# **EXPERIMENT REPORT**

| **Student Name** | Rusan Vaidya |
| --- | --- |
| **Project Name** | Performing RandomForest to predict if the customer purchases new car or not |
| **Date** | 27-04-2024 |
| **Deliverables** | 36106-AT2-24886400-experiment-5.ipynb  RandomForestClassifier, DecisionTreeClassifier  Precision, Recall, Imbalance Data, max\_depth, ColumnTransformer, RandomizedSearchCV |

| 1. **EXPERIMENT BACKGROUND** | | |
| --- | --- | --- |
| This project aims to forecast car repurchases by customers using RandomForestClassifier, contrasting its accuracy with the DecisionTreeClassifier model, which achieves 99% accuracy and recall of 0.81. The objective is to check if the RandomForestClassifier can surpass the DecisionTreeClassifier’s performance. | | |
| **1.a. Business Objective** | This project focuses on the problems from experiment 4 and looks at whether the RandomForestClassifier can better identify customers likely to repurchase cars, using fewer resources than the DecisionTreeClassifier. It also plans to check how reducing the waste of resources affects business results in car sales. | |
| **1.b. Hypothesis** | **Hypothesis:** The RandomForestClassifier is expected to outperform the DecisionTreeClassifier in predicting car repurchases.  **Explaination:** While the DecisionTreeClassifier boasts a 99% accuracy rate and a recall of 0.81 for class 1, there are still instances of incorrect true negative predictions that can be improved upon. | |
| **1.c. Experiment Objective** | The objective of this experiment is to refine prediction accuracy by evaluating if the RandomForestClassifier can exceed the performance of the DecisionTreeClassifier, remedying the issues noted in experiment 4. Although the DecisionTreeClassifier demonstrates robust accuracy and a more even prediction distribution, its reduced recall has resulted in errors, prompting an investigation into the RandomForest’s ability to enhance predictive outcomes. | |

| 1. **EXPERIMENT DETAILS** | | |
| --- | --- | --- |
| The experiment uses different data preprocessing tools to drop the columns and fill values to deal with the missing data from the dataset and also use tools such as OneHotEncoding to encode the categorical data used to predict the customer repurchasing behavior. | | |
| **2.a. Data Preparation** | 1. In this experiment, the `***age\_band***`and `***gender***` column from the dataset was filled with “Not Specified”. 2. Dropped the identifier ***`ID`*** from the dataset. 3. Dropped all the duplicate values. 4. For the column `***car\_model***`, and ***`car\_segment`***, it is encoded using OneHotEncoding. 5. The features are scaled using the StandardScaler | |
| **2.b. Feature Engineering** | 1. Over 85% of the entries in the ***`age\_band`*** column and over 50% of the entries in the ***`gender`*** column are missing from the dataset. These missing values have been filled with the term "Not Specified" to analyze whether this particular categorization leads to improved performance in the model. 2. OneHotEncoding of ***`car\_model`*** and ***`car\_segment`*** as its essential step as it is not an ordinal category. 3. Scaling is performed in order to maintain scaling across features. 4. Standardization and Encoding was performed using sklearn instead of an imblearn tool named Pipeline. 5. **The dataset used for this experiment is imbalanced.** However, for the DecisionTreeClassifier model, this doesn't matter as much because it looks at trees to make predictions, so the imbalance doesn't affect it much. | |
| **2.c. Modelling** | The RandomForestClassifier employs an ensemble of decision trees to model complex relationships between features and target labels, capturing non-linear patterns effectively. For this experiment, the hyperparameters were dynamically optimized using RandomizedSearchCV with stratified k-folds to enhance model accuracy and prevent overfitting. The specific parameters tested included:   1. **max\_depth**: ranging from 1 to 20, to control the depth of the trees. 2. **min\_samples\_split**: ranging from 1 to 200, to specify the minimum number of samples required to split an internal node. 3. **min\_samples\_leaf**: ranging from 1 to 100, to define the minimum number of samples required at a leaf node. 4. **max\_features**: set to 'sqrt', 'log2', or None, determining the number of features to consider when looking for the best split. 5. **n\_estimators**: between 5 and 200, setting the number of trees in the forest. 6. **max\_leaf\_nodes**: from 5 to 200, limiting the number of leaf nodes in the tree.   The best model parameters were identified through the RandomizedSearchCV process and applied to the final model, which was then used to predict and evaluate its performance on the unseen data. This approach aimed at optimizing the model's generalization capabilities and improving predictive accuracy. | |

| 1. **EXPERIMENT RESULTS** | | |
| --- | --- | --- |
| In terms of technical evaluation, the RandomForestClassifer model's effectiveness is evaluated through metrics like accuracy, precision, recall, and ROC AUC. Meanwhile, from a business standpoint, it is checked how different features impact car repurchase decisions. | | |
| **3.a. Technical Performance** | With max\_depth=10, precision and recall improved to 0.94 and 0.81 for class 1 which is minority class respectively, indicating balanced performance with high accuracy of 99%.  For RandomForest, the best model prediction has accuracy of 98% with higher precision but lower recall of 0.43 giving less accurate prediction for minority class. | |
| **3.b. Business Impact** | RandomForestClassifier model has lower recall which means it predicts less number of actual customers who would buy a second car as not a customer which leads to more waste of resources.  The DecisionTreeClassifier model is better and has a balanced prediction which is better in both major and minority class and predicts Customers who would “Repurchase” and customers who would “Not Repurchase”. | |
| **3.c. Encountered Issues** | **Issues:**   1. The model tends to overfit 2. Balancing recall and precision.   **Solution:**   1. Using a randomizedsearchCV to find best fit of model 2. Analyzing through AveragePrecision and ROC curves and visualization. | |

| 1. **FUTURE EXPERIMENT** | | |
| --- | --- | --- |
| This experiment handled the imbalance dataset and predicting customer repurchasing behavior and emphasizing evaluation and business applicability for improved decision making. | | |
| **4.a. Key Learning** | The experiment demonstrated that the SVC model excels in accurately predicting a higher number of customers likely to repurchase cars. This capability positions the SVC as a robust tool for enhancing decision-making in scenarios involving car repurchases. Meanwhile, the DecisionTreeClassifier showed strength in delivering balanced predictions with fewer errors, capturing a substantial number of actual customers, although slightly fewer than the SVC model. Further refinement of both models could lead to improved performance and more effective decision-making support in automotive sales contexts. | |
| **4.b. Suggestions / Recommendations** | 1. Further tuning for SVC if it could achieve higher precision for minority class. 2. Initiate deployment by integrating the model into the existing IT infrastructure, setting up continuous monitoring for performance. | |