|  |  |
| --- | --- |
| **DATA 430 Technical Report Assignment 3: Decision Trees** | **Matt Schnorr** |
| **Telecommunications Data – Decision Tree Model for Customer Classification** | |
| **URL to dataset: https://www.kaggle.com/datasets/prathamtripathi/customersegmentation** | |

This template should be used in conjunction with the assignment instructions. The size of the text area below will expand to the length of your response; the area should not be interpreted as a required or suggested length of response. Responses within the text area should be single spaced with Times New Roman 12pt font. The body of the document will likely be 6-9 pages, not including the Appendix; length may vary depending on specifics of the analysis and the dataset. As needed, APA format in-text citations should be included, along with a full references list at the end of the document.

|  |
| --- |
| **Overview** |
| **Problem Domain**: give some background and context about the problem domain (application area). For instance, if you are doing the analysis for predicting heart disease, provide some context about the disease and include some interesting statistics about it. Also, discuss how the method is relevant for the chosen problem. |
| The ‘Telecust’ dataset involves customer segmentation for a telecommunications company. Customer segmentation is crucial for businesses to understand and cater to the diverse needs of their clientele. By analyzing customer data, companies can identify distinct groups based on various attributes such as age, income, tenure, and usage patterns. This analysis helps in designing targeted marketing strategies, improving customer satisfaction, and enhancing service delivery. The relevance of using a decision tree classifier in this context lies in its ability to handle complex, multi-dimensional data and provide clear, interpretable rules for customer classification. |
| **Objective**: clearly state the objective of the analysis in relation to the kind of algorithm you are employing. Use specific language as to what question(s) you are trying to answer using the specific analysis/modeling type. |
| The objective of this analysis is to build a decision tree classifier to predict the customer category (‘custcat’) based on demographic and service-related features. The specific question I aim to answer is: "Which features are most important in determining the customer category, and how accurately can we predict a customer's category using these features?" |
| **Analysis** |
| **Exploratory Analysis**: describe the data including the source, the collection method, and variables. Perform exploratory analysis. Also, select few key variables (including the target variable for supervised learning) and study their distributions using plots such as histograms, box plot, bar chart, etc. |
| **Exploratory Analysis:** The Telecust1 dataset comprises 1000 entries with 12 columns.  The variables are: region, tenure, age, income, marital, address, ed, employ, retire, reside, gender, and custcat.  The data source is a telecommunications company, and the data was likely collected through customer surveys and service records.  Key variables analyzed:   * **Age**: Distribution of age across different customer categories was visualized using histograms. * **Income**: The relationship between income and customer category was explored using box plots. * **Tenure**: Histograms were used to understand the distribution of tenure.   (See Figure 1-6) |
| **Preprocessing**: armed with the exploratory analysis, perform the necessary preprocessing, both general and specific types appropriate for the modeling type being employed. |
| \* **Handling Missing Values**: No missing values were found in the dataset.  **\* Encoding Categorical Variables**: Categorical variables (region, marital, ed, retire, gender, and custcat) were encoded to numerical values using label encoding.  \* **Splitting Data**: The data was split into features (X) and target variable (y). Further, the data was split into training (80%) and testing (20%) sets using train\_test\_split. |
| **Model Fitting**: explain the key steps and activities you perform to fit the model. Experiment (as appropriate) with parameters tuning. This is key, what separates highly accurate model from a less accurate ones is the amount of performance tuning performed. |
| \* A Decision Tree Classifier was used for model fitting. The model was trained using the training data (X\_train, y\_train).  \* Default hyperparameters were used initially. The model was fitted, and predictions were made on the test data (X\_test).  \* Hyperparameter tuning can be performed using techniques like GridSearchCV for better accuracy, but was not explored in this initial analysis. |
| **Results** |
| **Model Properties:** explain the components of the fitted model and their characteristics. Leverage functions to summarize the model properties. Also, leverage visualization as required. |
| * The decision tree model was visualized to show its structure. It included nodes representing feature splits and leaves representing customer categories. * The tree's depth and number of leaves indicate the model's complexity. Functions like plot\_tree were used for visualization.   (See Figure 8) |
