ExtraaLearn Project

Classification and Hypothesis Testing: MIT-IDSS Data Science & Machine Learning

Nov 25, 2024

Contents / Agenda

- Business Problem Overview and Solution Approach
- Data Overview
- EDA Results - Univariate and Multivariate
- Data Preprocessing
- Model Performance Summary
- Conclusion and Recommendations

Business Problem Overview and Solution Approach

Business Problem : The EdTech startup ExtraaLearn faces a challenge in efficiently allocating resources to convert leads into paid customers amidst rapid industry growth and increasing lead generation. With a high volume of leads from multiple digital marketing channels, the company needs to prioritize leads most likely to convert, optimize its sales process, and understand the drivers of lead conversion.

Solution Methodology

Exploratory Data Analysis (EDA):

- Conduct univariate and bivariate analysis to identify key drivers of lead conversion.

Data Processing:

- minimal due to outliers not having a huge affect on the chosen models.

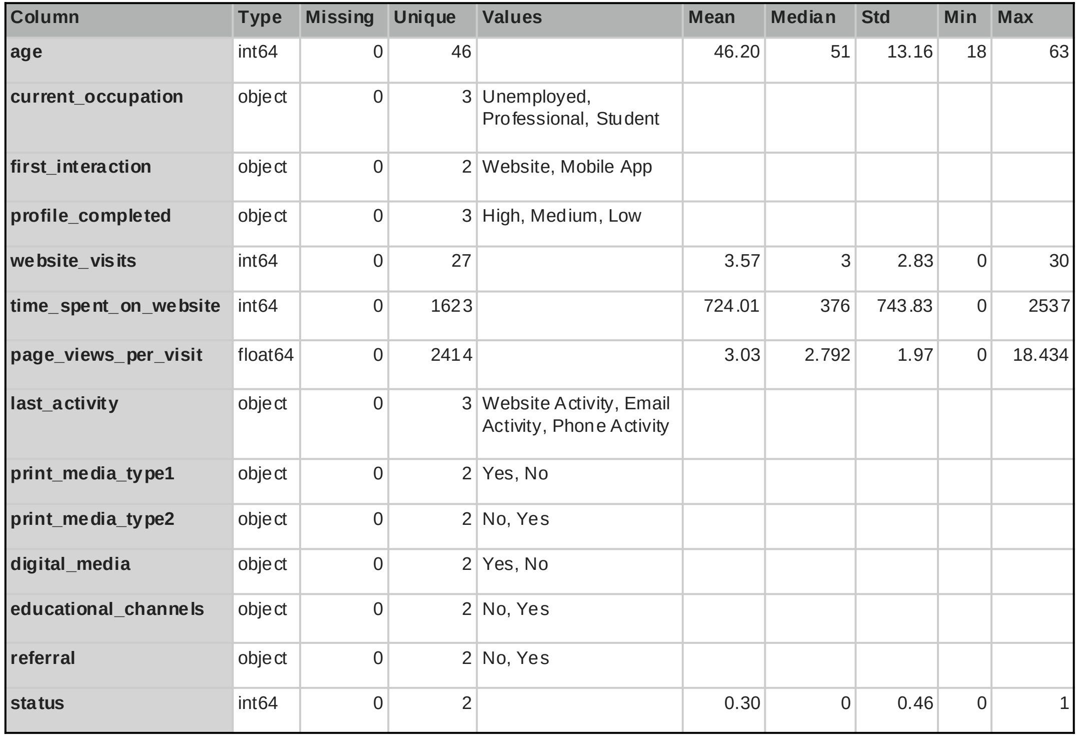
Model Development:

- Build and evaluated Decision Tree and Random Forest models (tuned vs untuned)

Business Insights:

- Identify high-conversion lead profiles for better resource allocation.
- Provide actionable predictions to optimize marketing and sales strategies.

Data Overview

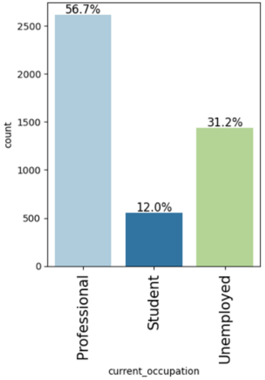


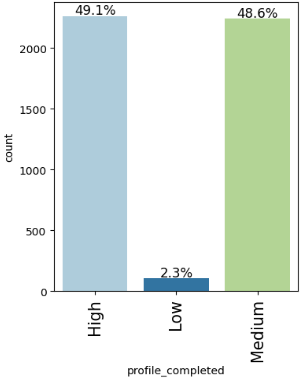
Column	Type	Missing	Unique	Values	Mean	Median	Std	Min	Max
age	int64	0	46		46.20	51	13.16	18	63
current_occupation	object	0	3	Unemployed, Professional, Student					
first_interaction	object	0	2	Website, Mobile App					
profile_completed	object	0	3	High, Medium, Low					
website_visits	int64	0	27		3.57	3	2.83	0	30
time_spent_on_website	int64	0	1623		724.01	376	743.83	0	2537
page_views_per_visit	float64	0	2414		3.03	2.792	1.97	0	18.434
last_activity	object	0	3	Website Activity, Email Activity, Phone Activity					
print_media_type1	object	0	2	Yes, No					
print_media_type2	object	0	2	No, Yes					
digital_media	object	0	2	Yes, No					
educational_channels	object	0	2	No, Yes					
referral	object	0	2	No, Yes					
status	int64	0	2		0.30	0	0.46	0	1

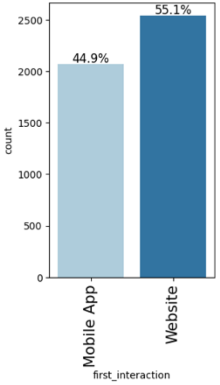
- 4612 x 15 dataset with the following contents:
- Demographics
  - age, occupation, profile completion
- Website metrics
  - first interactions, # website visits, time spent, page views per visit, etc
- Marketing Channels flags
  - Print, digital media, education, and referral channels
- Status is the target variable indicating whether leads have converted to paying customers

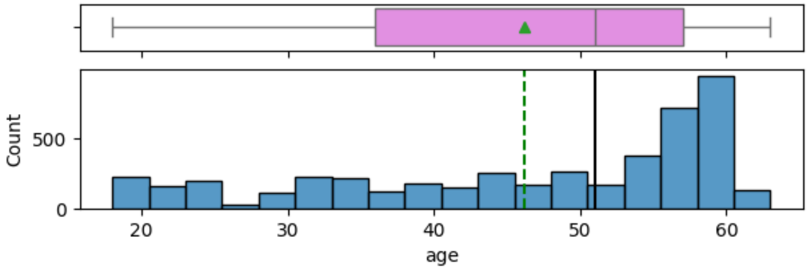
EDA Results – Univariate: Demographics

- Age ranges form 18-73 years old and is left skewed. With a mean around 45 and median around 50.5
- Most leads are professional (56.7%), followed by unemployed (31.2%), and finally students (12.0%)
- Only 2.3% of leads have a ‘low’ profile completion percentage (0-50% completion) while 48.6% have completed (50-75%) and 49.1% have a ‘high’ profile completion (75%-100%) percentage
- Slightly more leads first interact via Website (55.1%) compared to Mobile App (44.9%)



