LEAD SCORING ASSIGNMENT

By : Pooja Jagdale & Mayur Jangale

Goals to Achieve

Build a logistic regression model to assign a lead score between 0 and 100 to each of the leads which can be used by the company to target potential leads.

A higher score would mean that the lead is hot, i.e. is most likely to convert whereas a lower score would mean that the lead is cold and will mostly not get converted.

Problem Statement

- An education company named X Education sells online courses to industry professionals. On any given day, many professionals who are interested in the courses land on their website and browse for courses.
- The company wants its leads generation rate to be increased. For which they need to identify what are the potential leads aka Hot Leads.
- For this company wants to create a model where we can assign lead score to each of the identified leads for higher chances of conversion. Target for the company is 80 percent.

Problem Solving Methodology

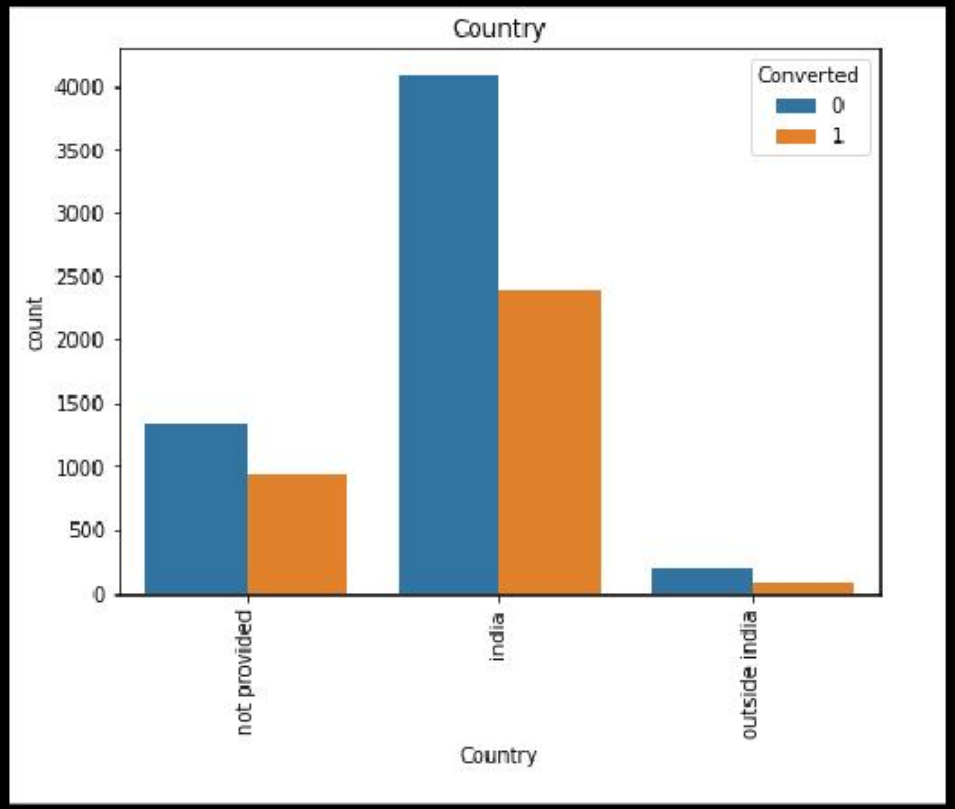
- Reading and Understanding the data
- Data Cleaning
- Data Visualization using EDA
- Data Preparation for Modelling
- Model Building: Building the model with features selected by RFE. Eliminate all features with high p-values and VIF values and finalize the model
- Model Evaluation: with various metrics like sensitivity, specificity, precision, recall, etc.

Reading and Understanding the data

We import all the necessary libraries for e.g. NumPy, pandas, matplotlib seaborn etc., import all the warnings.
We read the data and check the no. of rows and columns. We also, check if there are any missing/ null values or not. Afterwards, we see the statistical summary of the data.

Data Cleaning

- We saw there were few columns with high percentage of null values, so we decided to drop those columns.
- Few columns had null values but the columns were important for analysis, so we replaced all null values with 'Not Provided'.
- Few of the columns had values as 'Select' so we replaced it by 'NaN'
- Few columns were having outliers and the treatment of outliers was performed.



Data Visualization using EDA

- For data visualization we performed exploratory data analysis (EDA). We first did the univariate analysis of categorical and continuous data.
- Moved ahead with the bi-variate analysis of categorical and continuous data.
- We also found in country data most of the records were from India and few were from outside India and we classified as same.
- We found correlations between variables by using different plots.

