_segment** distribution was also visualised across both datasets prior to **one-hot-encoding**, converting the feature into four separate features with integer values of 1 and 0, Small/Medium, Large/SUV, LCV and Other

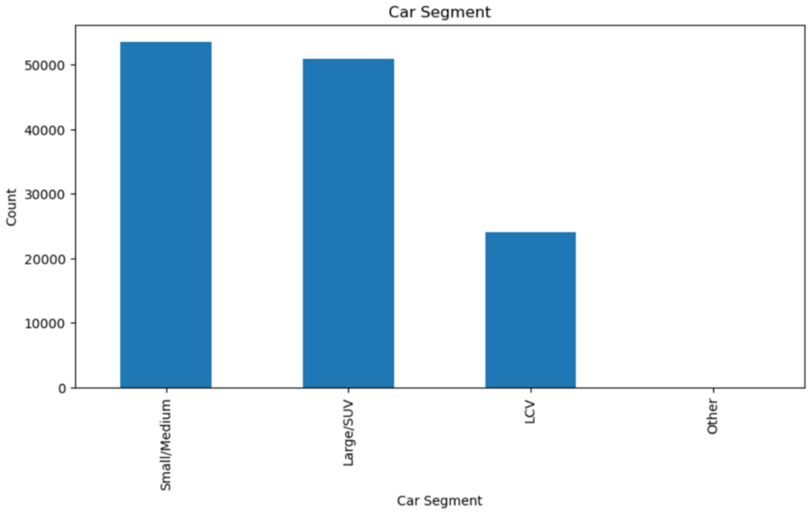


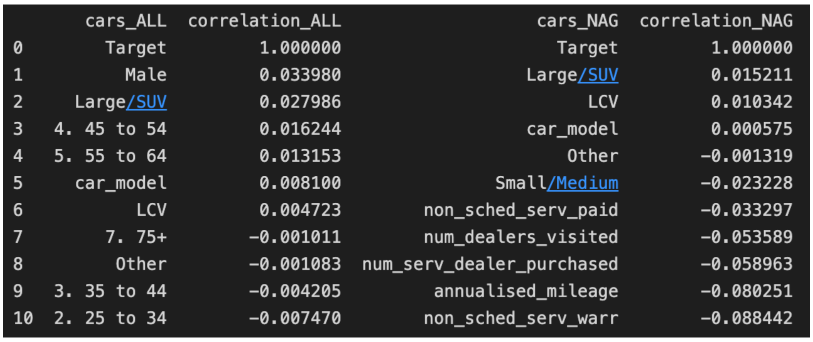
Figure 3 The car_segment feature of the cars_NAG data frame, visualised.

The **car_model** feature was the last categorical variable converted to an integer feature through **one-hot-encoding**, yielding nineteen features representing different car models with integer values of 1 and 0.

The ranges of each dataset were visualised as a final measure to ensure no outliers or abstract inputs were present.

Exploratory Data Analysis

Despite the problem residing in binary classification, linear correlations of all features against all other features of the dataset were first visualised using a heatmap, where the top 10 correlating features for each dataset were visualised in a table.



```
       cars_ALL  correlation_ALL                  cars_NAG  correlation_NAG
0        Target         1.000000                    Target         1.000000
1          Male         0.033980                 Large/SUV         0.015211
2     Large/SUV         0.027986                       LCV         0.010342
3   4. 45 to 54         0.016244                 car_model         0.000575
4   5. 55 to 64         0.013153                     Other        -0.001319
5     car_model         0.008100              Small/Medium        -0.023228
6           LCV         0.004723        non_sched_serv_paid        -0.033297
7        7. 75+        -0.001011        num_dealers_visited        -0.053589
8         Other        -0.001083  num_serv_dealer_purchased        -0.058963
9   3. 35 to 44        -0.004205           annualised_mileage        -0.080251
10  2. 25 to 34        -0.007470        non_sched_serv_warr        -0.088442
```

Figure 4 The cars_ALL dataset and the cars_NAG dataset highest correlating features.