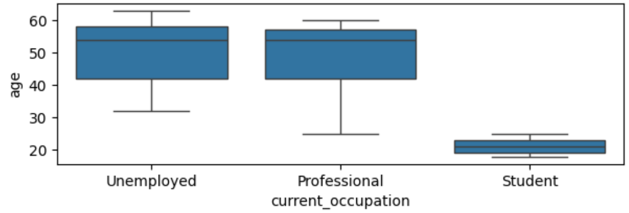


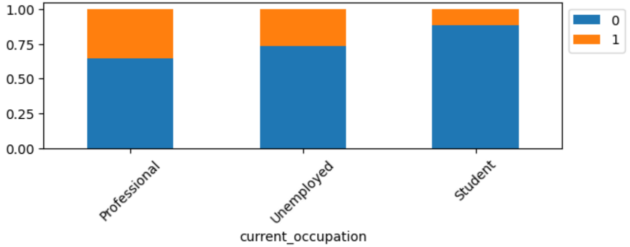


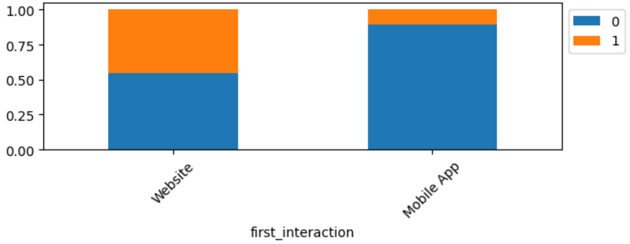


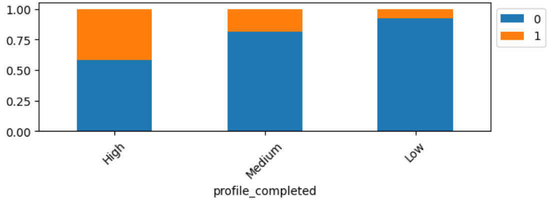
EDA Results - Bivariate: Demographics X Status

- Most leads on the lower end of the age range are students
- Professional and unemployed leads are seen to convert to paying customers more often than students
- Leads whose first interaction is on the website have a higher conversation rate (status) compared to mobile app
- Higher conversation status seen with higher profile completion percentage



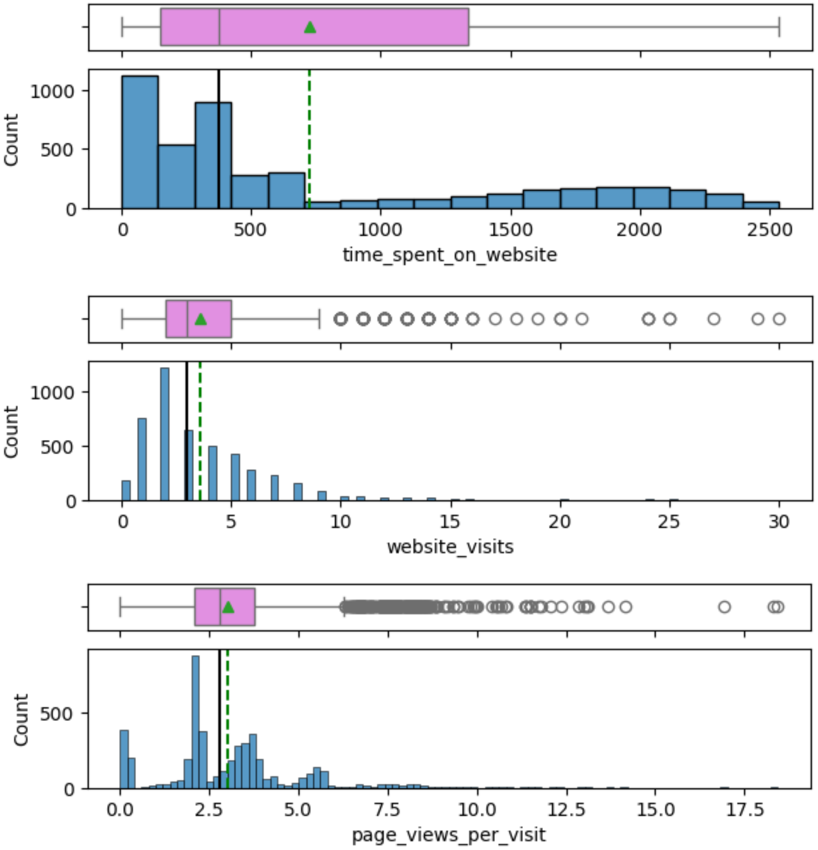


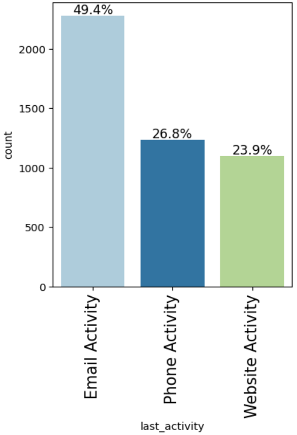




EDA Results – Univariate: Website Metrics

- Time spent, website visits, and page views per visit are all right skewed, and only time_spent_on_website does not have outliers
    - Page_views_per_visit appears multimodal with many outliers > 5.5
    - Website_visits is centered around 4 values > 10 considered outliers



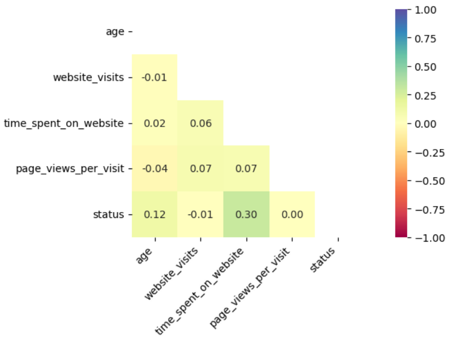


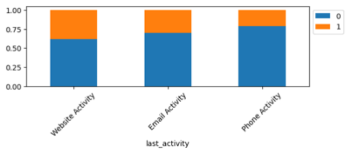
Email (49.4%) is the most common last interaction, followed by phone (26.8%), and website (23.9%)

