



Lead Scoring Case Study

Md Nurul Akter, Abhishek Vaibhav Pathak and Harshit Ranjan



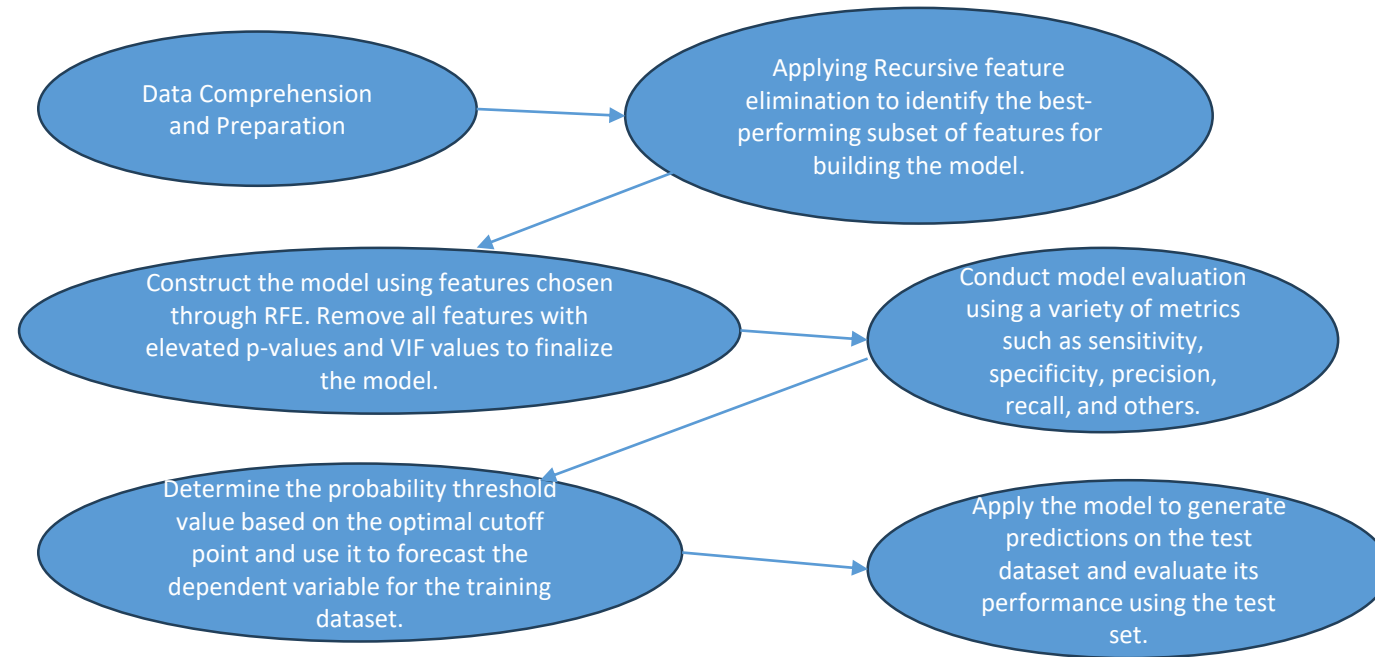
Business Objective

Develop a Logistic Regression model to estimate the likelihood of lead conversion for each potential client.