Data Preparation for Modelling

- Data preparation for multiple linear regression involves handling the categorical variables first and then performing dummy encoding.
- We then performed the train and test split using 70%-30% rule and then performed the scaling of variables. Since scaling of variables is an important step, we may have different variables of different scales. So, its important to have everything on the same scale for the model to be easily interpretable.
- Therefore, we used **MinMaxScaler** for the same.

```
X_train[['TotalVisits', 'Page Views Per Visit', 'Total Time Spent on Website']] = scaler.fit_transform(X_train[['TotalVisits', 'F
X_train.head()
```

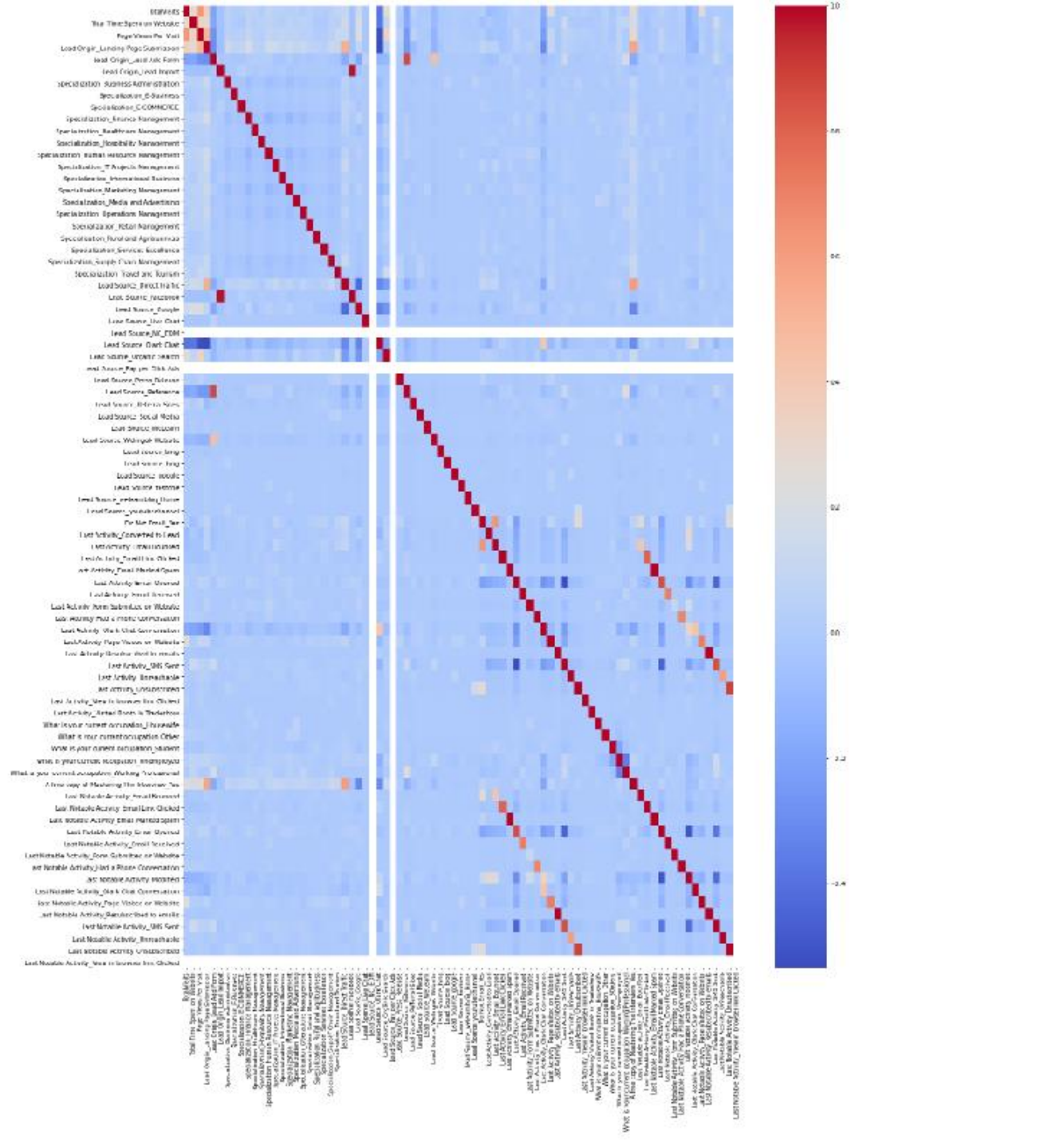
	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Lead Origin_Landing Page Submission	Lead Origin_Lead Add Form	Lead Origin_Lead Import	Specialization_Business Administration	Specialization_E-Business	Specialization_E-COMMERCE	Specialization_Fin Manage
1289	0.014184	0.612676	0.383333	1	0	0	0	0	0	
3604	0.000000	0.000000	0.000000	0	0	0	0	0	0	
5584	0.042553	0.751761	0.250000	1	0	0	0	0	0	
7679	0.000000	0.000000	0.000000	0	0	0	0	0	0	
7563	0.014184	0.787852	0.383333	1	0	0	0	0	0	