Next, the percentage of the **Target** variable class of interest (integer values of 1) was visualised with a pie graph. Indicating that only 2.7% of the 128611 entries were the population group of interest. This interpretation establishes the dataset as unbalanced.

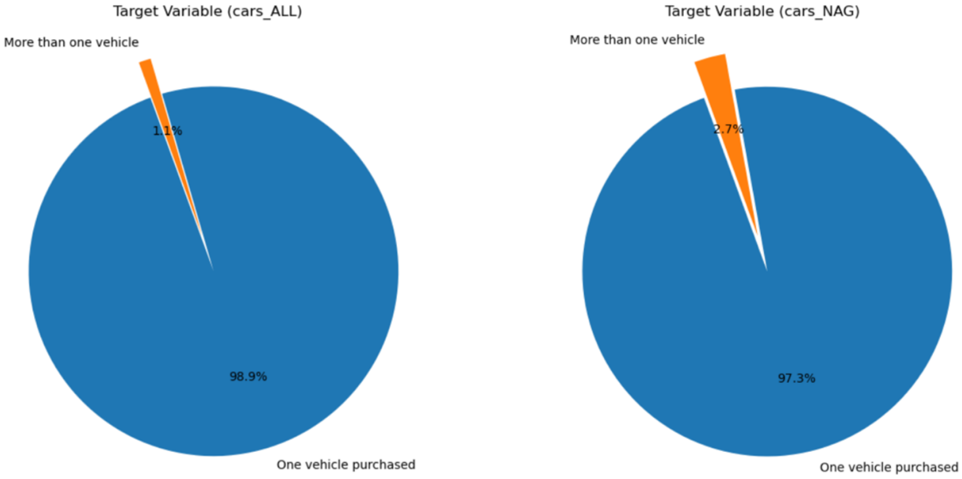


Figure 5 Pie graph visualisations of the Target variable and the class of interest.

Further visuals were subsequently constructed to gauge the perceived influence of the **gender** and **age_band** variables against other key classification features, such as the total, which share the **Target** variable and the relative **car_segment** and **car_model** choices made with their second car purchases. Overall it was interpreted that males tended to purchase a second car more than females, with the largest age brackets resting between 45 to 54. Out of these groups, the highest interest resided with Large/SUVs, models 5 and 3, among males. It is important to note that these features represent marginal percentages of the total and are not a full representation of the target cohort.

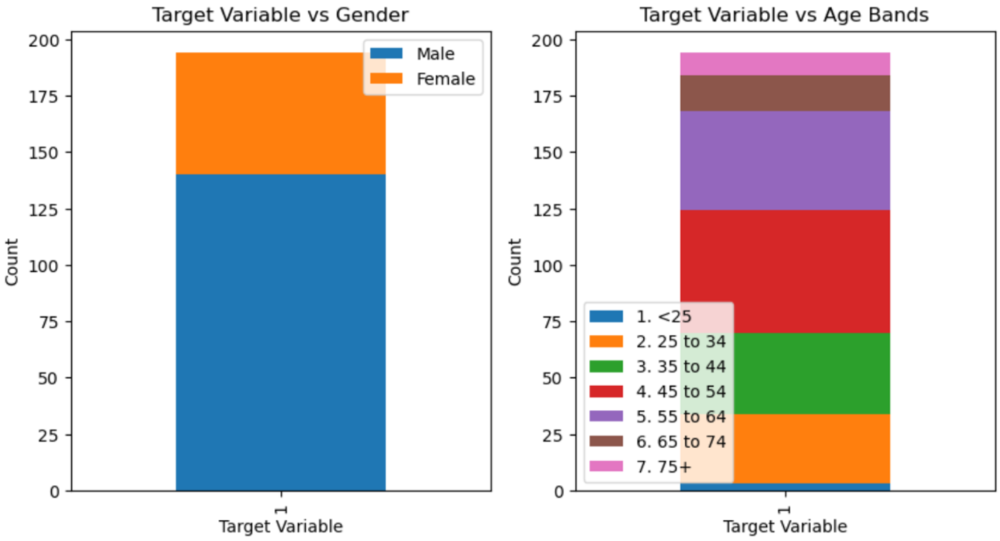


Figure 6 A segmented bar graph illustrating the male and female cohorts, beside the age band cohorts, with a Target value of 1.