EDA Results - Bivariate: Website Metrics X Status

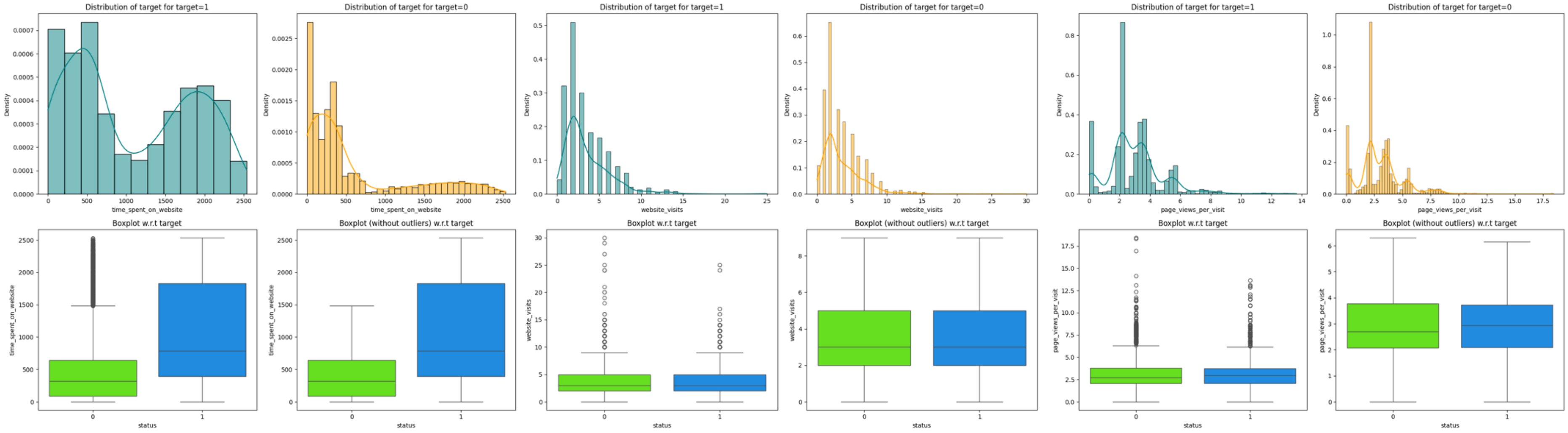
- Time spent on website has a bimodal distribution when leads convert (status = 1) compared to a right skewed distribution when status = 0. There is a weak positive correlation between the 2 variables (0.3) where converted leads spend more time on average and have a wider distribution (excluding outliers).

- Website visits appear and Page views per visit do not seem to be affected by status as there is similar distribution regardless of conversion status. There is also very little to no correlation between these metrics and conversion status



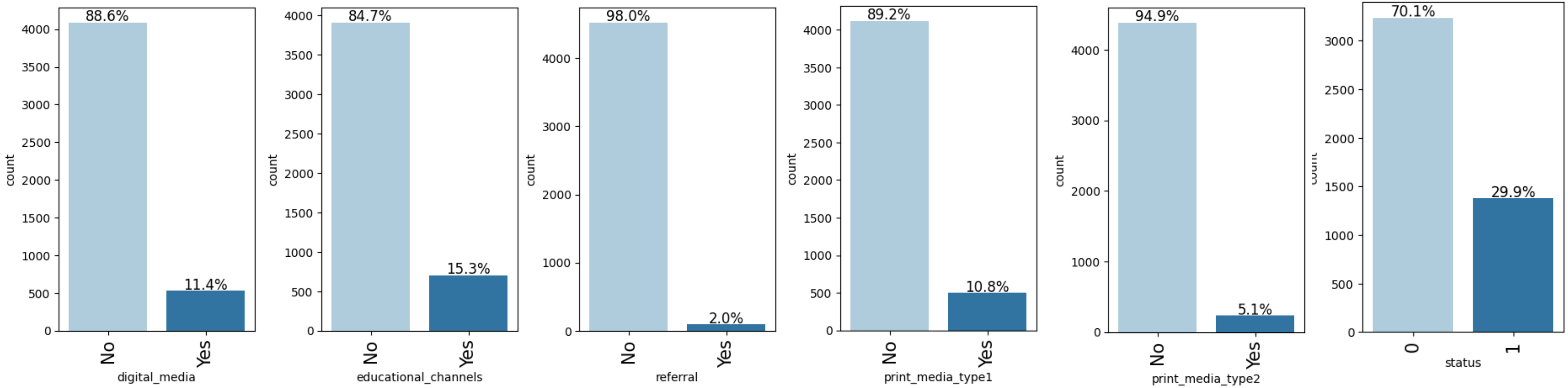


Website last activity has the highest conversion, followed by email and phone.



EDA Results – Univariate: Marketing Channels

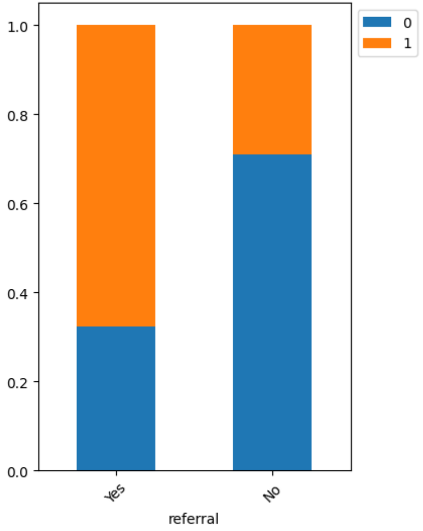
- Amongst the different channels in which a lead may have encountered an ad of ExtraaLearn, the most common is through education channels (15.3% yes), followed by digital_medial (11.4%), and print_media_type1 (10.8%)
- 29.9% of leads have converted to a paid customer

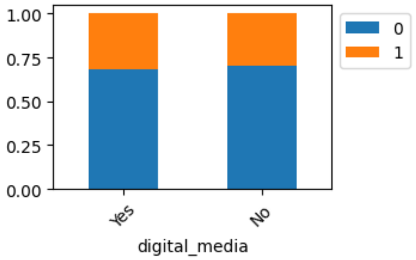


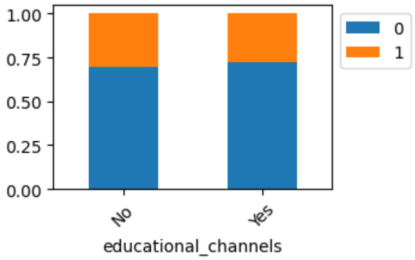
EDA Results - Bivariate: Marketing Channels X Status

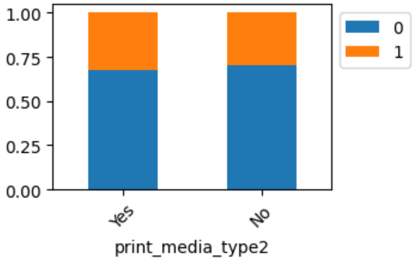
- Channels x Conversion status do appear to be significantly different for print media types, digital media, or education channels

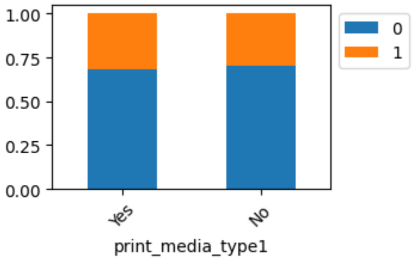
- Leads who hear about about ExtraaLearn through referral channels appear to be twice as likely to convert to paying customers (status = 1)



