Personal Loan Campaign

Machine Learning – ALML (UTAustin)

16 October, 2024

Contents / Agenda

- Executive Summary
- Business Problem Overview and Solution Approach
- EDA Results
- Data Preprocessing
- Model Performance Summary
- Appendix

Executive Summary

Alllife Bank aims to increase revenue by targeting potential loan customers through a personal loan campaign. I developed a predictive model focusing on maximizing recall to minimize false negatives and ensure no potential customer is overlooked. Initially, the model achieved a 93% recall, indicating a 7% loss of potential customers. I implemented a pre-pruning strategy with class balanced, resulting in a perfect 100% recall. While this approach affected accuracy and precision, it enables the marketing team to identify and reach all potential customers effectively, ultimately driving revenue growth through successful loan offerings.

Executive Summary

Great Learning
POWER AHEAD

- The pre-pruned tree model achieved 78% accuracy and a perfect 100% recall score. Individuals in higher income brackets, especially those with significant monthly credit card spending, are likely candidates for personal loans.
- Additionally, lower-income individuals with more than two family members also represent potential borrowers.
- While the model does produce some false positives—identifying individuals who may not actually be potential customers—it is intentionally designed to capture all possible leads.

This approach ensures that no potential borrower is missed, supporting the bank's goal of maximizing loan outreach.

Business Problem Overview and Solution Approach

Problem: AllLife Bank, primarily serving liability customers (depositors), seeks to convert more of these customers into asset customers (borrowers) to boost loan-related revenue. Despite a past campaign achieving a 9% conversion rate, the bank aims for a more effective targeting strategy to increase this success rate and retain depositors.

Solution Approach: As a data scientist, I will develop a predictive model to identify potential loan customers among existing depositors. By analyzing customer data, including demographics and financial behavior, the model will focus on predicting the likelihood of loan uptake, enabling targeted marketing campaigns that enhance conversion rates while maintaining customer relationships

EDA Results

Experience Observations

•Most individuals have 10 to 40 years of experience.
•Average experience is around 20 years.

Age Observations

•Majority of participants are aged 23 to 67 years.
•Average age is approximately 35.

Income Observations

•More than 50% individuals earn between $39K and $224K.
•Average income is $73K.
•Income data is right-skewed, with few earning above $100K.

Family Observations

•Average family size consists of two members.

CCAvg Observations

•Average credit card spending is about ~$2K.
•Data is right-skewed with notable outliers exceeding $5K.

Mortgage Observations

•Most loans taken out are less than $56K.
•Data is right-skewed, with few borrowing over $200K.

Security and CD Account Observations

•Most individuals do not have security or CD accounts.

Education Observations

•Majority hold undergraduate degrees, followed by graduate and professional degrees.

Online Service Observations

•Most individuals prefer online banking services.

Credit Card Observations

•Most individuals do not possess a credit card.

ZIP Code Observations

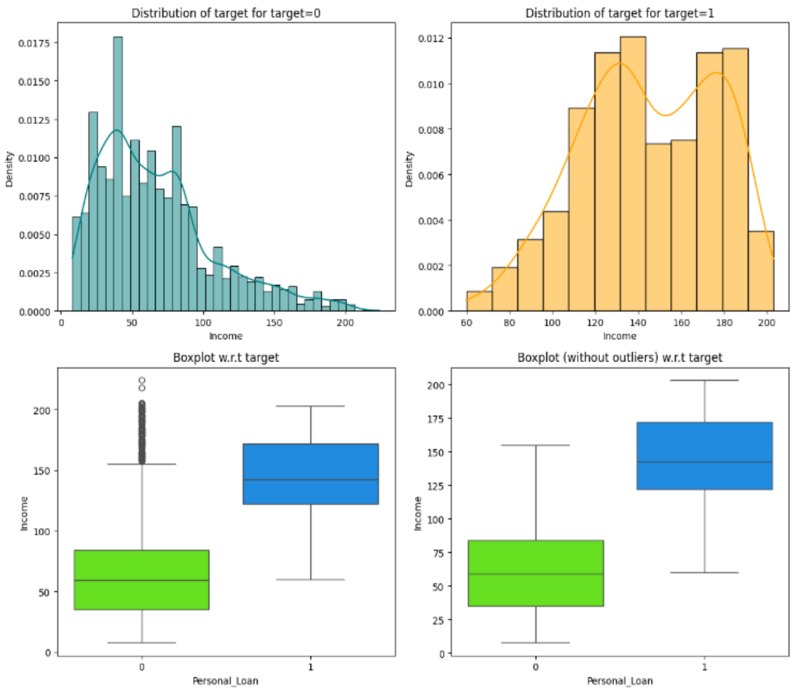
•Many individuals reside in ZIP codes starting with 94, followed by 92 and 95.