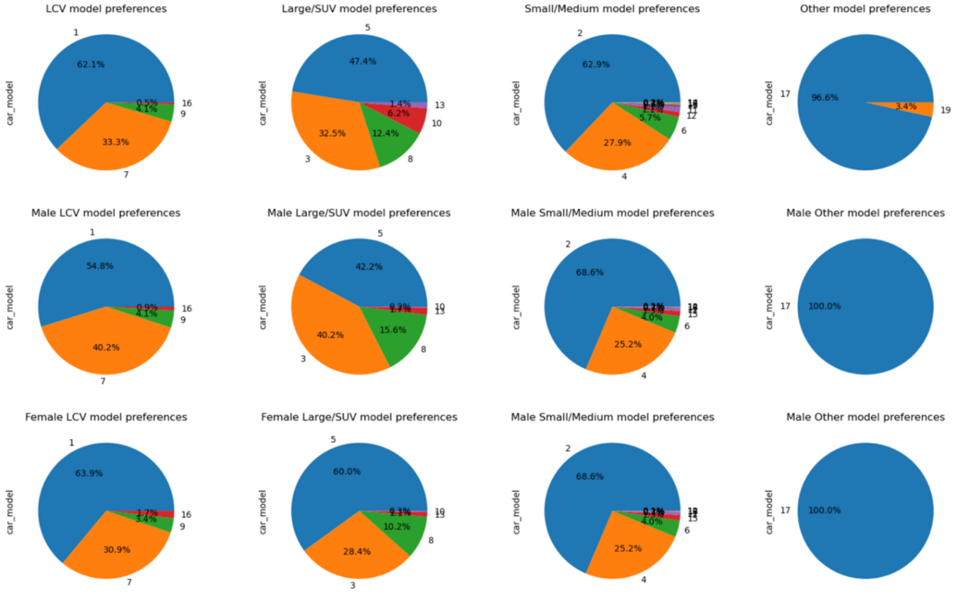


Figure 7 A series of pie charts illustrating the popularity of car models against the variety of car segments. The first row covers the overall majority, while the second and third rows cover the male and female preferences.

The final key exploratory analysis involved charting the frequencies of the **Target** variable with a value of 1 against **total_services**, **annualised_mileage**, **age_of_vehicle_years**, and **non_sched_serv_war**. The value was to visualise whether specific features of a customer's existing had an influence on their decision to purchase a new one. A key insight derived from this is that recurring customers typically engage with 2 to 3 total services (deciles), with slightly more non-scheduled services.

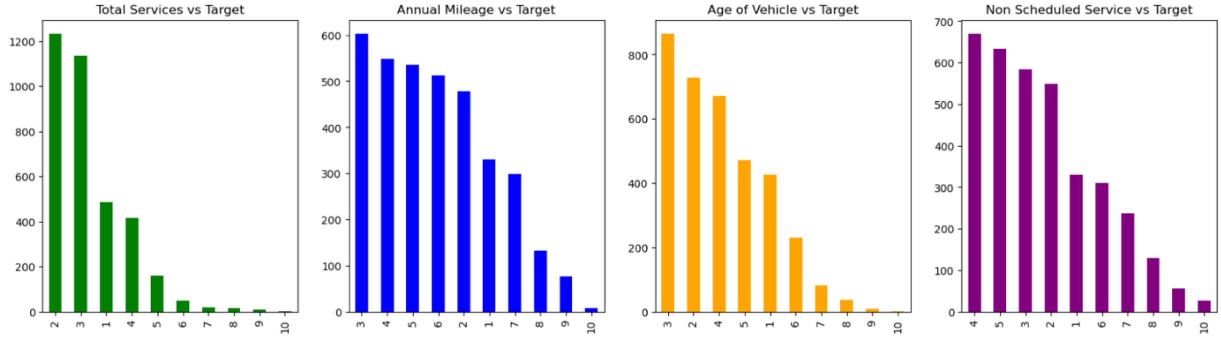


Figure 8 Four bar graphs illustrating total_services, annualised_mileage, age_of_vehicle_years, and non_sched_serv_war in deciles, against the target variable with a value of 1.