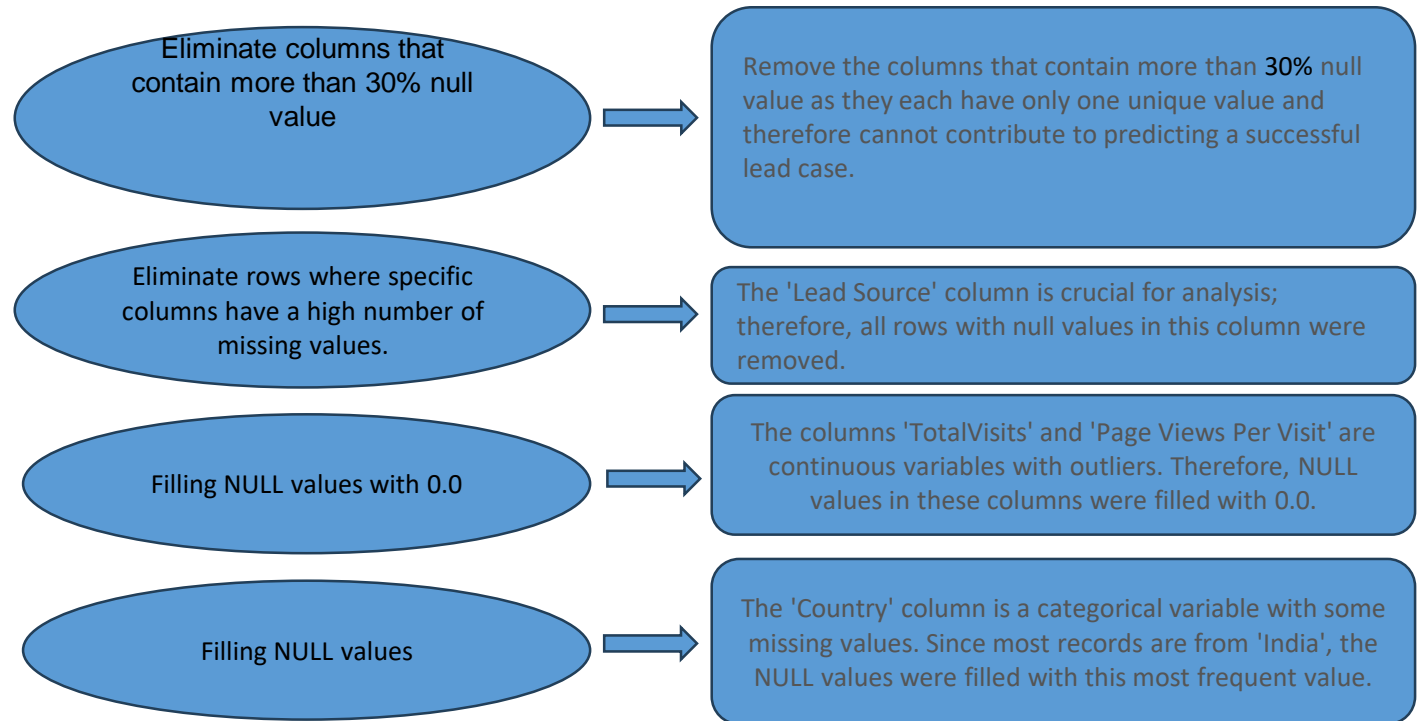
Determine a probability cutoff point, above which a lead will be classified as converted and below which it will be classified as not converted.

Calculate the Lead Score for each lead by multiplying the Lead Conversion probability.

