EXPERIMENT REPORT

Student Name	Shaqran Bin Saleh
Project Name	Classification Models
Date	26/4/2024
Deliverables	Notebook Name : 36106_AT2_25010238_experiment_3.ipynb model name : Decision Tree Classifier

1. EXPERIMENT BACKGROUND	
Provide information about the problem/project such as the scope, the overall objective, expectations. Lay down the goal of this experiment and what are the insights, answers you want to gain or level of performance you are expecting to reach.	
1.a. Business Objective	The goal of this project is to build a predictive model with accurate likeliness of repurchasing a car for customers. The model's predictions can be used in a myriad of ways. The model can be used to segment customers by repurchase likelihood, focusing marketing efforts on high-potential features of the dataset for efficient resource use. Insights from the data and trained model can also enhance customer retention strategies and optimize inventory management. Additionally, the results would provide feedback to refine sales tactics and customer interactions, continually improving approaches to meet customer needs effectively. Accurate predictions would mean enhance efficiency in targeting the right customers in reducing marketing efforts and higher sales. Through proper data exploration and trained model insights proper understanding of customer needs and help improve customer satisfaction. Inaccurate repurchase likelihood prediction by the model might result in diminished business opportunities, reduced ROI, wasted marketing budget, and unhappy clients. Customers may become disconnected from a brand due to improper targeting, which could be perceived as spam and harm the company's reputation.
1.b. Hypothesis	I wish to test the hypothesis that several characteristics: Gender, The model of vehicle, The type of vehicle, Age of their last vehicle, Number of scheduled services used under warranty, Number of non-scheduled services used under warranty, Amount paid for scheduled services, Amount paid for nonscheduled services, Amount paid in total for services, Total number of services, The number of months since the last service, Annualized vehicle mileage, Number of different dealers visited for servicing, Number of services had at the same dealer where the vehicle was purchased have significant effects on the customers' likeliness of repurchasing a vehicle. I like to get the answers of few questions like "What are the primary repurchasing variables for customers and how do they interact to influence their repurchasing likelihood?" By testing this hypothesis and answering the accompanying question, we can acquire a deeper understanding of the factors that influence customers' likelihood of repurchasing.

1.c. Experiment Objective	From this experiment we will be deploying a predictive model using Decision Tree Classifier model that accurately handles the data and make accurate predictions. The goal is to identifying key variable effecting the customers' likelihood of repurchasing, developing an accurate model, apply it on the dataset and find the accuracy of the model basing upon the different hyperparameter tuning.

2. EXPERIMENT DETAILS

Elaborate on the approach taken for this experiment. List the different steps/techniques used and explain the rationale for choosing them.

2.a. Data Preparation

For proper data preparation, the following steps were taken:

1. The assigned .csv file was loaded into the notebook as a dataframe for the experiment.
2. The dimensions (rows, columns) of the data loaded from the .csv file was found. This step was performed so that we can properly understand the amount of data we are working with. It also gives us information about the number of features present in the data.
3. We have 17 features among which 1 is our target variable which is “Target”. We explored all the features of the dataset.
4. We check all the features if there are any null values and handle the situation.
5. We check the statistics (count, mean, standard deviation, minimum values, maximum value) of the features.
6. We drop the rows containing missing values from the dataset.
7. We drop the rows containing duplicate data.
8. Since we have 4 features containing categorical data. We introduced OneHotEncoder through which we transform the data and concatenate the new columns into our dataset. And finally remove the previous columns on which the transformation was done.
9. The feature “Target” is our target variable so we separate it as our ‘y’ variable and the rest were are considering our independent variable which
10. Then we split the data to three categories. Which are training data, testing data and validation data.
11. We distribute the data firstly into two segments Train + Validation and Test sets. The training + validation set would hold 80% of the data and rest 20% is testing data.
12. Then we again split the Train + Validation to separate Training and Validation sets. The validation would hold 30% of the Train + Validation data, meaning it hold 24% of the original data. The rest of the 56% of the original data is reserved for the training dataset.
13. And finally, we performed feature scaling

All the above steps were vital in this experiment is also vital for the next experiments.

2.b. Feature Engineering

While performing data preparation, we found out that the features : 'age_band', 'gender', 'car_model', 'car_segment' are categorical and are all important in the influence of the target variable. So, we had to encode these features. We achieved this encoding using the OneHotEncoder()

As we concatenated the new columns, we made sure to drop the columns on which the transformation was performed.