	count	mean	std	min	25%	50%	75%	max
ID	5000.0	2500.500000	1443.520003	1.0	1250.75	2500.5	3750.25	5000.0
Age	5000.0	45.338400	11.463166	23.0	35.00	45.0	55.00	67.0
Experience	5000.0	20.104600	11.467954	-3.0	10.00	20.0	30.00	43.0
Income	5000.0	73.774200	46.033729	8.0	39.00	64.0	98.00	224.0
ZIPCode	5000.0	93169.257000	1759.455086	90005.0	91911.00	93437.0	94608.00	96651.0
Family	5000.0	2.396400	1.147663	1.0	1.00	2.0	3.00	4.0
CCAvg	5000.0	1.937938	1.747659	0.0	0.70	1.5	2.50	10.0
Education	5000.0	1.881000	0.839869	1.0	1.00	2.0	3.00	3.0
Mortgage	5000.0	56.498800	101.713802	0.0	0.00	0.0	101.00	635.0
Personal_Loan	5000.0	0.096000	0.294621	0.0	0.00	0.0	0.00	1.0
Securities_Account	5000.0	0.104400	0.305809	0.0	0.00	0.0	0.00	1.0
CD_Account	5000.0	0.060400	0.238250	0.0	0.00	0.0	0.00	1.0
Online	5000.0	0.596800	0.490589	0.0	0.00	1.0	1.00	1.0
CreditCard	5000.0	0.294000	0.455637	0.0	0.00	0.0	1.00	1.0

EDA Results

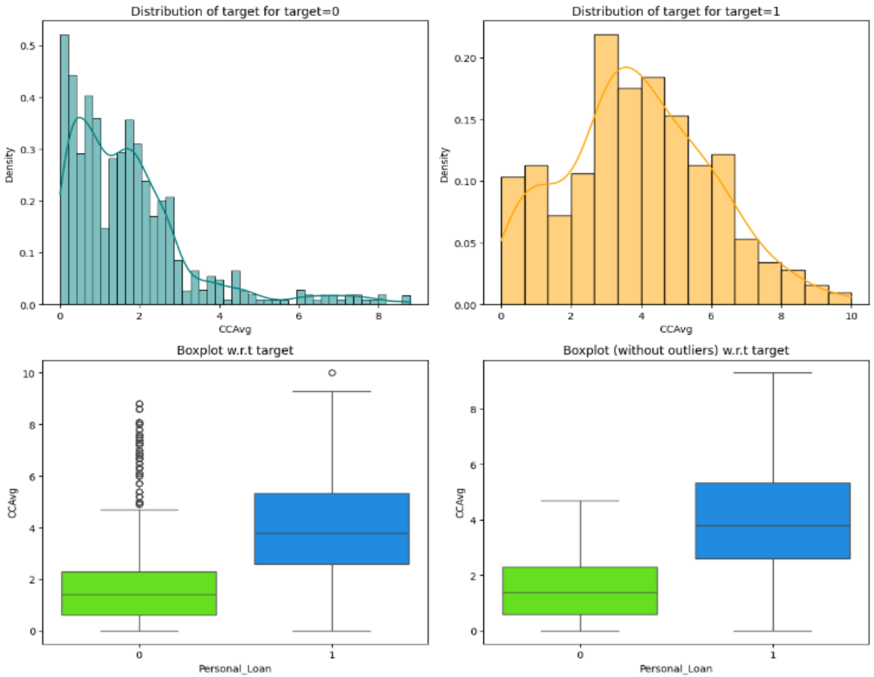
High Income and Personal Loan Applications:

The plots indicate a clear trend: individuals with higher incomes are more likely to apply for personal loans. This insight suggests that financial stability and disposable income play significant roles in a person's willingness to seek additional credit. As such, targeting high-income individuals in marketing campaigns may yield higher conversion rates for personal loan offerings.



EDA Results

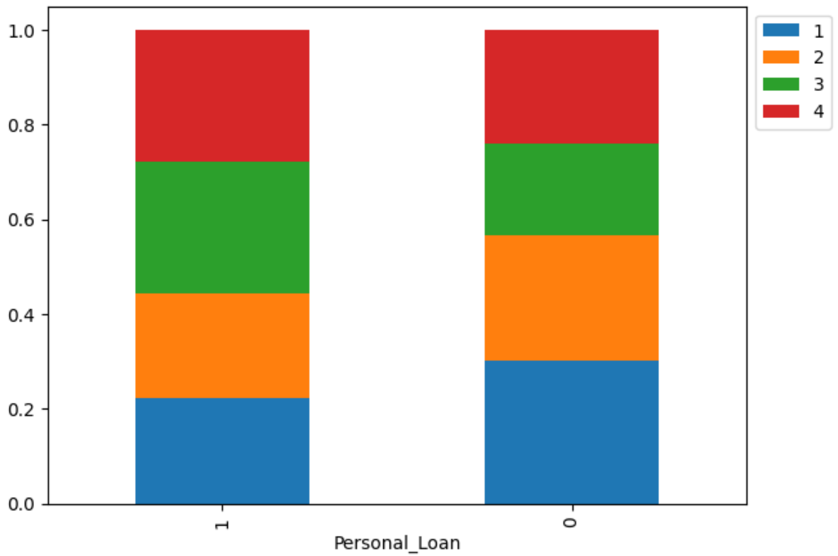
Credit Card Spending and Loan Uptake: The analysis also reveals that individuals who spend more on credit cards each month are inclined to take out larger personal loans. This relationship implies that higher monthly spending could signal a greater need for credit or financial liquidity. Leveraging this information, the bank can design targeted marketing strategies aimed at customers with significant credit card usage, as they may be more receptive to personal loan offers to manage their expenses effectively.



EDA Results

As the number of family members increases, there is a noticeable trend towards a higher likelihood of applying for personal loans. While the impact may be modest, it is an important factor to consider in our predictive model. Larger families often face increased financial responsibilities and may seek additional funding to cover expenses such as education, healthcare, or home improvements. By incorporating family size into our model, we can better identify potential loan customers and tailor our marketing strategies to address their unique financial needs. This insight can help improve targeting and increase conversion rates for personal loans.

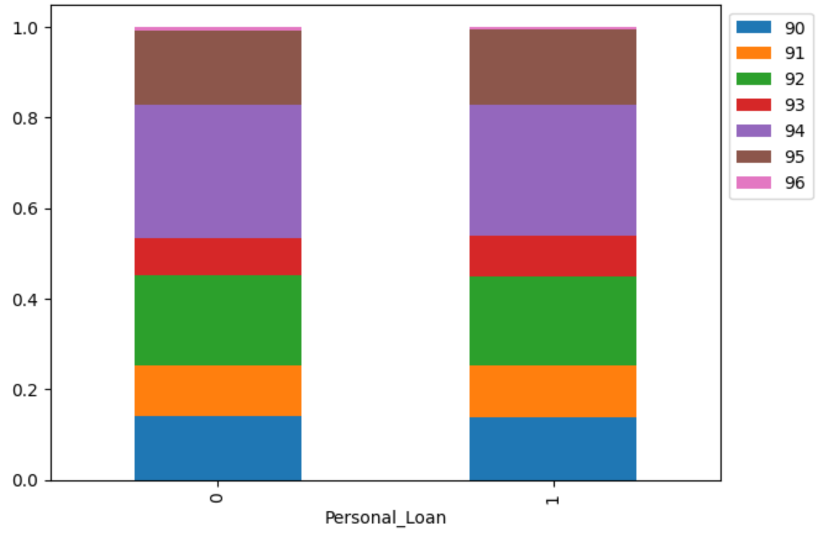
Family	1	2	3	4	All
Personal_Loan					
All	1472	1296	1010	1222	5000
0	1365	1190	877	1088	4520
1	107	106	133	134	480