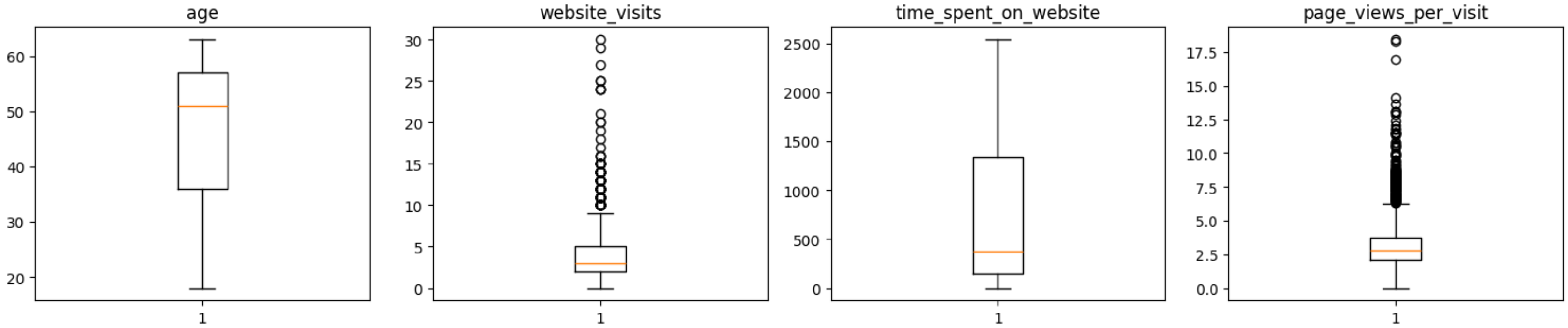








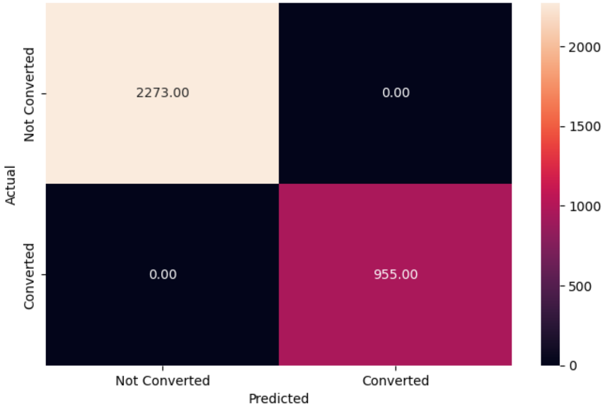
Data Preprocessing



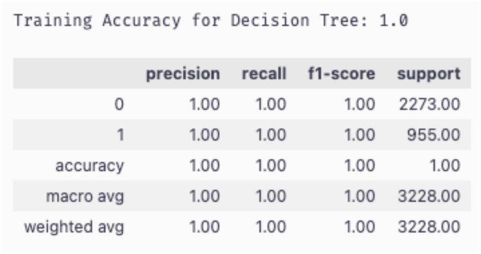
- Given the presence of outliers in website_visits and page_views_per_visit, it may be worth exploring methods to handle these outliers.
- However, since Decision Trees and Random Forests are robust to outliers due to their splitting mechanisms, addressing these outliers is not strictly necessary when using these models. If other models that are sensitive to outliers (e.g., linear regression or logistic regression) are chosen, we might consider combining the numeric features into a composite score or applying transformations to reduce the impact of the outliers.
    - See Appendix for sample feature engineering considerations

Model Building – Decision Tree

- Using a base Decision Tree, we observe that that there is overfitting in the model as there is perfect accuracy in the training data but only ~81% accuracy in the testing data
- Since losing a potential customer is a greater loss, we want to maximize Recall (to minimize False Negatives).
    - Recall is pretty low here as well (70%)

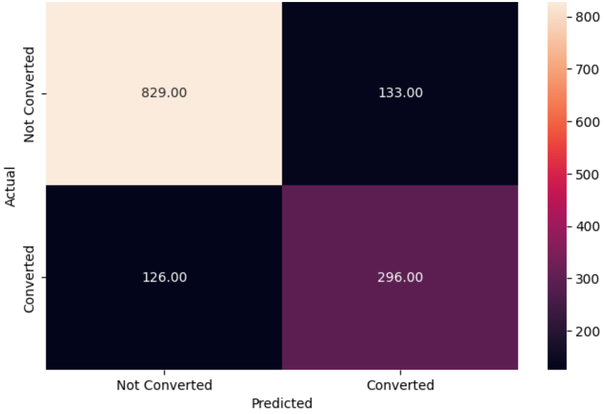


Training

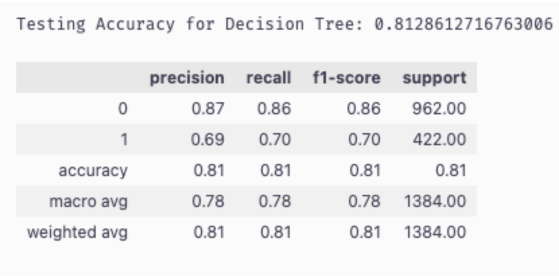


```
Training Accuracy for Decision Tree: 1.0
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2273.00
1	1.00	1.00	1.00	955.00
accuracy	1.00	1.00	1.00	1.00
macro avg	1.00	1.00	1.00	3228.00
weighted avg	1.00	1.00	1.00	3228.00



Testing

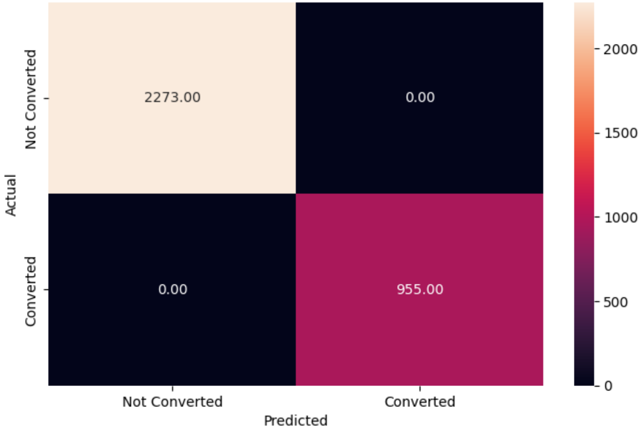


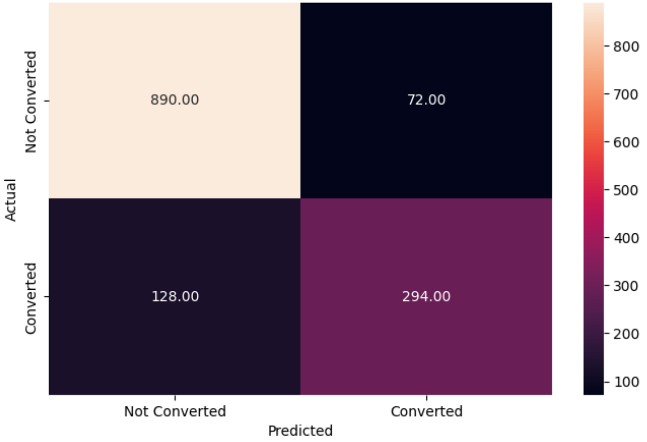
```
Testing Accuracy for Decision Tree: 0.8128612716763006
```

	precision	recall	f1-score	support
0	0.87	0.86	0.86	962.00
1	0.69	0.70	0.70	422.00
accuracy	0.81	0.81	0.81	0.81
macro avg	0.78	0.78	0.78	1384.00
weighted avg	0.81	0.81	0.81	1384.00

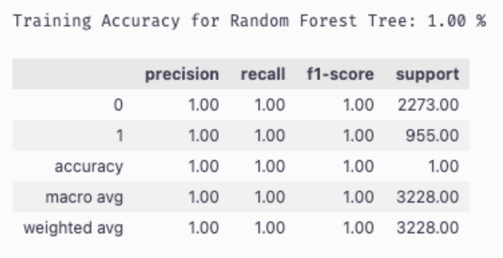
Model Building – Random Forest

- Again, we see that the base Random Forest model is overfitting on training data with 100% but fails to generalize to testing data (86%)
- We notice here that the testing accuracy (86%) is higher than the testing accuracy (81%) using the base Decision Tree
- Recall is comparably low (70%)





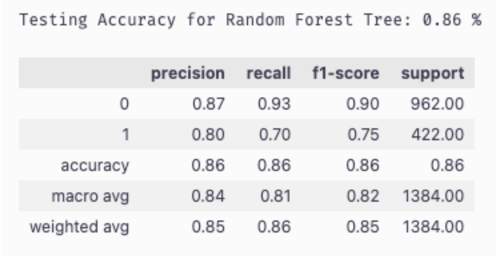
Training