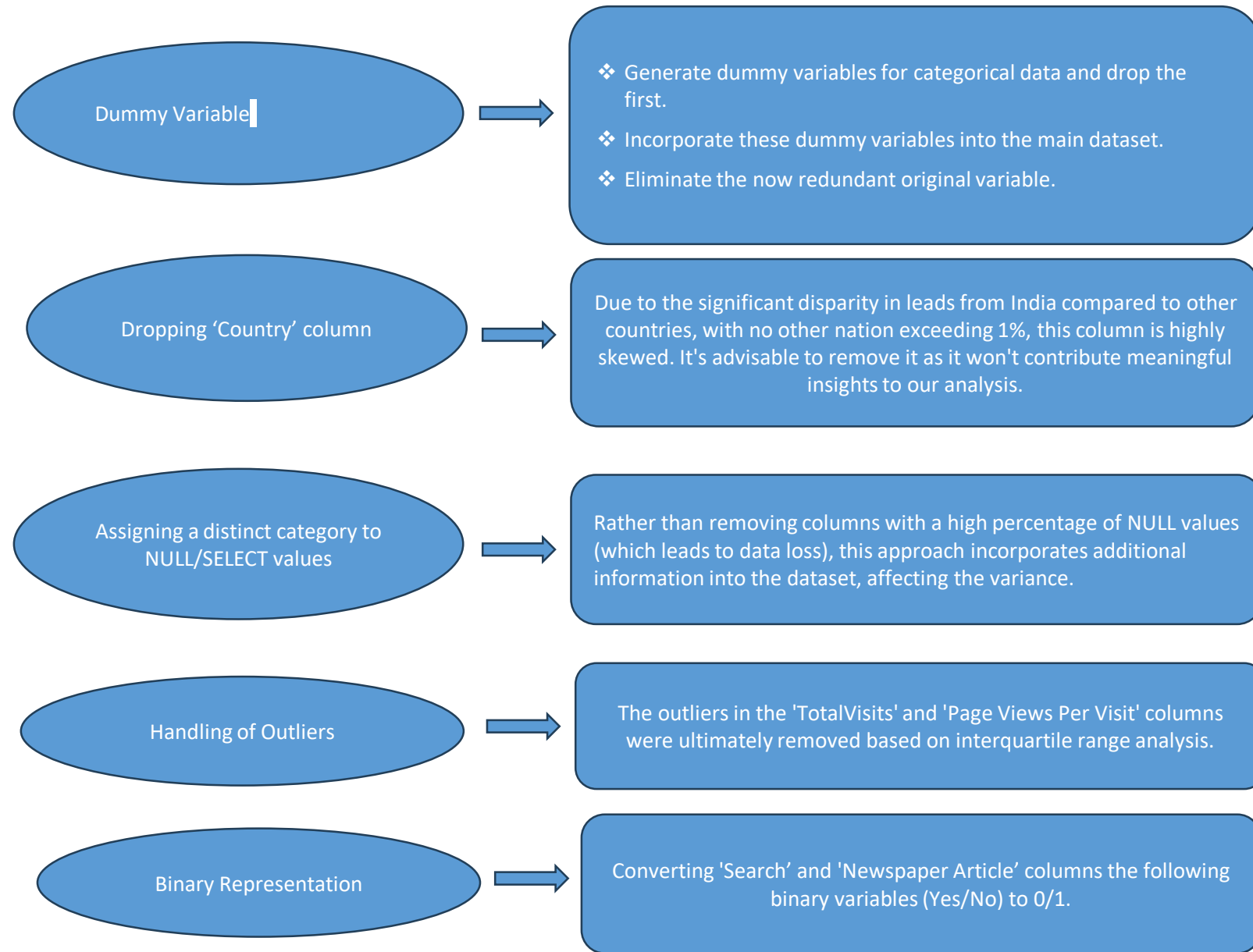
Approach to Resolving Issues



Data preprocessing and Feature Development



Data preprocessing and Feature Development



Data preprocessing and Feature Development

Training and Testing Data Split



- ❖ The initial dataframe was divided into training and testing datasets. The training dataset was utilized to develop the model, while the testing dataset was used to assess its performance.

Feature Normalization



- ❖ Scaling facilitates interpretation. It's crucial to have all variables, especially categorical ones with binary values (0 and 1), on the same scale to make the model more interpretable.
- ❖ 'Standardization' was employed to scale the data for modeling. This process transforms the data into a standard normal distribution with a mean of zero and a standard deviation of one.

Feature Selection via Recursive Feature Elimination (RFE)

Recursive Feature Elimination (RFE) is an optimization method used to identify the most effective subset of features. It involves repeatedly building a model, selecting the best features based on their coefficients, and then removing those features from consideration. This process continues with the remaining features until all have been evaluated. The features are then ranked based on the order in which they were eliminated.

RFE

```
1 # Creating an object
2 logreg=LogisticRegression()
```

```
1 rf0 = RFE(estimator=logreg, n_features_to_select=19)
2 rf0 = rf0.fit(X_train, y_train)
```

```
1 rf0.support_
```

```
1 col=X_train.columns[rf0.support_]
2 X_train_1=sm.add_constant(X_train[col])
```



Executing RFE with the number of variables set to 19.


Developing the model

- The Logistic Regression model is constructed using Generalized Linear Models from StatsModels.
- The model is initially developed using the 19 variables chosen by RFE.
- Unnecessary features are sequentially removed after evaluating p-values (< 0.5) and VIF (< 5), and the model is rebuilt several times.
- The final model, which includes 13 features, successfully meets both the significance and multicollinearity criteria.