b. Feature Engineering

*< Aside from applying one-hot-encoding on **age_band**, **gender**, **car_model** and **car_segment**, no further feature engineering was conducted for Experiment 1. >*

c. Modelling

Selecting a performance metric

As the business objective resides with a binary classification problem, **Precision** was selected as the key performance metric to determine the model's success. Precision examines the proportion of true positives among the total positive predictions. By applying precision, a reduction in predicting the wrong customer is achieved. Should precision metrics share output similarity across models, an additional metric, **Recall**, will also be evaluated. Recall examines the proportion of true positives among all actual positive outcomes, meaning higher recall results in more lost opportunities for the marketing campaign.

Establishing the model:

All models across all experiments, will be constructed through a Python function. The following establishes consistency across all metrics within the data set and the transformation of the data prior to and during machine learning. The models selected for this stage of the experiment were intended to act as foundational models, representing the overall classification viability of the dataset prior to investing time with more complicated algorithms.

Univariate Logistic Regression

$$P = \frac{e^{a+bX}}{1+e^{a+bX}}$$

Complexity: Simple

A function for **univariate_logistic_regression** was defined, where subsequent iterations of the function would interchange the **dataset**, **feature**, **target,** and **model**. The function would first drop all null values (**.dropna**) as a precautionary practice (despite conducting this previously in the data cleaning phase), create an **X** and **y** variable, and split the data into a training and testing set with an embedded stratification (operating under the conventional **80% training / 20% testing** approach, where 80-20 was selected over smaller training sets due to the limited rows available within the **cars_ALL** data frame and the unbalanced class of interest in the **Target** variable). The function would then instantiate and fit (**.fit**) the model with **X_train** and **y_train**, predict (**.predict**) a test set and report the precision and recall scores as performance metrics (**precision_score**, **recall_score**) for both the training and testing sets.

An additional function **univariate_logistic_regression_confusion_matrix** was also created, with similar precursor steps as the previous, to display a **confusion matrix**, a table that displays the number of true positives, true negatives, false positives and false negatives.

No baseline metric (assessing null accuracy) was applied to compare performance against naïve predictions and determine whether the model is adding value. This was selected, as the majority of the data in the **Target** cohort is negative, making the null accuracy equal to the proportion of negative samples.

These finalised functions were first applied to the **cars_ALL** and **cars_NAG** data frames, defining the data set, feature, **Target** and model with each iteration.

KNeighbour Classification

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Complexity: Simple

To carry out a KNeighbour classification, the previous **univariate_logistic_regression** function was applied, where the “model” definition was substituted from **LogisticRegression()** to **KNeighborsClassifier()**. The function would first drop all null values (**.dropna**) as a precautionary practice, create an **X** and **y** variable, and split the data into a training and testing set. The function would then instantiate and fit (**.fit**) the model with **X_train** and **y_train**, predict (**.predict**) a test set and report the precision and recall scores as performance metrics (**precision_score**, **recall_score**) for both the training and testing sets.

The same procedure was applied with the **univariate_logistic_regression_confusion_matrix** function, generating a confusion matrix for the training and testing sets, further applying no baseline metric due to the majority class of the **Target** variable.

Models to consider for future experiments:

Multivariate Logistic Regression – evaluating relationships between features.

Support Vector Classification

- Kernel functionality for higher dimensional class separation.
- Strong when boundaries between classes are clear.
- Capable of handling large datasets.