```
Training Accuracy for Random Forest Tree: 1.00 %
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2273.00
1	1.00	1.00	1.00	955.00
accuracy	1.00	1.00	1.00	1.00
macro avg	1.00	1.00	1.00	3228.00
weighted avg	1.00	1.00	1.00	3228.00

Testing

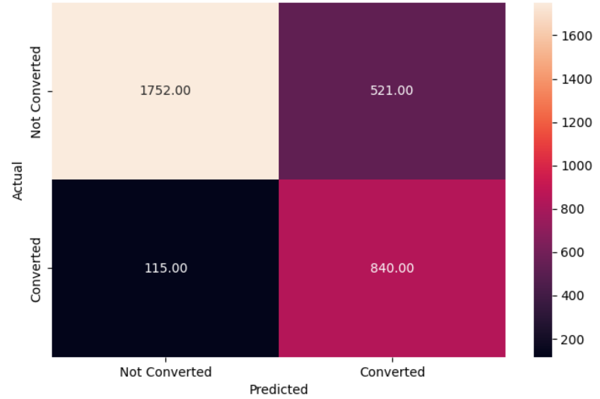


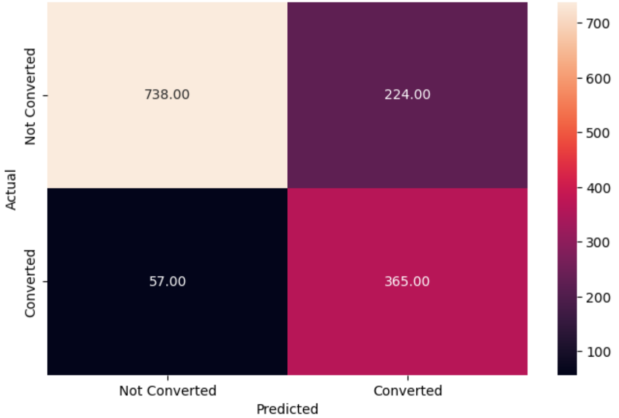
```
Testing Accuracy for Random Forest Tree: 0.86 %
```

	precision	recall	f1-score	support
0	0.87	0.93	0.90	962.00
1	0.80	0.70	0.75	422.00
accuracy	0.86	0.86	0.86	0.86
macro avg	0.84	0.81	0.82	1384.00
weighted avg	0.85	0.86	0.85	1384.00

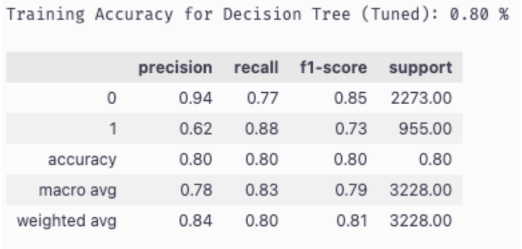
Model Building – Decision Tree - Hyperparameter Tuning

- After hyperparameter tuning, we now see that training and testing accuracy are similar (80%) – indicating the model is generalizing well and not overfitting
- Recall is 88% and 86% for training and testing, respectively. Much better than the previous model (70%)
- The precision is slightly lower (62%) than the testing precision of the base model (69%). However, less important than Recall





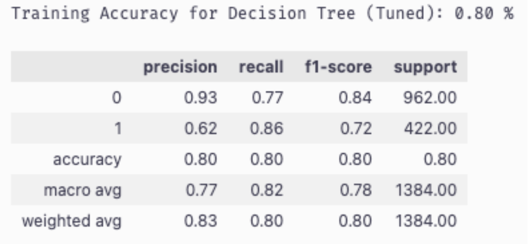
Training



```
Training Accuracy for Decision Tree (Tuned): 0.80 %
```

	precision	recall	f1-score	support
0	0.94	0.77	0.85	2273.00
1	0.62	0.88	0.73	955.00
accuracy	0.80	0.80	0.80	0.80
macro avg	0.78	0.83	0.79	3228.00
weighted avg	0.84	0.80	0.81	3228.00

Testing

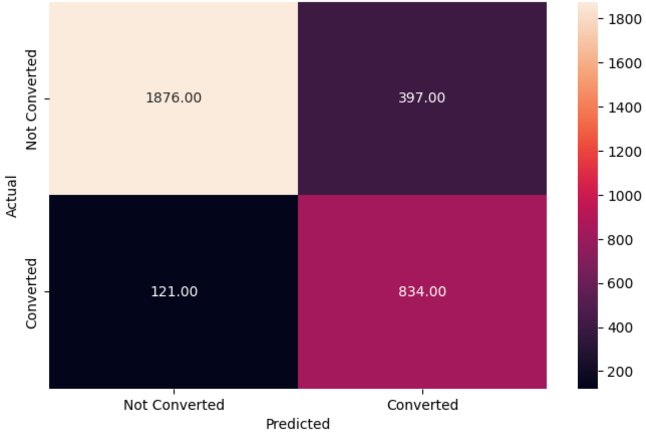


```
Training Accuracy for Decision Tree (Tuned): 0.80 %
```

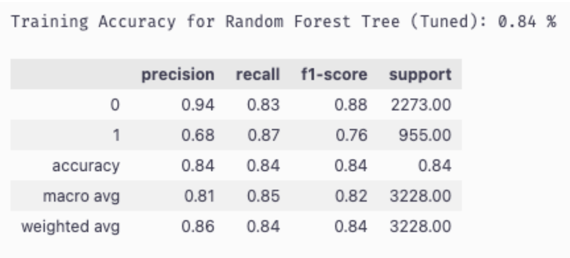
	precision	recall	f1-score	support
0	0.93	0.77	0.84	962.00
1	0.62	0.86	0.72	422.00
accuracy	0.80	0.80	0.80	0.80
macro avg	0.77	0.82	0.78	1384.00
weighted avg	0.83	0.80	0.80	1384.00

Model Building – Random Forest - Hyperparameter Tuning

- Similarly, after tunning, the training and testing accuracies are comparable (84%) and (86%), respectively
- The precision for class 0 performance (94%) is very high, indicating that when the model predicts a lead will not covert, it is correct most of the time, this cannot be said for class 1 (68%) indicating some false positives
- High recall for class 1 (85%) ensures model effectively identifies converted leads