| **Output Interpretation**: explain the result and interpret the final model output using terms that reflect the application area and in relation to the stated objective. This is where you check whether or not the stated objective is met. |
| I was able to create a decision tree, with some accuracy. However, the accuracy was not as high as I would recommend. My initial objective of building a model and determining which factors were most impactful was able to be accomplished.  **Accuracy Score**: The accuracy score of the model on the test set was approximately 0.33 (33%). This means that 33% of the predictions made by the model were correct.  **Confusion Matrix**: The confusion matrix provided a detailed breakdown of the model's performance. Each cell in the matrix represented the number of predictions made by the model that fall into specific actual and predicted class combinations:   * True Positives (TP): Correctly predicted customer categories. * False Positives (FP): Incorrectly predicted customer categories (model predicted a category different from the actual). * True Negatives (TN): Correctly identified non-target categories. * False Negatives (FN): Incorrectly identified non-target categories (model missed the actual category).   **Classification Report**: The classification report included precision, recall, and F1-score for each customer category:   * **Precision**: The proportion of correct positive predictions (True Positives) out of all positive predictions (TP / (TP + FP)). High precision indicates fewer false positives. * **Recall**: The proportion of actual positives that were correctly predicted (TP / (TP + FN)). High recall indicates fewer false negatives. * **F1-Score**: The harmonic mean of precision and recall, providing a single metric to evaluate the model's performance. |
| **Evaluation**: employ appropriate metrics to quantitatively evaluate the performance of the fitted model. For supervised classification, this includes simple accuracy, precision & recall (or sensitivity & specificity), all of which can be generated from a confusion matrix, or ROC. |
| * **Confusion Matrix**:   + The confusion matrix showed that the model had the highest correct predictions (True Positives) for categories 1 and 3.   + There were some misclassifications between categories 2 and 4, indicating room for improvement in distinguishing these classes. * **Classification Report**:   + **Precision** ranged from 0.25 to 0.44, indicating how often the model was correct when it predicted a certain category.   + **Recall** ranged from 0.27 to 0.37, indicating how well the model identified all actual instances of each category.   + **F1-Score** provided a balance between precision and recall, indicating the overall effectiveness of the model for each category. * **Accuracy**:   + The overall accuracy of 33% indicated that the model was correct in its predictions 33% of the time.   + This accuracy suggests a poor model, with room for major improvement, especially in distinguishing between certain categories. * **Feature Importance**:   + **Tenure**, **age**, and **income** were the most influential features, indicating that how long a customer has been with the company, their age, and their income significantly affect their customer category.   + **Employ** and **address** also had notable importance, but less than the top three features.   + **Gender** and **retire** had the least impact on the model’s predictions.   **Visualization**:   * The decision tree plot visualized how the model made splits based on the features and how it arrived at its final predictions for customer categories. * The feature importance bar plot provided a clear visual representation of the relative importance of each feature in the model.   (See Figure 7 and Figure 10) |
| **Conclusion** |
| **Summary**: highlight the main findings in relation to the stated objective. You don’t need to discuss the details of the analysis and the model such as accuracy here, just focus on the key findings. |
| **Summary:**   * The decision tree classifier successfully classified customer categories with poor accuracy (33%). * Key findings include the importance of features like tenure, age, and income in predicting customer categories. * The model's interpretability provides clear rules for customer segmentation, helping the telecommunications company understand and predict customer behavior better. |
| **Limitations & Improvement areas**: discuss the limitations of the analysis and identify potential improvement areas for future work. This could be related to the data, algorithm, or a combination of the two. |
| * **Data Limitations**: The dataset might not capture all relevant features influencing customer behavior. Additional data on customer preferences and service usage could improve the model. * **Algorithm Limitations**: Decision trees are prone to overfitting. Ensemble methods like Random Forests or boosting techniques could provide better performance. * **Hyperparameter Tuning**: Further tuning of model parameters could enhance accuracy and robustness. * **Class Imbalance**: Addressing any class imbalances in the dataset could improve model performance for underrepresented categories. |

|  |
| --- |
| **Appendix** |
| Figure 1-6:              Figure 7:    Figure 8:    Figure 9:    Figure 10: |