Lead Scoring Case Study with Logistic Regression

Approach in Nutshell:

1. Importing Data, Inspecting the Dataframe
2. Data Preparation (Encoding Categorical Variables, Handling Null Values)
3. EDA (univariate analysis, outlier detection, checking data imbalance)
4. Dummy Variable Creation
5. Test-Train Split
6. Feature Scaling
7. Looking at Correlations
8. Model Building (Feature Selection Using RFE, Improvising the model further inspecting adjusted R-squared, VIF and p-vales)
9. Build final model
10. Model evaluation with different metrics Sensitivity, Specificity

Importing the data and Inspecting the dataframe:

Prospect ID	9240
Lead Number	9240
Lead Origin	5
Lead Source	21
Do Not Email	2
Do Not Call	2
Converted	2
TotalVisits	41
Total Time Spent on Website	1731
Page Views Per Visit	114
Last Activity	17
Country	38
Specialization	19
How did you hear about X Education	10
What is your current occupation	6
What matters most to you in choosing a course	3
Search	2
Magazine	1
Newspaper Article	2
X Education Forums	2
Newspaper	2
Digital Advertisement	2
Through Recommendations	2
Receive More Updates About Our Courses	1
Tags	26
Lead Quality	5
Update me on Supply Chain Content	1
Get updates on DM Content	1
Lead Profile	6
City	7
Asymmetrique Activity Index	3
Asymmetrique Profile Index	3
Asymmetrique Activity Score	12
Asymmetrique Profile Score	10
I agree to pay the amount through cheque	1
A free copy of Mastering The Interview	2

- We are using Leads.csv dataset for this case study.
- First we imported modules like Pandas, NumPy for performing different operations on the dataset and SKLearn, Statsmodels etc. for building for selecting important features, building Logistic Regression Model and evaluating final model's performance.

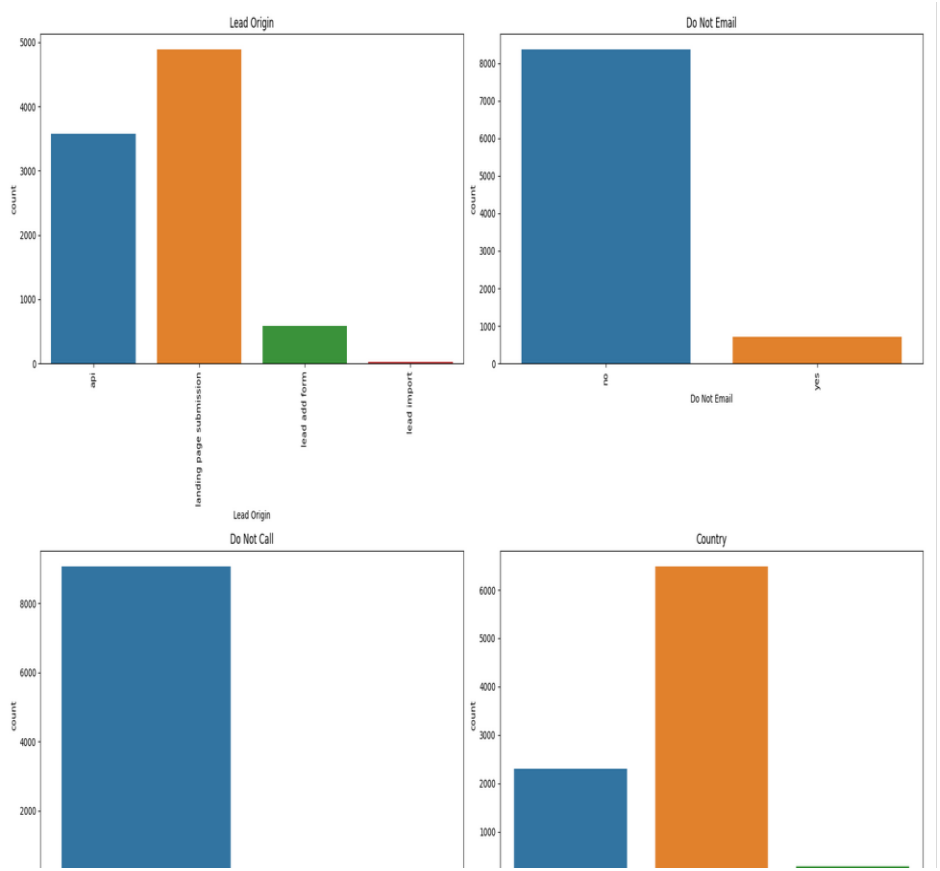
2. Data Preparation : Missing Value Handling