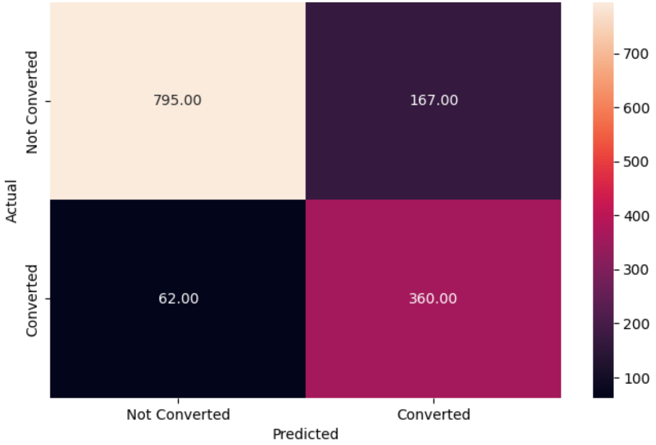


Training

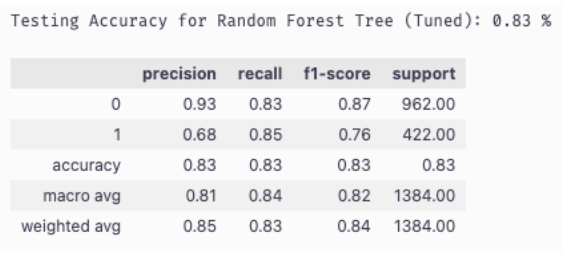


```
Training Accuracy for Random Forest Tree (Tuned): 0.84 %
```

	precision	recall	f1-score	support
0	0.94	0.83	0.88	2273.00
1	0.68	0.87	0.76	955.00
accuracy	0.84	0.84	0.84	0.84
macro avg	0.81	0.85	0.82	3228.00
weighted avg	0.86	0.84	0.84	3228.00



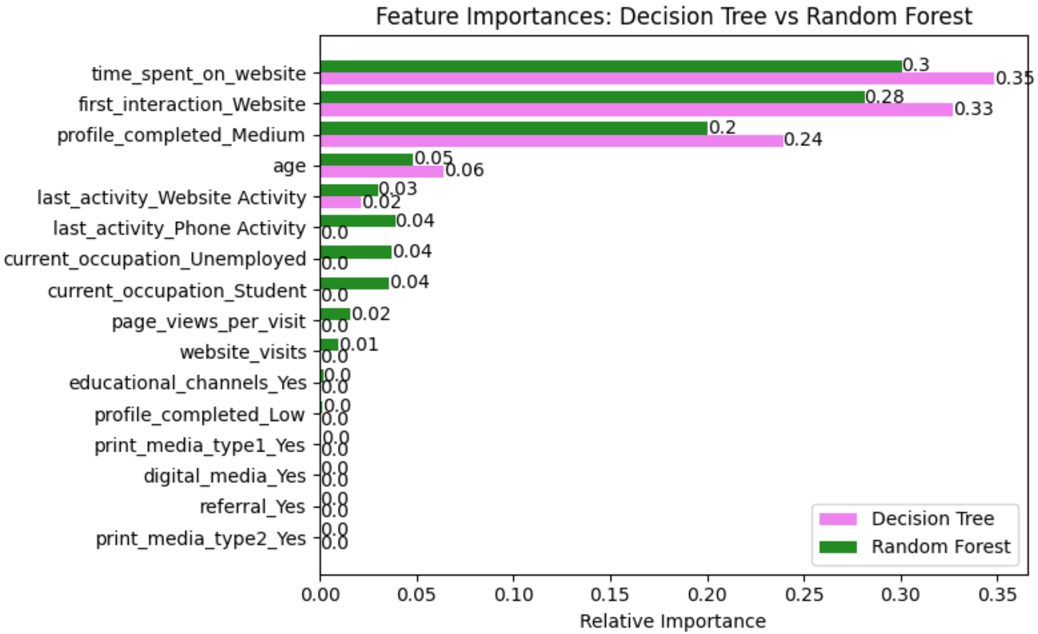
Testing



```
Testing Accuracy for Random Forest Tree (Tuned): 0.83 %
```

	precision	recall	f1-score	support
0	0.93	0.83	0.87	962.00
1	0.68	0.85	0.76	422.00
accuracy	0.83	0.83	0.83	0.83
macro avg	0.81	0.84	0.82	1384.00
weighted avg	0.85	0.83	0.84	1384.00

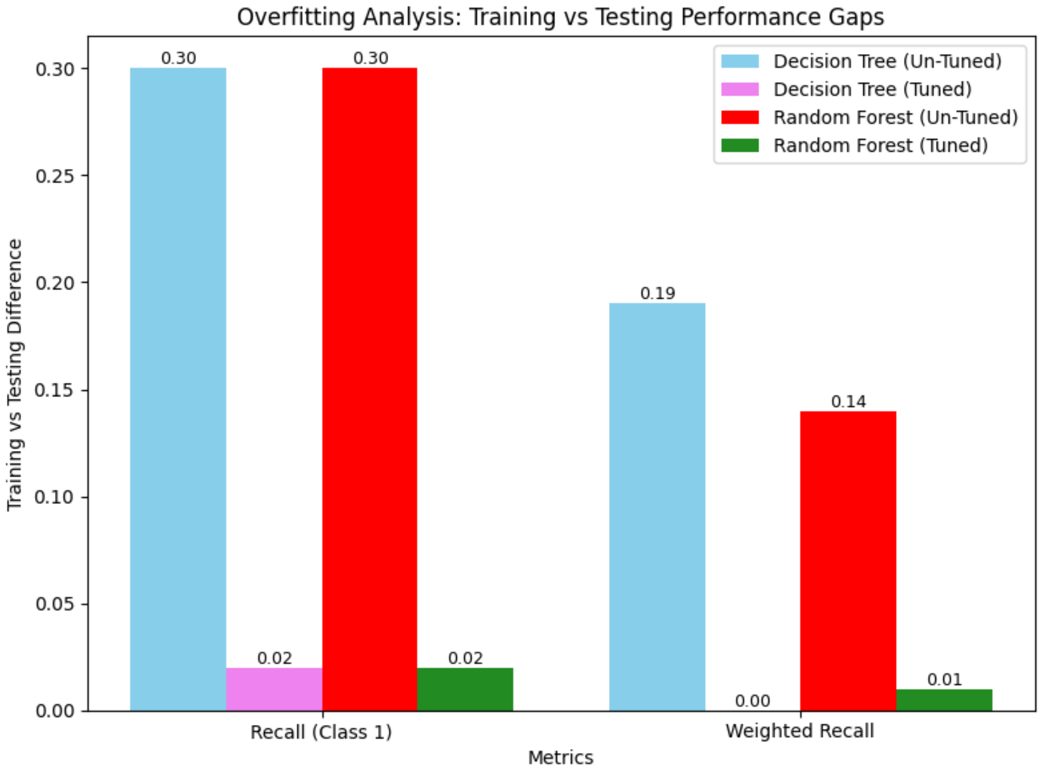
Feature Importance - Insights



- Decision Tree: Highly focused on a few dominant features, making it interpretable but potentially less robust.
  - 90% of importance covered by top 3 features
- Random Forest:
  - The importance is slightly more distributed compared to the decision tree, with features like age (5%), last_activity_Phone Activity (4%), and current_occupation_Unemployed (4%) playing a slightly larger role