Random Forest Classification

- Reduced overfitting functionality through decision trees.
- Accounts for missing values, outliers, and nonlinearity.
- Wide range of hyperparameter tweaking possibilities.

XGBoosting Classification

- Compounds simpler models to build a larger analysis.
- Capable of accounting for missing data.
- Requires additional data preparation and parameter tuning than previous algorithms.

Models not selected:

Univariate **Linear** Regression – not applicable for binary classification.

EXPERIMENT RESULTS

a. Technical Performance

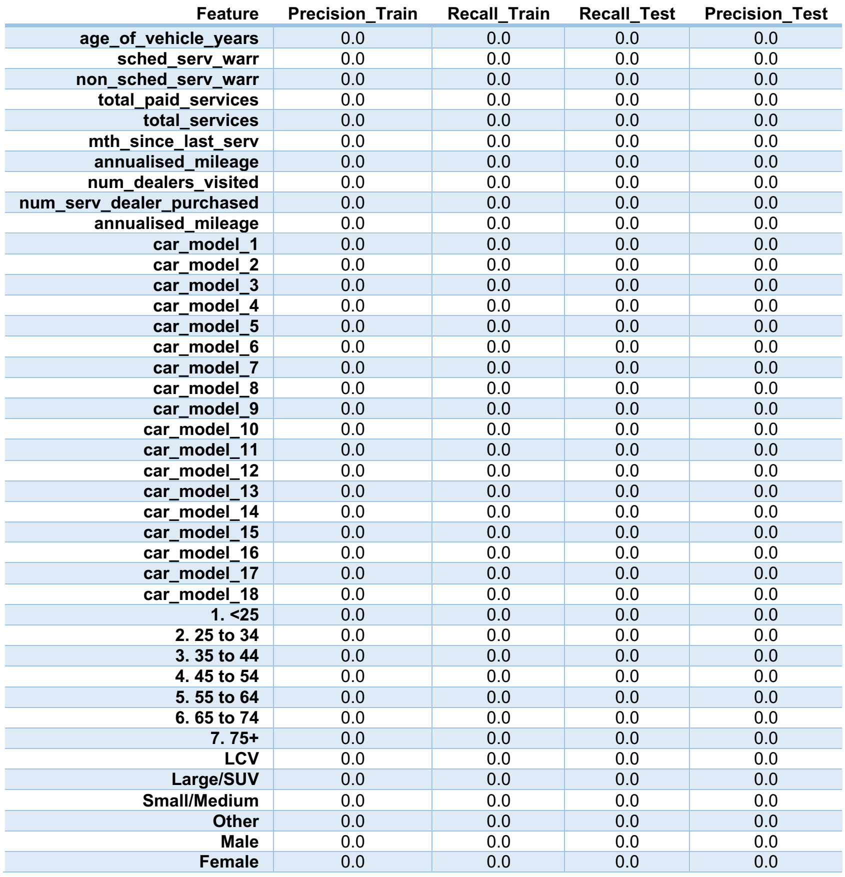
Evaluating the performance metrics

The selected performance metrics were **precision** and **recall**. Both the univariate logistic regression and KNeighbour classifier functions included subsequent loops to carry out regressions across all features and input their data into an empty Pandas data frame.

Technical evaluation

The first set of tables displays the precision and recall output of the **cars_ALL** data frame, while the second set displays the result for the cars_NAG data frame. Both tables illustrate an unsuccessful performance with no classification of the target variable against any of the features, as all results equal 0 (see Table 3 and Table 6). The subsequent confusion matrix illustrates 99% true negatives and 0% true positives. The metric suggests that the model is predicting only the majority class, with no positive predictions across both training and testing sets (see Tables 4 & 5 and Tables 7 & 8). The root cause of this outcome is likely the lack of sophistication in the model algorithm, where its simplicity isn't enough to cover the classification ranges present in the data. It is likely that the data imbalance and limitations of features contribute to this outcome as well.