Features	VIF
const	763.28
Lead Origin_Lead Add Form	4.03
Lead Source_Reference	4.00
Last Notable Activity_Modified	1.52
Last Notable Activity_SMS Sent	1.33
Last Activity_Olark Chat Conversation	1.28
Last Activity_Converted to Lead	1.18
Total Time Spent on Website	1.11
Lead Source_Direct traffic	1.11
What is your current occupation_Working Profes...	1.10
Do Not Email	1.08
Lead Number	1.07
Last Activity_Page Visited on Website	1.07

Estimating the likelihood of conversion and the predicted outcomes

- ❖ Constructing a dataframe that includes the actual conversion indicators and the estimated probabilities.
- ❖ Displaying the top 5 entries of the dataframe in the image to the right.

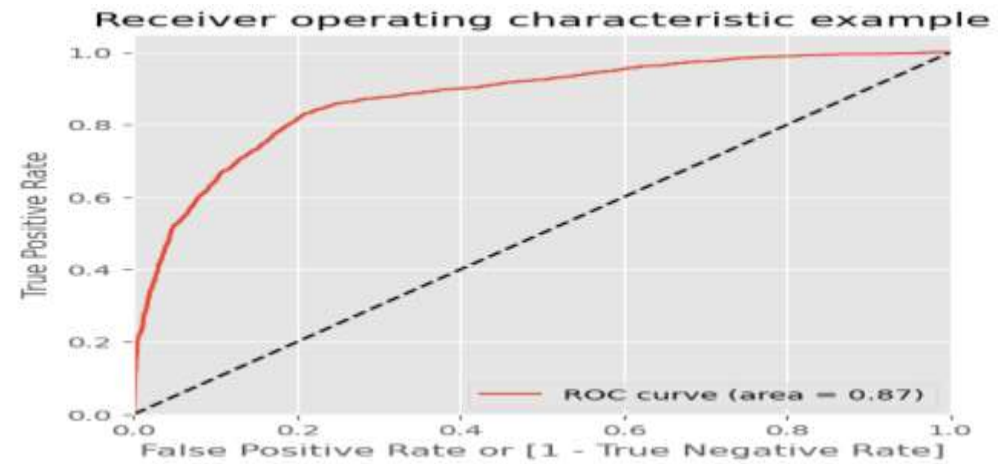


	Converted	Converted_probability	ID
1871	0	0.207182	1871
6795	0	0.237922	6795
3516	0	0.225560	3516
8105	0	0.768386	8105
3934	0	0.130633	3934

Generating the ROC curve

Receiver Operating
Characteristic (ROC)
Curve

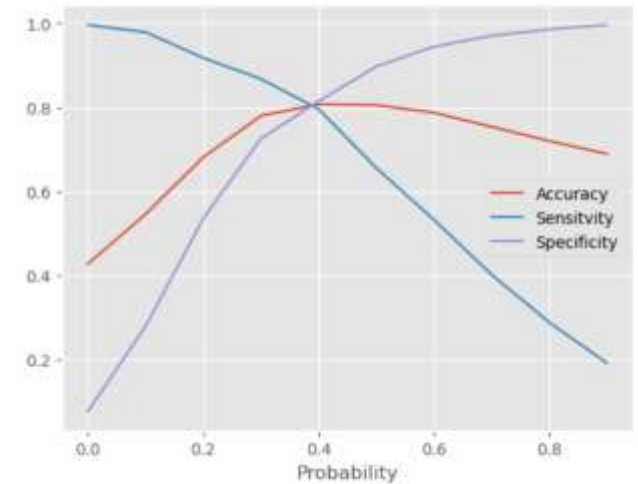
It illustrates the tradeoff
between sensitivity and
specificity, where an
increase in sensitivity is
typically associated with a
decrease in specificity.