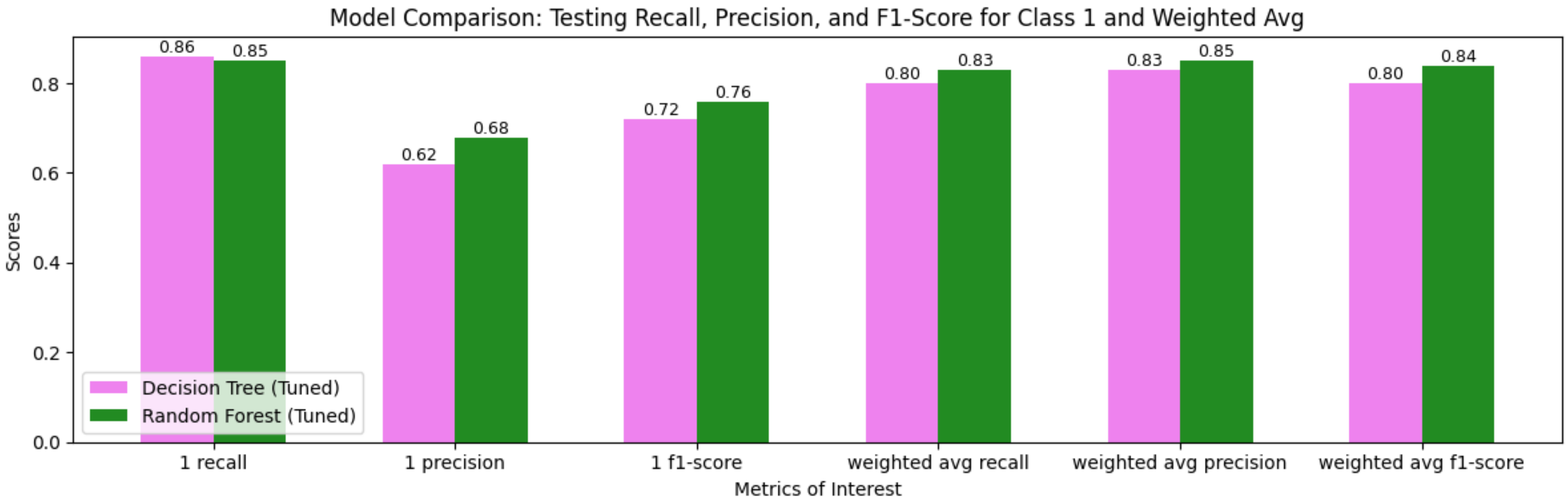
- Both models consistently rank time_spent_on_website, first_interaction_Website, and profile_completed_Medium as the most important.
- Features like educational_channels_Yes, digital_media_Yes, and print_media_type1_Yes contribute nothing to either model. These might be candidates for removal if their lack of importance persists in other analyses.

Model Performance Summary



- Clearly, we do not want to choose the un-tuned versions of either Decision Trees or Random Forests as we see those models have a larger performance gaps between their testing and training performance across the metrics, we are most concerned with :
  - Recall (Class 1)
  - Weighted Recall

Model Performance Summary



- Other than Class 1 Recall, which the Tuned Random Forest model loses to the tuned Decision Tree by 1%, the tuned Random Forest Model outperforms the Decision Tree Model in almost every other metric.
- The weighted avg recall reflects how well the model across all classes, weighted by support
- Weighted avg F1- score balances precision and recall across all classes

Conclusions and Recommendations

Conclusions

1. **Model Choice**: The **tuned Random Forest model** is the optimal choice for predicting lead conversion likelihood due to its superior balance between precision and recall, especially for Class 1 (converted leads).
2. **Key Drivers**: Features such as **time spent on the website** and **first interaction through the website** are strong indicators of lead conversion, making them crucial touchpoints for engagement.

Business Recommendations

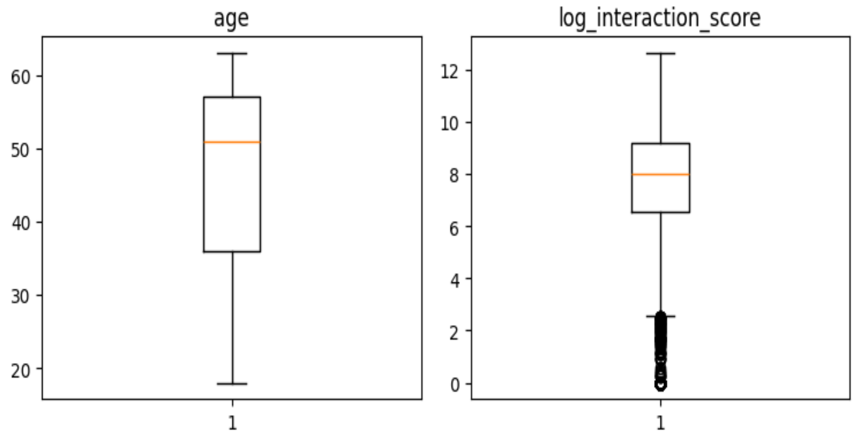
- **Enhance Engagement**: Invest in strategies that encourage website interactions and increase time spent on the platform.
- Tailor follow-ups for leads with medium-to-high profile completion to improve conversion rates.
- **Optimize Marketing Efforts**: Leverage insights on key drivers to design targeted campaigns, particularly for digital channels that emphasize website engagement.
    - Features like educational_channels_Yes and digital_media_Yes have minimal or no impact on conversion in the model. If these efforts are not yielding results, consider reallocating the budget to higher-impact channels such as **website engagement strategies** or **direct interactions via email or phone**.

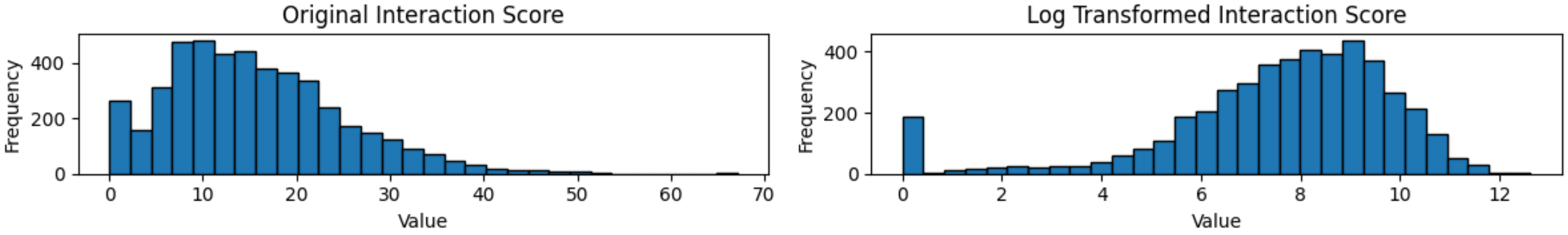
APPENDIX

Data Preprocessing : Feature Engineering

- An example of potential feature engineering is the combination of website_visits, time_spent_on_website, and page_views_per_visit into a single composite metric, such as an 'interaction_score’.