Table 3 Precision and Recall scores of the univariate logistic regression function and the KNeighbours classification function on the cars_ALL data frame.



Feature	Precision_Train	Recall_Train	Recall_Test	Precision_Test
age_of_vehicle_years	0.0	0.0	0.0	0.0
sched_serv_warr	0.0	0.0	0.0	0.0
non_sched_serv_warr	0.0	0.0	0.0	0.0
total_paid_services	0.0	0.0	0.0	0.0
total_services	0.0	0.0	0.0	0.0
mth_since_last_serv	0.0	0.0	0.0	0.0
annualised_mileage	0.0	0.0	0.0	0.0
num_dealers_visited	0.0	0.0	0.0	0.0
num_serv_dealer_purchased	0.0	0.0	0.0	0.0
annualised_mileage	0.0	0.0	0.0	0.0
car_model_1	0.0	0.0	0.0	0.0
car_model_2	0.0	0.0	0.0	0.0
car_model_3	0.0	0.0	0.0	0.0
car_model_4	0.0	0.0	0.0	0.0
car_model_5	0.0	0.0	0.0	0.0
car_model_6	0.0	0.0	0.0	0.0
car_model_7	0.0	0.0	0.0	0.0
car_model_8	0.0	0.0	0.0	0.0
car_model_9	0.0	0.0	0.0	0.0
car_model_10	0.0	0.0	0.0	0.0
car_model_11	0.0	0.0	0.0	0.0
car_model_12	0.0	0.0	0.0	0.0
car_model_13	0.0	0.0	0.0	0.0
car_model_14	0.0	0.0	0.0	0.0
car_model_15	0.0	0.0	0.0	0.0
car_model_16	0.0	0.0	0.0	0.0
car_model_17	0.0	0.0	0.0	0.0
car_model_18	0.0	0.0	0.0	0.0
1. <25	0.0	0.0	0.0	0.0
2. 25 to 34	0.0	0.0	0.0	0.0
3. 35 to 44	0.0	0.0	0.0	0.0
4. 45 to 54	0.0	0.0	0.0	0.0
5. 55 to 64	0.0	0.0	0.0	0.0
6. 65 to 74	0.0	0.0	0.0	0.0
7. 75+	0.0	0.0	0.0	0.0
LCV	0.0	0.0	0.0	0.0
Large/SUV	0.0	0.0	0.0	0.0
Small/Medium	0.0	0.0	0.0	0.0
Other	0.0	0.0	0.0	0.0
Male	0.0	0.0	0.0	0.0
Female	0.0	0.0	0.0	0.0

Table 4 & 5 Confusion matrix results of the univariate logistic regression function and the KNeighbours classification function on the cars_ALL data frame.

CM_train	
14476	0
155	0

CM_test	
3619	0
39	0

Table 6 Precision and Recall scores of the univariate logistic regression function and the KNeighbours classification function on the cars_NAG data frame.

Feature	Precision_Train	Recall_Train	Recall_Test	Precision_Test
age_of_vehicle_years	0.0	0.0	0.0	0.0
sched_serv_warr	0.0	0.0	0.0	0.0
non_sched_serv_warr	0.0	0.0	0.0	0.0
total_paid_services	0.0	0.0	0.0	0.0
total_services	0.0	0.0	0.0	0.0
mth_since_last_serv	0.0	0.0	0.0	0.0
annualised_mileage	0.0	0.0	0.0	0.0
num_dealers_visited	0.0	0.0	0.0	0.0
num_serv_dealer_purchased	0.0	0.0	0.0	0.0
annualised_mileage	0.0	0.0	0.0	0.0
car_model_1	0.0	0.0	0.0	0.0
car_model_2	0.0	0.0	0.0	0.0
car_model_3	0.0	0.0	0.0	0.0
car_model_4	0.0	0.0	0.0	0.0
car_model_5	0.0	0.0	0.0	0.0
car_model_6	0.0	0.0	0.0	0.0
car_model_7	0.0	0.0	0.0	0.0
car_model_8	0.0	0.0	0.0	0.0
car_model_9	0.0	0.0	0.0	0.0
car_model_10	0.0	0.0	0.0	0.0
car_model_11	0.0	0.0	0.0	0.0
car_model_12	0.0	0.0	0.0	0.0
car_model_13	0.0	0.0	0.0	0.0
car_model_14	0.0	0.0	0.0	0.0
car_model_15	0.0	0.0	0.0	0.0
car_model_16	0.0	0.0	0.0	0.0
car_model_17	0.0	0.0	0.0	0.0
car_model_18	0.0	0.0	0.0	0.0
LCV	0.0	0.0	0.0	0.0
Large/SUV	0.0	0.0	0.0	0.0
Small/Medium	0.0	0.0	0.0	0.0
Other	0.0	0.0	0.0	0.0