5 rows × 81 columns

Feature Selection Using RFE

- Recursive feature elimination is an optimization technique for finding the best performing subset of features.
- It is based on the idea of repeatedly constructing a model and choosing either the best (based on coefficients), setting the feature aside and then repeating the process with the rest of the features.
- This process is applied until all the features in the dataset are exhausted. Features are then ranked according to their time of elimination.

To check the correlation among varibles



Generalized Linear Model Regression Results

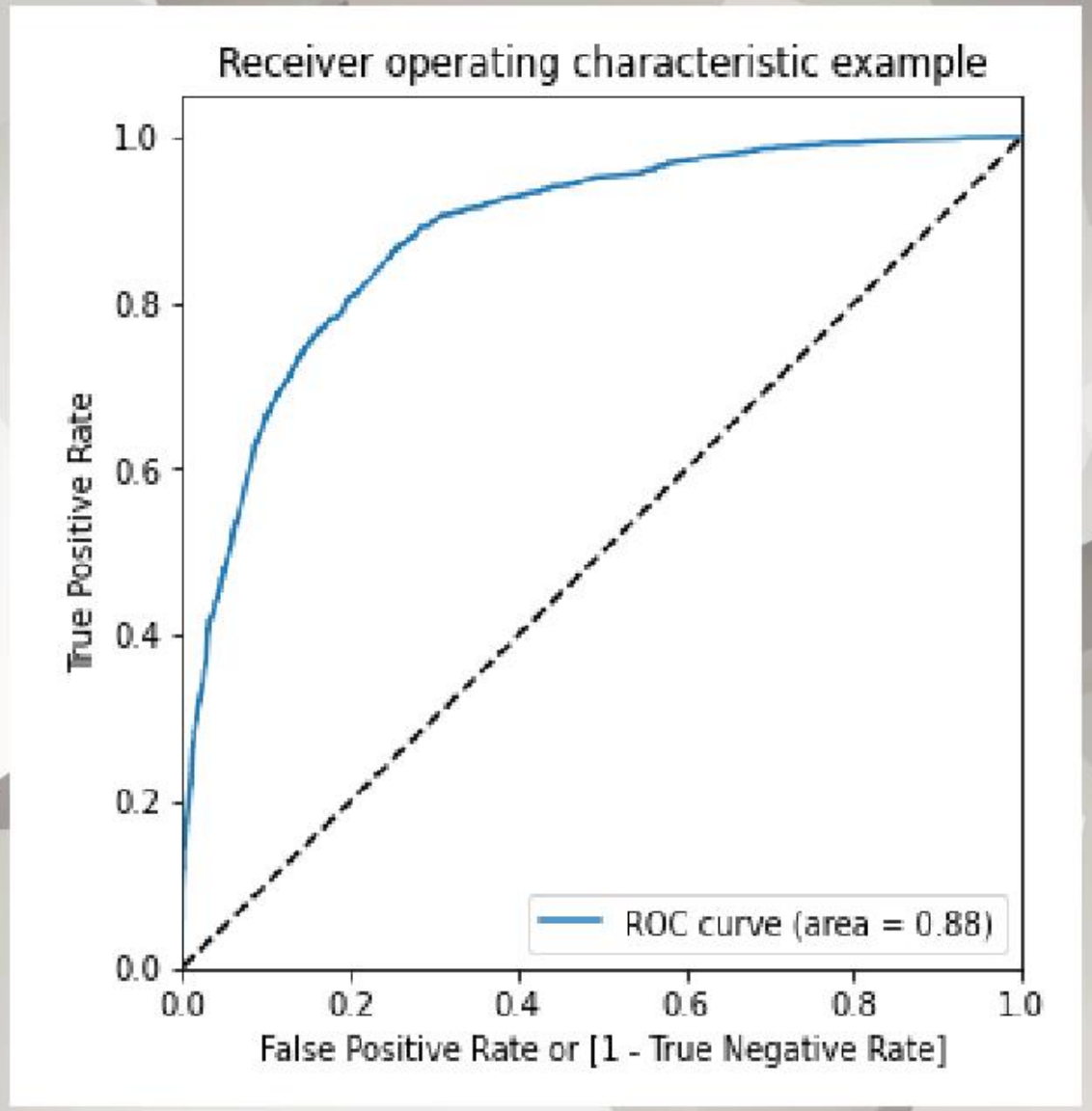
Dep. Variable:	Converted	No. Observations:	6351
Model:	GLM	Df Residuals:	6337
Model Family:	Binomial	Df Model:	13
Link Function:	logit	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-2651.3
Date:	Fri, 21 May 2021	Deviance:	5302.6
Time:	01:21:25	Pearson chi2:	6.50e+03
No. Iterations:	7		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-3.4533	0.113	-30.579	0.000	-3.675	-3.232
TotalVisits	5.5427	1.444	3.838	0.000	2.712	8.373
Total Time Spent on Website	4.6048	0.166	27.690	0.000	4.279	4.931
Lead Origin_Lead Add Form	3.7501	0.225	16.651	0.000	3.309	4.192
Lead Source_Olark Chat	1.5802	0.111	14.187	0.000	1.362	1.798
Lead Source_Welingak Website	2.5821	1.033	2.500	0.012	0.558	4.607
Do Not Email_Yes	-1.4360	0.170	-8.437	0.000	-1.770	-1.102
Last Activity_Olark Chat Conversation	-1.3974	0.167	-8.348	0.000	-1.725	-1.069
Last Activity_SMS Sent	1.2672	0.074	17.164	0.000	1.123	1.412
What is your current occupation_Other	2.1567	0.755	2.857	0.004	0.677	3.636
What is your current occupation_Student	1.2456	0.226	5.502	0.000	0.802	1.689
What is your current occupation_Unemployed	1.1632	0.086	13.582	0.000	0.995	1.331
What is your current occupation_Working Professional	3.6797	0.204	18.008	0.000	3.279	4.080
Last Notable Activity_Unreachable	1.8153	0.601	3.022	0.003	0.638	2.993