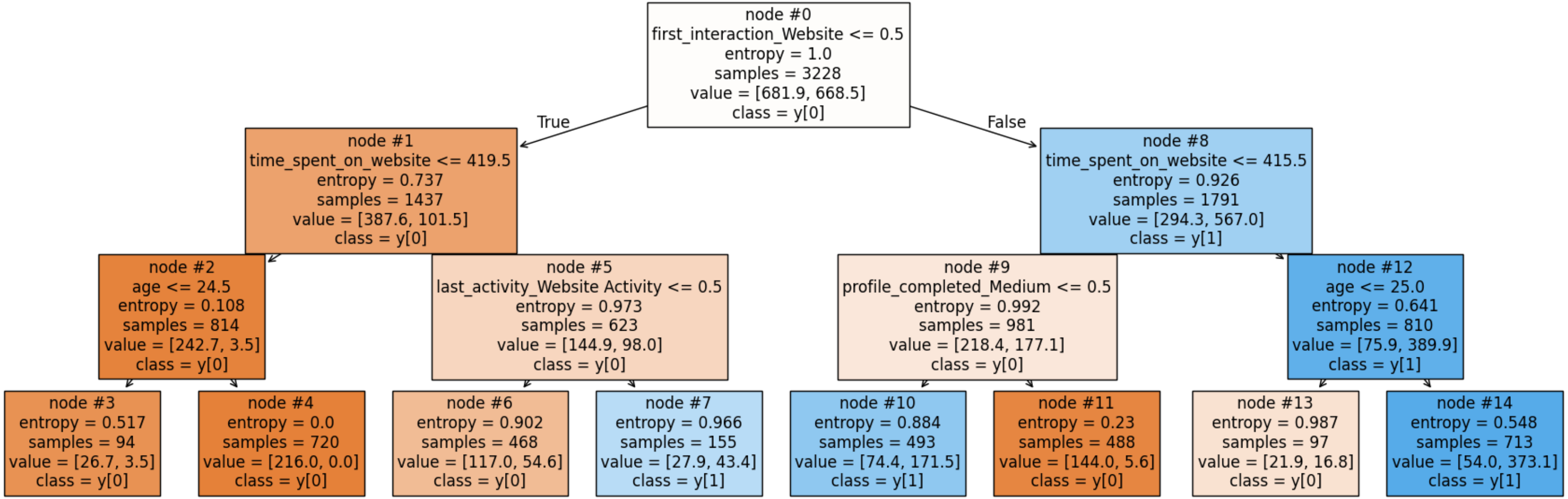
- While this approach may enhance model performance, it can reduce interpretability, making it harder to derive clear insights and actionable business recommendations





Decision Tree (Tuned) : Visualized

- Please add any other pointers or screenshots (if needed)



node #0
first_interaction_Website <= 0.5
entropy = 1.0
samples = 3228
value = [681.9, 668.5]
class = y[0]

True

node #1
time_spent_on_website <= 419.5
entropy = 0.737
samples = 1437
value = [387.6, 101.5]
class = y[0]

node #2
age <= 24.5
entropy = 0.108
samples = 814
value = [242.7, 3.5]
class = y[0]

node #3
entropy = 0.517
samples = 94
value = [26.7, 3.5]
class = y[0]

node #4
entropy = 0.0
samples = 720
value = [216.0, 0.0]
class = y[0]

node #5
last_activity_Website Activity <= 0.5
entropy = 0.973
samples = 623
value = [144.9, 98.0]
class = y[0]

node #6
entropy = 0.902
samples = 468
value = [117.0, 54.6]
class = y[0]

node #7
entropy = 0.966
samples = 155
value = [27.9, 43.4]
class = y[1]

False

node #8
time_spent_on_website <= 415.5
entropy = 0.926
samples = 1791
value = [294.3, 567.0]
class = y[1]

node #9
profile_completed_Medium <= 0.5
entropy = 0.992
samples = 981
value = [218.4, 177.1]
class = y[0]

node #10
entropy = 0.884
samples = 493
value = [74.4, 171.5]
class = y[1]

node #11
entropy = 0.23
samples = 488
value = [144.0, 5.6]
class = y[0]

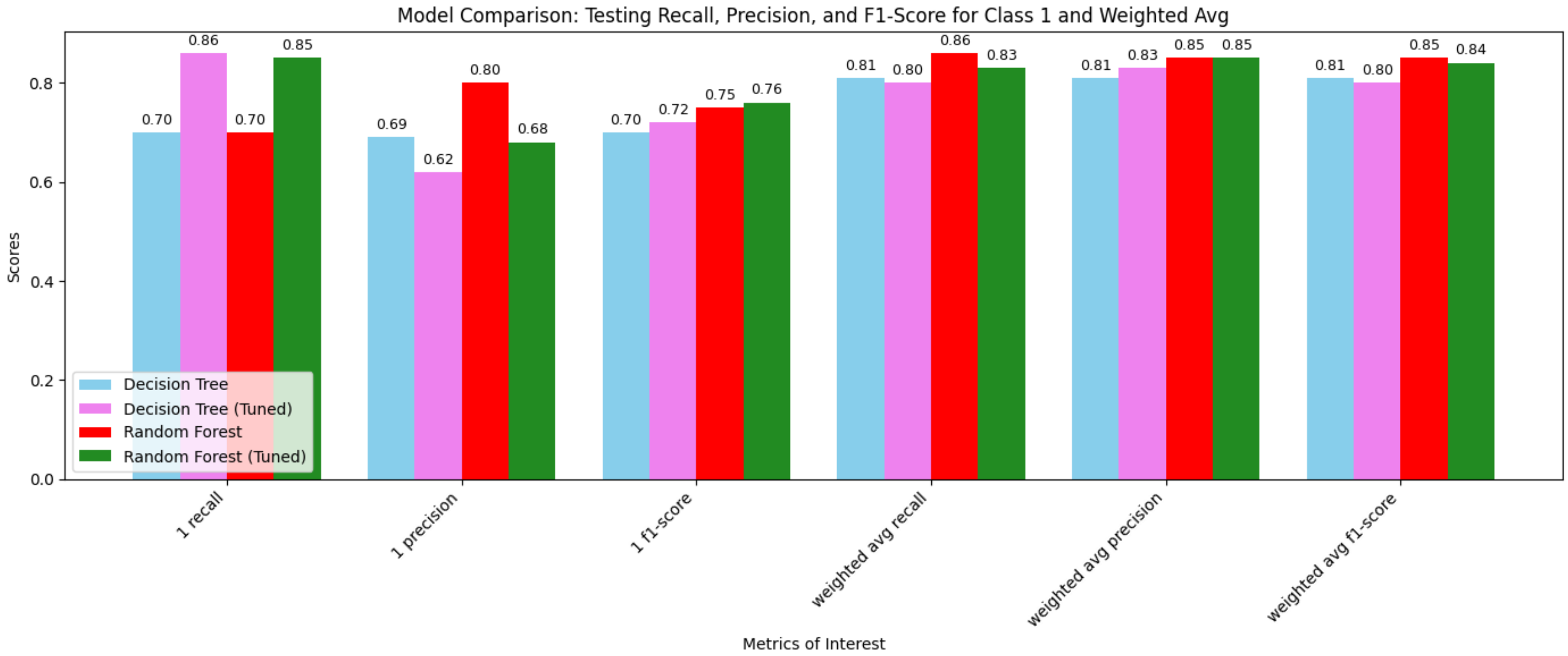
node #12
age <= 25.0
entropy = 0.641
samples = 810
value = [75.9, 389.9]
class = y[1]

node #13
entropy = 0.987
samples = 97
value = [21.9, 16.8]
class = y[0]

node #14
entropy = 0.548
samples = 713
value = [54.0, 373.1]
class = y[1]

Model Comparison (All models)

- Please add any other pointers or screenshots (if needed)



Happy Learning !