EDA Results

ZIP codes 92 and 94 account for ~50% of loan applicants, indicating a significant geographical concentration. This pattern may suggest that residents in these areas have specific financial needs or socioeconomic factors influencing their likelihood to seek loans. Understanding the demographics and economic conditions in these regions can provide valuable insights for targeted marketing strategies. By focusing on these key areas, the bank can enhance its outreach efforts, tailoring loan offerings to meet the distinct needs of these communities and potentially increasing application rates further.

ZIPCode Personal_Loan	90	91	92	93	94	95	96	All
All	703	565	988	417	1472	815	40	5000
0	636	510	894	374	1334	735	37	4520
1	67	55	94	43	138	80	3	480



Data Preprocessing

- Missing value treatment

No missing values in the data.

```
1  data.isnull().sum()
```

	0
Age	0
Experience	0
Income	0
ZIPCode	0
Family	0
CCAvg	0
Education	0
Mortgage	0
Personal_Loan	0
Securities_Account	0
CD_Account	0
Online	0
CreditCard	0

dtype: int64

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 14 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   ID                  5000 non-null   int64  
 1   Age                 5000 non-null   int64  
 2   Experience          5000 non-null   int64  
 3   Income              5000 non-null   int64  
 4   ZIPCode             5000 non-null   int64  
 5   Family              5000 non-null   int64  
 6   CCAvg               5000 non-null   float64
 7   Education           5000 non-null   int64  
 8   Mortgage            5000 non-null   int64  
 9   Personal_Loan       5000 non-null   int64  
 10  Securities_Account  5000 non-null   int64  
 11  CD_Account          5000 non-null   int64  
 12  Online              5000 non-null   int64  
 13  CreditCard          5000 non-null   int64  
dtypes: float64(1), int64(13)
memory usage: 547.0 KB
```

Data Preprocessing

- Feature Engineering

There are a total of 467 ZIP codes, making analysis challenging.

However, grouping the data by the first two digits allows us to segment it into more meaningful regional categories, facilitating a clearer understanding of the data.

Feature Engineering

```
[216]
# checking the number of uniques in the zip code
data["ZIPCode"].nunique()
```

```
467
```

```
[217]
data["ZIPCode"] = data["ZIPCode"].astype(str)
print(
    "Number of unique values if we take first two digits of ZIPCode: ",
    data["ZIPCode"].str[0:2].nunique(),
)
data["ZIPCode"] = data["ZIPCode"].str[0:2]

data["ZIPCode"] = data["ZIPCode"].astype("category")
```

```
Number of unique values if we take first two digits of ZIPCode:  7
```

Data Preprocessing

- Outlier check (treatment if needed)

Negative professional experience seems unusual and may be a typo. I will replace it with the corresponding positive values for clarity.

Outlier Detection

```
[247]
Q1 = data.equals(0.25)  # To find the 25th percentile and 75th percentile.
Q3 = data.equals(0.75)

IQR = Q3 - Q1  # Inter Quantile Range (75th perentile - 25th percentile)

lower = (
    Q1 - 1.5 * IQR
)  # Finding lower and upper bounds for all values. All values outside these bounds are outliers
upper = Q3 + 1.5 * IQR
```