Table 7 & 8 Confusion matrix results of the univariate logistic regression function and the KNeighbours classification function on the cars_NAG data frame.

CM_train	
100071	0
2817	0

CM_test	
25019	0
704	0

b. Business Impact

As the results of the **univariate logistic regression** and **KNeighbour classification** bear no real data about the relationship between the **Target** variable and any other variable in the corresponding dataset, it is unfortunately deemed an unsuccessful experiment and provides no input towards meeting the business objective. This data will ultimately not aid in evaluating existing customers to procure leads for marketing.

The results are verified to be accurate through examining the confusion matrix, though as this experiment model will not be provided to the business, there is no presently concern over providing inaccurate data and the ramifications which would follow.

c. Encountered Issues

Preliminary exploratory data analysis

Following an exploration of the missing values present within the **age_band** and **gender** features, deliberation and evaluation of their value ensued, as the weight in retaining these classification features needed to be considered. While the decision to retain these features in a separate data frame was decided, it ultimately scaled the workload. This decision provided avenues to drop the data frame following evaluation following further experiments.

Discerning performance metric suitability

As applying **precision** alone can result in similar outcomes between models, the choice to simultaneously incorporate a **recall** metric in conjunction would serve to aid in gauging the distinction between future modelling scenarios.

Coding learning curves and mitigating human error

After concluding that the most time-effective method in meeting the business goal was through constructing a classification function with an embedded splitting, model fitting and scoring capacity, establishing an approach occupied days of my time. It is anticipated that with practice, this issue will be avoided for future projects. By consulting library descriptions, forums and advisory web pages, the solution was found by interchanging the **dataset, feature, target,** and **model** with each function iteration. This issue is fortunately lessened with future experiments, as understanding the foundation to build future models was a priority.

Overcoming human error likewise presented itself as an inconvenience. As the functions are quite large, duplicating each and interchanging the variables inputted resulted in some initial incorrect results, remedied through constant revision and repetition.

Time limitations

Given the scale of the project, managing lifestyle responsibilities, and, more prominently, the **processing times** required to generate and run each cell (namely the KNeighbour classification), scheduling was applied to meet the business objective by the deadline.

FUTURE EXPERIMENT

a. Key Learning

Exploratory Data Analysis

As a preface, note that some key reflections of the exploratory data analysis represent minor percentages of the total data frame provided, in addition to representing even smaller percentages when factoring in entries with a target value of 1 (3521 rows).

- While some features express marginally stronger **linear** correlations to the **Target** variable than others (**Male, Large/SUV, age_band 45 to 54**), none express more than 4% correlation. This analysis is presented to emphasise the distinction between linearity and classification problems.
- Only 2.7% of the total variables in the Target feature represent the class of interest across all completed features (**cars_NAG**, see Figure 5)
- Out of the customers surveyed, the proportion of males who provided information about their gender was larger than the proportion of females in regard to the likelihood of purchasing a new vehicle.
- The most popular **car_segment** across both genders was Large/SVU.
- Proportionately younger age bands of the total surveyed (25 to 54) prefer Large/SUV vehicles, while older age bands (55 to 64) prefer Small/Medium vehicles.
- There are varying preferences for **car_segment** and **car_model** depending on gender (see Figure 7).
- Out of the customers surveyed, the proportion of “45 to 54 year olds” who provided information about their age band was the highest in regard to the likelihood of purchasing a new vehicle.
- While marginal, there is a proportionately lower **age_of_vehicle_years** in older customers.
- Of the 2.7% (3521) **Target** class with a value of 1, the majority make a second purchase following a value count of 2 to 3 services (deciles, 20 to 30, Figure 8).