Prospect ID	0.00
Lead Number	0.00
Lead Origin	0.00
Lead Source	0.39
Do Not Email	0.00
Do Not Call	0.00
Converted	0.00
TotalVisits	1.48
Total Time Spent on Website	0.00
Page Views Per Visit	1.48
Last Activity	1.11
Country	26.63
Specialization	36.58
How did you hear about X Education	78.46
What is your current occupation	29.11
What matters most to you in choosing a course	29.32
Search	0.00
Newspaper Article	0.00
X Education Forums	0.00
Newspaper	0.00
Digital Advertisement	0.00
Through Recommendations	0.00
Tags	36.29
Lead Quality	51.59
Lead Profile	74.19
City	39.71
Asymmetrique Activity Index	45.65
Asymmetrique Profile Index	45.65
Asymmetrique Activity Score	45.65
Asymmetrique Profile Score	45.65
A free copy of Mastering The Interview	0.00
Last Notable Activity	0.00

dtype: float64

- There are some categorical features having a label as “SELECT”. This means the person might not have selected any value for that field. Hence this is as good as a missing value. So converting SELECT into the NaN
- After identifying all the missing data, dropped columns having more than 70% null values
- As the Lead Quality depends upon the intuition of the employee, it will be safer to update the NaN to “Not Sure”
- There are too many variations in the columns (‘Asymmetrique Activity Index’,’Asymmetrique Activity Score’,’Asymmetrique Profile Index’,’Asymmetrique Profile Score’) and it is not safer to impute any values in the columns and hence we will drop these columns with very high percentage of missing data
- We can impute the MUMBAI into all the NULLs as most of the values belong to MUMBAI

3. EDA (univariate analysis, outlier detection, checking data imbalance)



- **OBSERVATION:**
- API and Landing Page Submission has less conversion rate(~30%) but counts of the leads from them are considerable
- The count of leads from the Lead Add Form is pretty low but the conversion rate is very high
- Lead Import has very less count as well as conversion rate and hence can be ignored
- **To improve the overall lead conversion rate, we need to focus on increasing the conversion rate of 'API' and 'Landing Page Submission' and also increasing the number of leads from 'Lead Add Form'**

4. Dummy Variable Creation

As logistic regression can work with numeric data only, creating dummy variables for the categorical columns.

	TotalVisits	Total Time Spent on Website	Lead Origin_lead add form	Lead Source_direct traffic	Lead Source_google	Lead Source_organic search	Lead Source_welingak website	Do Not Email_yes	Last Activity_olark chat conversation	Last Activity_sms sent	What is your current occupation_housewife	c
1289	0.014184	0.612676	0	0	1	0	0	0	0	0	0	
3604	0.000000	0.000000	0	0	0	0	0	0	0	0	0	
5584	0.042553	0.751761	0	0	0	1	0	1	0	0	0	
7679	0.000000	0.000000	0	0	0	0	0	0	0	0	0	
7563	0.014184	0.787852	0	0	0	1	0	1	0	0	0	

5. Splitting Data into Training and Test set

Next, the dataset was split into training and test set, to train model first with a chunk of data and then evaluate its performance on unseen data.

[601]:

Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	6351
Model:	GLM	**Df Residuals:**	6335
Model Family:	Binomial	**Df Model:**	15
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2741.3
Date:	Mon, 11 Nov 2024	**Deviance:**	5482.6
Time:	15:54:43	**Pearson chi2:**	6.64e+03
No. Iterations:	22	**Pseudo R-squ. (CS):**	0.3758
Covariance Type:	nonrobust		