Predicting the Conversion Probability and Predicted column

Creating new column 'predicted' with 1 if Conversion Probability > 0.5 else 0

	Converted	Conversion_Prob	LeadID	Predicted	Lead_Score
0	1	0.647883	1289	1	64
1	0	0.133180	3604	0	13
2	0	0.232946	5584	0	23
3	0	0.133180	7679	0	13
4	0	0.495090	7563	0	49



Plotting ROC and AUC

- **ROC : Receiver Operating Characteristics:** It shows the trade-off between sensitivity and specificity (any increase In sensitivity will be accompanied by a decrease in specificity).
- **AUC : Area under the Curve :** By determining the Area under the curve (AUC) of the ROC curve, the goodness of the model is determined. Since the ROC curve is more towards the upper-left corner of the graph, it means that the model is very good. The larger the AUC, the better will is the model.

	Converted	LeadID	Conversion_Prob	final_predicted	Lead_Score
0	0	8308	0.456551	1	45
1	1	7212	0.839834	1	83
2	1	2085	0.982741	1	98
3	1	4048	0.878240	1	87
4	0	4790	0.108266	0	10

Evaluating the Model on Train Dataset

- The final model on the train dataset is used to make predictions for the test dataset
- The train data set was scaled using the scaler.transform function that was used to scale the train dataset.
- The Predicted probabilities were added to the leads in the test dataframe.
- Using the probability threshold value of 0.335, the leads from the test dataset were predicted if they will convert or not.