2.c. Modelling

For the experiment to predict customer repurchase likelihood, the Decision Tree Classifier model was chosen.

Decision Tree Classifier is straightforward to implement and understand, it works effectively in high-dimensional spaces.

- One of the most significant advantages of decision trees is their high interpretability. They are very intuitive and can be easily visualized, which helps in understanding how decisions are made by the model—making them a preferred choice for business-level presentations and decisions.
- Decision trees can handle both numerical and categorical data and are capable of modeling complex, non-linear relationships that other linear predictors are not capable of.
- Decision trees do not require normalization or scaling of data since the nodes split data by sorting values along individual features and splitting them along distinct points. This reduces the preprocessing steps and complexities.
- Without proper tuning, decision trees can easily overfit, especially with very noisy data.

The decision to use Decision Tree Classifier should be aligned with the specific characteristics of the data and the business or research objectives. Its powerful capabilities in handling complex, high-dimensional, and imbalanced data make it a robust choice for many classification challenges.

While performing the experiment hyperparameters tuning was done quite some time. The primary hyperparameter which is min_samples_split was tuned. And the following values were tested: 5, 10, 20, 30.

The hyperparameter min_samples_split significantly impacts by specifying the minimum number of samples a node must have before it can be split into two or more sub-nodes. This parameter is a way to control the depth and complexity of the decision tree and is crucial for preventing the model from overfitting.
For future we plan on exploring Random Forest and also Extra Trees.

14. EXPERIMENT RESULTS

Analyse in detail the results achieved from this experiment from a technical and business perspective. Not only report performance metrics results but also any interpretation on model features, incorrect results, risks identified.

3.a. Technical Performance

The performance metrics used in this experiment are :
1. Accuracy Score
2. Precision Score
2. F1 Score
3. Confusion Matrix

We used the DecisionTreeClassifier model and trained it multiple times as we performed multiple Hyperparameter tuning.

The results of the metrics according to different hyperparameters are as follow:

1. For default hyperparameters:
   - Accuracy Scores:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - F1 Scores:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - Precision Score:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - Confusion Matrix:
     ```
     [[3619    0]
      [  0   43]]
     ```

2. For Hyperparameter min_samples_split =5:
   - Accuracy Scores:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - F1 Scores:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - Precision Score:
     a. On Training set: 1.0
     b. On Validation set: 1.0
     c. On Testing set: 1.0
   - Confusion Matrix:
     ```
     [[3619    0]
      [  0   43]]
     ```

3. For Hyperparameters min_samples_split = 10:

- Accuracy Scores:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- F1 Scores:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- Precision Score:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- Confusion Matrix:

```
[[3619    0]
 [  0   43]]
```

4. For Hyperparameters min_samples_split =20:

- Accuracy Scores:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- Precision Score:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- F1 Scores:
  a. On Training set: 1.0
  b. On Validation set: 1.0
  c. On Testing set: 1.0
- Confusion Matrix:

```
[[3619    0]
 [  0   43]]
```

5. For Hyperparameters min_samples_split =30:

- Accuracy Scores:
  d. On Training set: 1.0
  e. On Validation set: 1.0
  f. On Testing set: 1.0
- Precision Score:
  d. On Training set: 1.0
  e. On Validation set: 1.0
  f. On Testing set: 1.0
- F1 Scores:
  d. On Training set: 1.0
  e. On Validation set: 1.0
  f. On Testing set: 1.0
- Confusion Matrix:

```
[[3619    0]
 [  0   43]]
```