Determining the best probability cutoff value

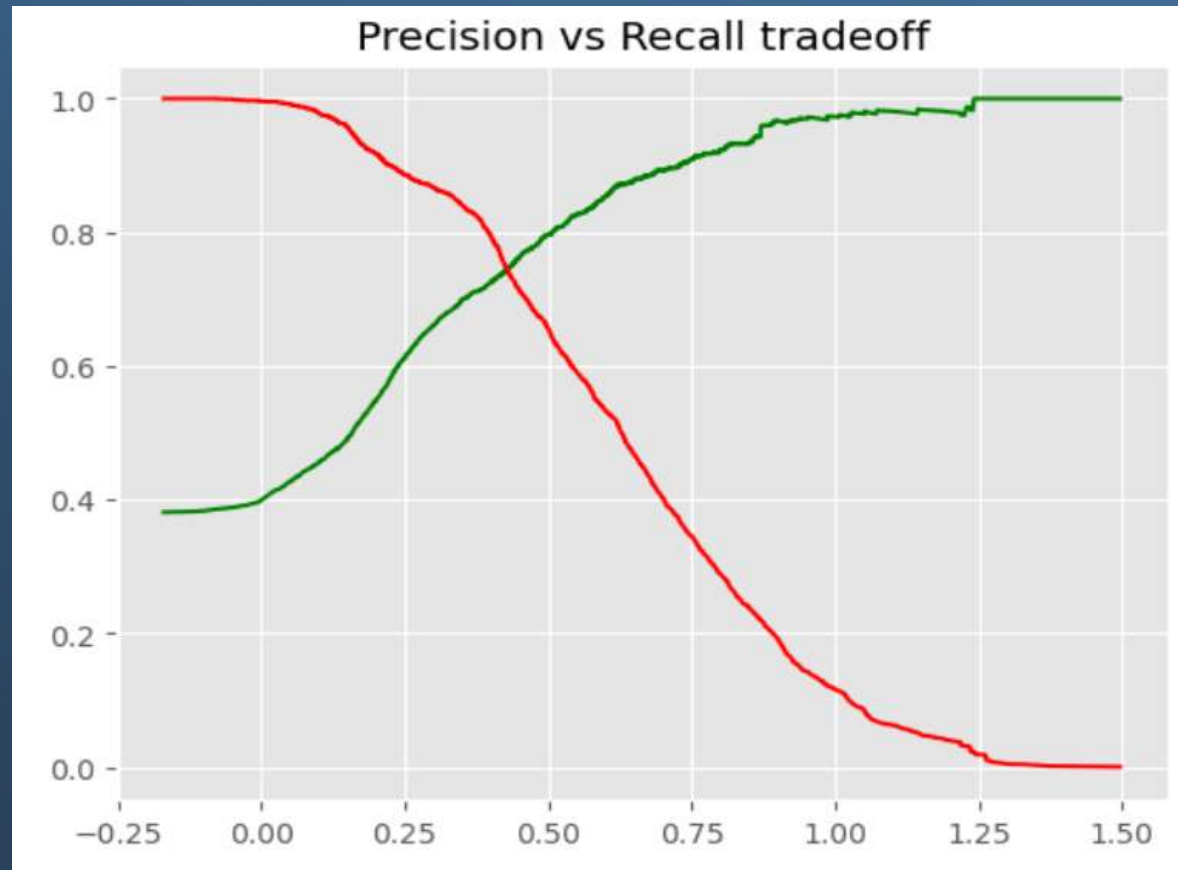
The ideal cutoff probability is the point where sensitivity and specificity are balanced.

- ❖ The accuracy, sensitivity, and specificity were computed for different probability thresholds and illustrated in the graph on the right.
- ❖ The curve indicates that a cutoff probability of 0.4 is the optimal point.
- ❖ At this threshold, all three metrics—accuracy, sensitivity, and specificity—were found to be around 80%, which is considered a satisfactory level.



Precision and Recall tradeoff

There's a balance
between the two,
converging at 0.5.