Univariate logistic regression

- Following modelling for a univariate logistic regression and machine learning, no key insights into the data and feature importance could be extracted. All performance metrics yielded 0, with a confusion matrix majority of true negatives and no true positives. Because of this, further use of univariate models in this project will be terminated.
- No discernible data can be provided to the business to meet the objective.
- As only a simple model has been applied, multivariate models should be explored next, as these models offer tools that may offer depth and tuning approaches.

KNeighbour classification

- Following modelling for a KNeighbour classification and machine learning, no key insights into the data and feature importance could be extracted. All performance metrics yielded 0, with a confusion matrix majority of true negatives and no true positives.
- No discernible data can be provided to the business to meet the objective.
- Future experiments should explore the way independent variables interact with each other through multivariate analysis and hyperparameter tuning.

Feature retention and value

- Despite the overwhelming differences of missing values in the **age_band** and **gender** features, minor insights into the interactions that these classification features have were extracted. As the **Target** variable is unbalanced and the results reflect a minority of the total count, this preliminary exploration should not be published to the business, as these visuals are not accurate enough to represent the total population.

b. Suggestions / Recommendations

Future experiments should explore feature interactions in conjunction with others through **multivariate classification**, in addition to exercising applications through tweaking hyperparameters. The following offers more complicated methods of manipulating and transforming data for meaningful interpretation, as presently, the data has only been explored with a simple model. Future experiments may also benefit from further **feature engineering**, offering a means to transform data into more interpretable relationship modelling. As feature engineering can also reduce the dimensionality of the data and remove redundant aspects, it may lead to more promising precision and recall performance metric outcomes.

Support Vector Classification (SVC)

This classification algorithm may prove useful in future experiments, as it finds a hyperplane to maximally separate two classes, offering kernel functions like linear, polynomial, or radial options to better separate these classes.

Pros:

- Provides an effective means of separating non-linear datasets.
- Handles “high-dimensional” data.
- Regularisation functionality to prevent overfitting.

Cons:

- Slow for large datasets and computationally taxing.
- Sensitive to choice of kernel function and hyperparameters.

Random Forest Classification

This classification algorithm may also provide value in future experiments by predicting classes of a new observation based on a highly adjustable set of input features.

Pros:

- Offers functionality to handle outliers (not applicable) and missing data.
- Lower susceptibility to overfitting through bagging.
- Provides modelling functionality to a non-linear set of data.

Cons:

- Requires a large set of trees to achieve optimal performance, which results in computationally intensive processing requirements.
- As the existing dataset is imbalanced, there is a poorer performance probability.

XGBoost Classification

The XGB classification algorithm is a popular algorithm that combines multiple weak prediction models, like decision trees, iteratively adding trees and correcting errors of previous trees to generate an accurate prediction.

Pros:

- Efficient processing times.
- Capable of handling **imbalanced** and missing data.
- Scalable for larger datasets.

Cons:

- Typically requires more data preparation and parameter tuning than other algorithms.
- Typically doesn’t perform well with too many categorical features.
- Prone to overfitting.

- XGBoost is an optimised gradient-boosting algorithm that builds an ensemble of weak models to make accurate predictions.
- It can handle missing data and has excellent predictive power for high-dimensional data.
- However, XGBoost may require more data preparation and parameter tuning than other algorithms and can be more prone to overfitting.

Deployment

As the experiment did not achieve the required outcome for the business, deployment into production is not recommended at this current time.