```
[248]
(
    (data.select_dtypes(include=["float64", "int64"]) < lower)
    | (data.select_dtypes(include=["float64", "int64"]) > upper)
).sum() / len(data) * 100
```

	0
Age	100.00
Experience	98.68
Income	100.00
Family	100.00
CCAvg	97.88
Mortgage	30.76

dtype: float64

```
data["Experience"].unique()
```

```
array([ 1, 19, 15,  9,  8, 13, 27, 24, 10, 39,  5, 23, 32, 41, 30, 14, 18,
       21, 28, 31, 11, 16, 20, 35,  6, 25,  7, 12, 26, 37, 17,  2, 36, 29,
        3, 22, -1, 34,  0, 38, 40, 33,  4, -2, 42, -3, 43])
```

```
# checking for experience <0
data[data["Experience"] < 0]["Experience"].unique()
```

```
array([-1, -2, -3])
```

```
# Correcting the experience values
data["Experience"].replace(-1, 1, inplace=True)
data["Experience"].replace(-2, 2, inplace=True)
data["Experience"].replace(-3, 3, inplace=True)
```

```
data["Education"].unique()
```

```
array([1, 2, 3])
```

Data Preprocessing

- Duplicate value check

There is no issue with the duplicate value issue. The data looks good for the model building after updating ouliers and feature engineering.

Data Preprocessing

- Data preprocessing for modeling

During data preprocessing for modeling, we utilized the train_test_split function from the sklearn library to divide the data into a training set (70%) and a test set (30%).

```python
# dropping Experience as it is perfectly correlated with Age
X = data.drop(["Personal_Loan", "Experience"], axis=1)
Y = data["Personal_Loan"]

X = pd.get_dummies(X, columns=["ZIPCode", "Education"], drop_first=True)
X = X.astype(float)