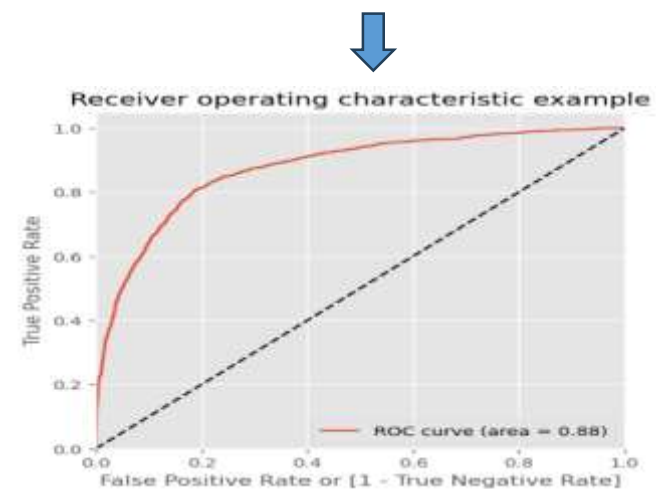
Generating forecasts for the test set

- The final model, trained on the training dataset, is utilized to predict outcomes for the test dataset.
- The predicted probabilities were incorporated into the leads within the test dataframe.
- Applying a probability threshold of 0.4, predictions were made on the test dataset to determine whether the leads would convert or not.

	Converted	Converted_probability	ID
1871	0	0.234848	1871
6795	0	0.232586	6795
3516	0	0.255542	3516
8105	0	0.719279	8105
3934	0	0.081274	3934

Plotting the ROC curve

The curve leans left with 88% area under it, indicating a highly accurate model. AOC measures the tradeoff between true and false positive rates and reflects model stability. A larger AOC suggests better class distinction. Our second model, with 88% AOC, outperforms the first.



Scaling and Predicting test dataset

- The test dataset is scaled using the same transformation applied to the training set, specifically for the 'Total Time Spent on Website' feature.
- Selected columns from the training data are applied to the test data, excluding the first column, to prepare the test dataset for prediction.
- A constant term is added to the test dataset to ensure the model includes an intercept during prediction.
- The model's predictions on the test dataset are stored in a new DataFrame, capturing both the actual conversion values and the predicted probabilities.

	Converted	Converted_Probability	ID
4269	1	0.655122	4269
2376	1	0.918245	2376
7766	1	0.523668	7766
9199	0	-0.005999	9199
4359	1	0.663867	4359

Model Evaluation

- The model's predictions on the test dataset were generated using a probability cutoff of 0.4, creating a new column to indicate predicted outcomes.
- A 'Predicted' column was added to the test dataset, where probabilities above 0.4 were classified as 1, and those below were classified as 0.
- The DataFrame now displays the actual conversion, predicted probability, ID, and the predicted outcome for each test case.
- The model achieved a precision score of 0.7398, indicating that about 73.98% of the predicted positive cases were true positives.
- The recall score was 0.7817, meaning the model correctly identified 78.17% of the actual positive cases.

	Converted	Converted_Probability	ID	Predicted
4269	1	0.655122	4269	1
2376	1	0.918245	2376	1
7766	1	0.523668	7766	1
9199	0	-0.005999	9199	0
4359	1	0.663867	4359	1

Lead Score

- New columns 'Lead Number' and 'Lead Score' were added to the test dataset for better tracking and evaluation.
- The 'Lead Number' was assigned using the corresponding index from the original dataset, linking each prediction back to its lead.
- The 'Lead Score' was calculated by multiplying the predicted probability by 100 and rounding the result to create a score out of 100.
- This scoring system provides an easy-to-understand metric for assessing the likelihood of a lead converting.
- The resulting DataFrame now includes the original conversion status, predicted probability, ID, predicted outcome, lead number, and lead score for each test case.

	Converted	Converted_Probability	ID	Predicted	Lead Number	Lead Score
4269	1	0.655122	4269	1	0	66
2376	1	0.918245	2376	1	0	92
7766	1	0.523668	7766	1	0	52
9199	0	-0.005999	9199	0	0	-1
4359	1	0.663867	4359	1	0	66

Conclusion

- The model is stable and performs reliably, with a recall score that surpasses the precision score, indicating its strong ability to identify actual positive cases. The accuracy, precision, and recall metrics are all within acceptable limits, suggesting that the model is both effective and dependable. Additionally, the model is adaptable, making it suitable for the company's evolving needs. Key features influencing lead conversion include the lead's origin from the "Lead Add Form," the current occupation being "Working Professional," and the last activity being "SMS Sent."



THANK YOU