Evaluating the Model on the Test Set

```
# Creating confusion matrix
confusion2 = metrics.confusion_matrix(y_pred_final['Converted'], y_pred_final.final_predicted )
confusion2
```

```
array([[1378,  365],
       [ 174,  805]], dtype=int64)
```

```
# Calculating the sensitivity
TP/(TP+FN)
```

```
0.822267620020429
```

```
# Calculating the specificity
TN/(TN+FP)
```

```
0.7901376146788991
```

Evaluating the Model on the Test Set

```
# Creating confusion matrix
confusion2 = metrics.confusion_matrix(y_pred_final['Converted'], y_pred_final.final_predicted )
confusion2
```

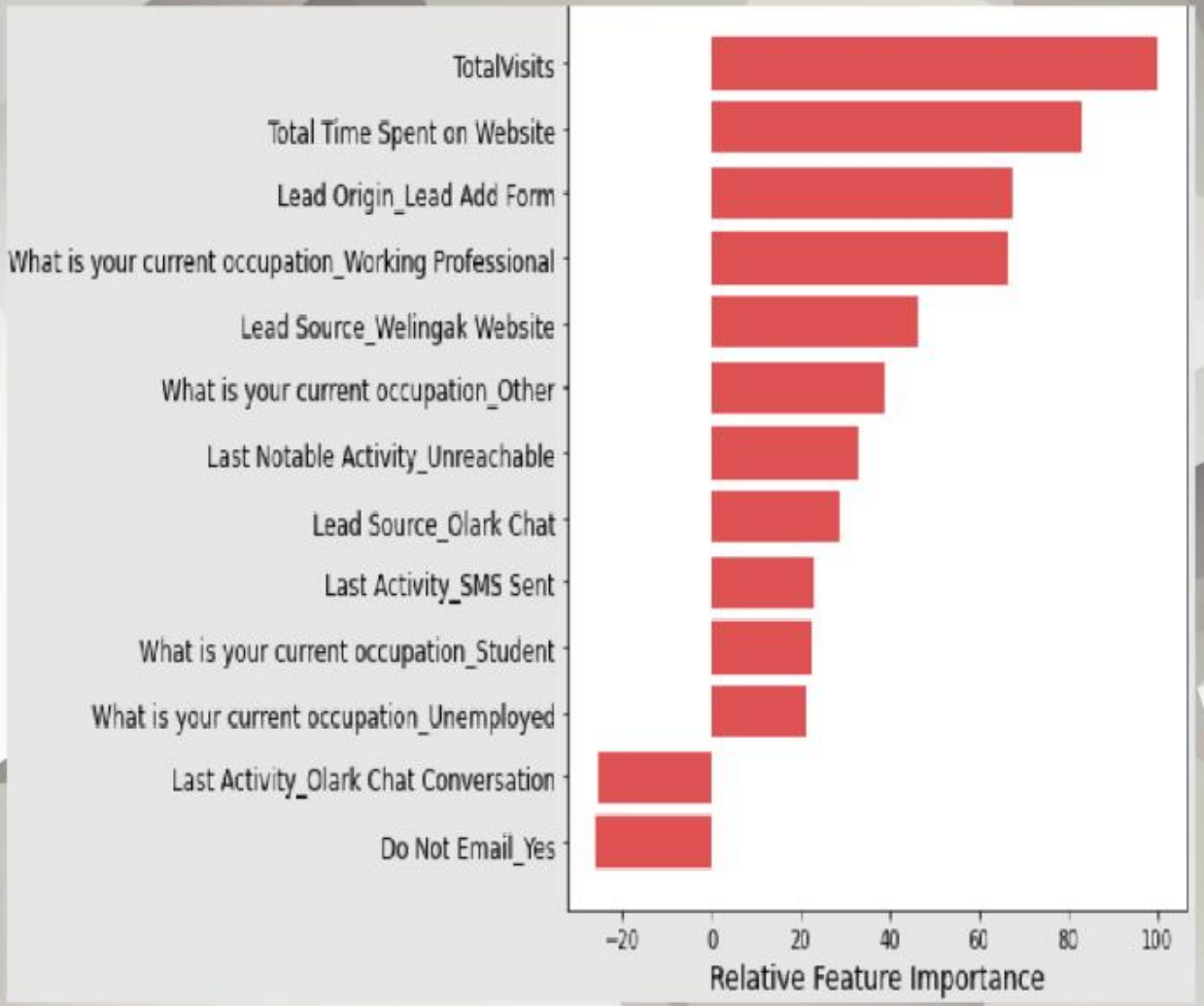
```
array([[1378,  365],
       [ 174,  805]], dtype=int64)
```

```
# Calculating the sensitivity
TP/(TP+FN)
```

```
0.822267620020429
```

```
# Calculating the specificity
TN/(TN+FP)
```

```
0.7901376146788991
```



Feature Determination

- The Coefficient (beta) values for each of these features from the model parameters are used to determine the order of importance of these features.
- Features with high positive beta values are the ones that contribute most towards the probability of a lead getting converted.
- Similarly, features with high negative beta values contribute the least

THANK YOU