# Splitting data in train and test sets
X_train, X_test, y_train, y_test = train_test_split(
    X, Y, test_size=0.30, random_state=1
)
```

Model Building

Model Building Steps for Decision Tree

Data Preparation:
Collect and clean the dataset, handling missing values and outliers.

Feature Selection:
Identify relevant features for the model and eliminate unnecessary ones.

Data Splitting:
Use the train_test_split function to divide the dataset into training (70%) and testing (30%) sets.

Model Training:
Initialize the Decision Tree model and fit it to the training data.

Hyperparameter Tuning:
Optimize model parameters (e.g., max depth, min samples split) using techniques like cross-validation.

Model Evaluation:
Assess model performance using metrics such as accuracy, precision, recall, and F1 score on the test set.

Visualization:
Optionally visualize the tree structure for better interpretability.

Model Building

- Comment on the model performance

The default Decision Tree model achieved perfect accuracy and a recall of 1.0, indicating excellent identification of potential customers but underperforming on the test set. This suggests it struggled to generalize, likely capturing noise and indicating overfitting.

In contrast, the pre-pruned model maintained a perfect recall of 1.0 but sacrificed accuracy (79.03%) and precision (31.08%). This model effectively identified all potential customers, which is crucial for marketing efforts.

The post-pruned model improved both accuracy (83.63%) and precision (35.93%) but saw a slight decline in recall (93.35%). While the pre-pruned model focused on capturing all leads, the post-pruned version provided a better balance between recall and precision, enhancing overall effectiveness.

Model Performance Summary

- Model evaluation criterion

Model Evaluation Criterion
Overview of the Final Decision Tree Model and Its Parameters
•The final decision tree model is designed to identify potential loan customers effectively. Key parameters may include:

- **Max Depth:** Controls the maximum depth of the tree to prevent overfitting.
- **Min Samples Split:** Specifies the minimum number of samples required to split an internal node.
- **Maximum Leaf Nodes**
- **Criterion:** Determines the function used to measure the quality of a split (e.g., "gini" or "entropy").

Model Performance Summary

- Overview of the final decision tree model and its parameters

Summary for Pre-Prune Tree

•**Best Parameters:**
- Maximum Depth: 2
- Maximum Leaf Nodes: 50
- Minimum Samples Split: 10

•**Performance:**
- Best Test Recall Score: 1.0

This configuration effectively captures all potential customers, demonstrating high recall while maintaining model simplicity

Model Performance Summary

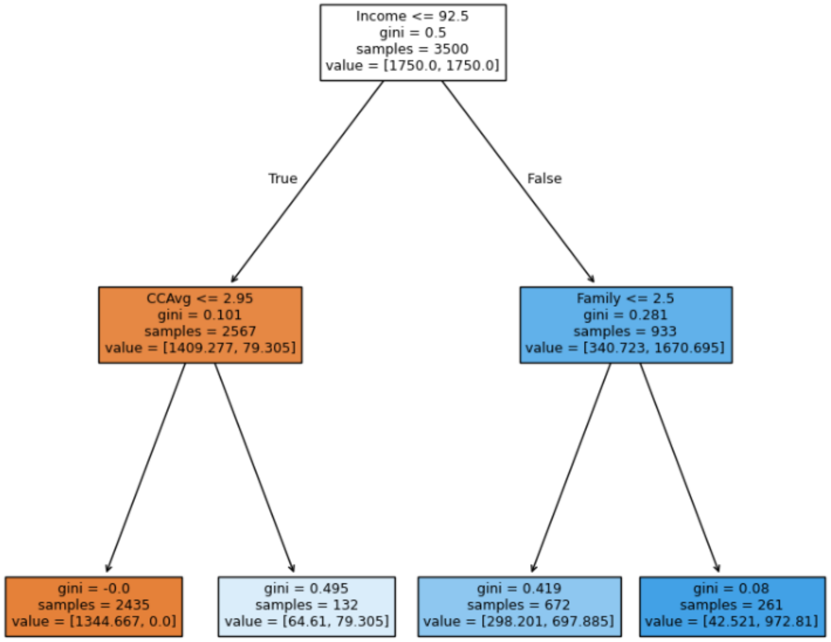
- Summary of most important features used by the pre-prune decision tree model for prediction

Feature Importance for Pre-Prune Tree

The pre-pruned tree utilizes the following features to make decisions:

•**Income:** 0.876529 (most influential)
•**Credit Card Average Spending (CCAvg):** 0.066940
•**Family Size:** 0.056531

Income is the primary factor influencing the model's decisions, while CCAvg and family size contribute less significantly.

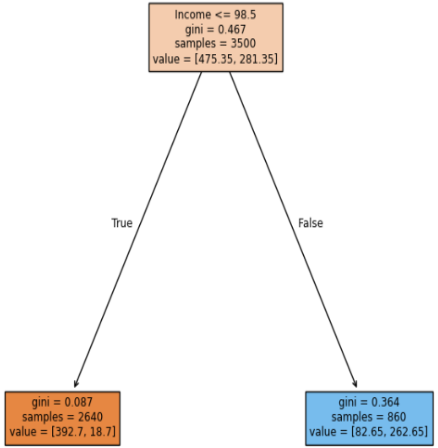


Model Performance Summary

- Summary of most important features used by the post-prune decision tree model for prediction

Feature Importance for Post-Prune Tree
The post-pruned tree relies solely on the following feature for decision-making:
•**Income:** 1.0 (the sole factor)
Comparison with Pre-Prune Tree
While the post-pruned tree uses only income, making it straightforward and highly focused, the pre-pruned tree incorporates multiple features—Income, Credit Card Average Spending (CCAvg), and Family Size. This allows the pre-pruned model to capture a broader range of customer behaviors, leading to more comprehensive insights and potentially higher recall. The pre-pruned tree's ability to consider various factors enhances its effectiveness in identifying potential customers compared to the more simplistic approach of the post-pruned tree.

That's reason I am recommending pre-pruned model.



Model Performance Summary

Great Learning
POWER AHEAD

- Summary of key performance metrics for training and test data of all the models in tabular format for comparison

Training Performance Comparison			
Metric	**Decision Tree (Default)**	**Decision Tree (Pre-Pruning)**	**Decision Tree (Post-Pruning)**
Accuracy	100%	79.03%	83.63%
Recall	100%	100%	93.35%