	Main Issues and Causes: • For the default hyperparameters and all experimented min_samples_split values, all performance metrics (Accuracy, F1 Score, Precision) are 1.0 across training, validation, and testing sets. Additionally, the confusion matrices show no false positives or false negatives. This suggests a likely overfitting scenario where the model has perfectly learned the training data, including noise and outliers, thus not generalizing well. • Perfect scores in multiple configurations across different splits of the data are unusual and raise concerns about data leakage or very homogeneous data. • Increasing min_samples_split generally increases the bias of the model (making it simpler), but here it seems to reveal underlying issues with how the model handles class imbalance or minority classes.
3.b. Business Impact	According to our objective which we set earlier which was to deploy a predictive Decision Tree Classifier model that performs well. We can come the following conclusions about the results of this experiment: • For most configurations, the model achieved perfect scores (1.0) for accuracy, precision, and F1 score across the training, validation, and test sets. This suggests that the model is highly effective at classifying the given data under these configurations. • Despite changes in hyperparameters, the confusion matrix remains consistent with no false positives or negatives, which initially suggests exceptional model performance. And the impacts the business might face due to imperfect results would be: • If the model is deployed in this critical decision-making business, consistent inaccuracies or a failure to perform as expected could lead to loss of trust among stakeholders. Because even though perfect scores across all datasets might be initially impressive but are suspicious and could suggest overfitting or data leakage. • Missed opportunities due to incorrect model predictions can be significant. For instance, in a marketing campaign of this business, failing to correctly identify potential repurchasing customers (false negatives) could mean missed revenue opportunities.
3.c. Encountered Issues	As we performed this experiment, we were encountered with the following issues: • Our experiment identified a potential case of overfitting despite achieving perfect scores on the test set. While perfect test set performance is typically positive, the uniformity of these perfect scores across all metrics and datasets raises concerns.. • An increase in the minimum samples required per split (min_samples_split) from 5 to 30 did not result in any significant changes to the model's performance metrics. This observation suggests two possibilities: either the model is inherently insensitive to this parameter due to the characteristics of the data, or the data itself does not necessitate intricate decision boundaries that would be influenced by this parameter.

1. FUTURE EXPERIMENT

Reflect on the experiment and highlight the key information/insights you gained from it that are valuable for the overall project objectives from a technical and business perspective.

4.a. Key Learning

Reflecting on the outcomes and performance of the Decision Tree model experiments for predicting customer repurchase likelihood, several key insights emerge.

- Our observation of high performance across the training, validation, and testing sets warrants further investigation into potential data leakage or overfitting. This highlights the critical need for robust data integrity and meticulous data handling practices. These measures are essential to mitigate unintentional biases and inaccuracies that could lead to excessively positive model performance.
- The current experiment design necessitates investigation into the data's complexity and diversity. Limited data complexity or diversity may inadequately challenge the model, resulting in inflated performance metrics. This underscores the potential need for further experiments utilizing more intricate or noisy datasets to comprehensively evaluate the model's robustness.
- The model's consistent production of perfect results suggests a potential over-generalization issue. This could be problematic when applied to real-world data, which is typically more diverse and exhibits less predictability.
- The findings suggest a need to explore other machine learning models. This could help in finding a model that might be more sensitive to underlying patterns.

Given the insights and the issues encountered, here are the reasons to continue experimenting with the Decision Tree Classifier:

- Decision Trees are excellent at handling feature interactions naturally. Further exploration might reveal more about how different features interact to affect predictions.
- While this experiment showed insensitivity to the min_samples_split hyperparameter, experimenting with other hyperparameters like max_depth, min_samples_leaf, and max_features might provide more understanding of how the model reacts to changes and how these hyperparamters affect its performance.
- Decision Trees can serve as a good baseline model and can serve as a benchmark to models like Random Forests.

4.b. Suggestions / Recommendations

The experimentation with the Decision Tree Classifier for predicting customer repurchase likelihood identified several pathways for further improvement. These refinements can be categorized into two main phases: model optimization and deployment preparation.

1 Model Optimization:

- Experiment with increasing or decreasing the complexity of the Decision Tree. This could involve adjusting parameters such as max_depth, min_samples_leaf, and max_features to see how they affect model performance..

- The performance of the decision tree model should be evaluated alongside alternative models, such as random forests. This comparative analysis will provide valuable insights into the dataset's characteristics and guide the selection of the most appropriate modeling technique.

2 Deployment Preparation:

Following the optimization phase, a series of steps will ensure a successful deployment of the chosen model:

1. Once all testing and validation efforts are complete, the model configuration demonstrating the best performance will be selected for deployment.

2. A comprehensive plan outlining the integration of the chosen model into the existing IT infrastructure should be formulated.

3. A monitoring system should be established to track the model's performance in a production environment using real-world data. This will be accompanied by a maintenance schedule for periodically updating the model to maintain its effectiveness.

4. Training materials and comprehensive documentation should be developed to educate end-users on the model's functionality, its intended use within business processes, and its limitations.

5. The deployment process should adhere to all relevant compliance regulations and data privacy rules, ensuring the integrity and security of customer data.

By prioritizing model optimization followed by a well-defined deployment plan, we aim to maximize the predictive power of the customer repurchase prediction model while guaranteeing a smooth and effective transition to real-world applications.