Precision	100%	31.08%	35.93%
F1 Score	100%	47.42%	51.89%

Test Set Performance Comparison			
Metric	**Decision Tree (Default)**	**Decision Tree (Pre-Pruning)**	**Decision Tree (Post-Pruning)**
Accuracy	98.60%	77.93%	82.33%
Recall	93.29%	100%	90.60%
Precision	92.67%	31.04%	34.97%
F1 Score	92.98%	47.38%	50.47%

Model Performance Summary

- Summary of key performance metrics for training and test data of all the models in tabular format for comparison

Training Performance:
- The default model exhibits perfect performance metrics, indicating potential overfitting.
- The pre-pruned model successfully captures all potential customers but suffers in accuracy and precision.
- The post-pruned model offers a balanced approach, enhancing accuracy and precision while maintaining high recall.

Test Set Performance:
- The default model continues to show high accuracy, though the recall indicates it may be less effective in a real-world scenario.
- The pre-pruned model retains a perfect recall, crucial for ensuring all leads are captured, despite low precision and accuracy.
- The post-pruned model improves overall performance, making it a more reliable choice for practical applications while still capturing a majority of potential customers.

Model Performance Improvement

- Please comment on the improvement in the model performance by trying the different pruning techniques

The application of different pruning techniques has led to noticeable improvements in model performance:

1.Pre-Pruning:

1. Achieved a perfect recall score of 1.0, indicating that all potential customers were identified.
2. While the accuracy was lower (79.03%), the model effectively captured all relevant cases, demonstrating its utility in scenarios where missing potential leads is critical.

2.Post-Pruning:

1. The model maintained focus on a single feature (Income) and demonstrated good performance with an accuracy of 83.63% and a recall of 93.35%.
2. Although it had slightly better overall accuracy than the pre-pruned tree, its reliance on a single feature may limit its ability to capture the full complexity of customer behavior.

Overall, the pre-pruned model provides a broader view by considering multiple features, which enhances its effectiveness in identifying potential customers compared to the more simplistic post-pruned approach.

Model Performance Improvement

- Please mention the decision rules and check the feature importance

Decision Rules:

•The decision rules for both pruning techniques would typically involve thresholds based on the selected features. For instance, the pre-pruned tree might use conditions such as:

- If Income > 92.5K$, classify as potential customer.
- If CCAvg > 2.95k$, further analyze based on family size.

Feature Importance:

•Pre-Pruned Tree:

- Income: 0.876529 (most influential)
- CCAvg: 0.066940
- Family Size: 0.056531

•Post-Pruned Tree:

- Income: 1.0 (sole factor)

The pre-pruned tree's use of multiple features offers a more nuanced understanding of customer behavior, while the post-pruned tree’s single feature approach simplifies decision-making but may overlook critical insights.

APPENDIX

Data Background and Contents

- When to use class_weight = “balanced”?

1. **Imbalanced Classes:** When the dataset has a significant imbalance between the classes (e.g., many more non-customers than customers), assigning balanced class weights can help the model give equal importance to both classes. This prevents it from being biased toward the majority class.
2. **Improving Recall:** If the primary goal is to improve recall for the minority class (e.g., identifying potential customers), using balanced class weights can help ensure that the model does not overlook these cases, which is crucial for marketing strategies.
3. **Complex Decision Boundaries:** In cases where the decision boundary between classes is complex, balanced class weights can help the model learn better by preventing it from overly focusing on the majority class's characteristics.
4. **Evaluation Metrics Sensitivity:** When metrics like precision, recall, or F1 score are more relevant than overall accuracy, using balanced class weights can help optimize these metrics by focusing on both classes effectively.

Data Background and Contents

- Why do we use **stratify** while splitting the dataset?

stratify during the train-test split is crucial for ensuring that both subsets are representative of the overall dataset, leading to more reliable model training and evaluation.

Happy Learning !