6. Feature Scaling

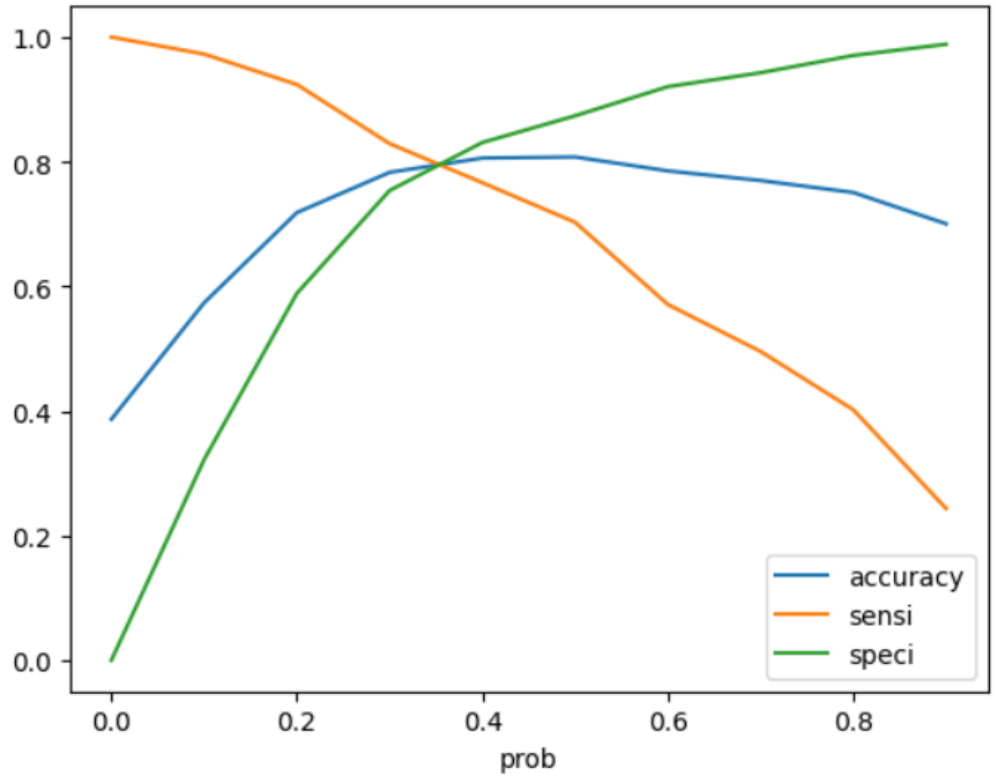
Feature Scaling is required before Logistic Regression to bring all the features in same scale, this ensures that features with high magnitude are not given higher importance by Logistic Regression Model.

]:

	Features	VIF
1	Total Time Spent on Website	2.34
0	TotalVisits	2.28
4	Lead Source_google	2.04
3	Lead Source_direct traffic	1.91
5	Lead Source_organic search	1.60
9	Last Activity_sms sent	1.49
2	Lead Origin_lead add form	1.47
6	Lead Source_welingak website	1.31
11	What is your current occupation_working profes...	1.17
7	Do Not Email_yes	1.10
8	Last Activity_olark chat conversation	1.02
10	What is your current occupation_other	1.01
12	Last Notable Activity_unreachable	1.01

7. Checking Correlation

Since Logistic Regression Model is high affected by multi-collinearity, removing the features showing high correlation.



8. Model Building (Feature Selection Using RFE, Improvising the model further inspecting adjusted R-squared, VIF and p-vales)

Next, we are using RFE to obtain top 15 features to begin with.

After that, manually inspecting p-values and VIF to improve the model even further

	coef	std err	z	P>|z|	[0.025	0.975]
const	-1.2461	0.081	-15.396	0.000	-1.405	-1.087
TotalVisits	4.6490	1.403	3.314	0.001	1.899	7.399
Total Time Spent on Website	4.5480	0.162	28.098	0.000	4.231	4.865
Lead Origin_lead add form	2.6841	0.224	11.957	0.000	2.244	3.124
Lead Source_direct traffic	-1.4736	0.114	-12.954	0.000	-1.697	-1.251
Lead Source_google	-1.1551	0.109	-10.580	0.000	-1.369	-0.941
Lead Source_organic search	-1.2633	0.134	-9.426	0.000	-1.526	-1.001
Lead Source_welingak website	2.5921	1.033	2.509	0.012	0.567	4.617
Do Not Email_yes	-1.4146	0.168	-8.437	0.000	-1.743	-1.086
Last Activity_olark chat conversation	-1.4765	0.165	-8.932	0.000	-1.800	-1.152
Last Activity_sms sent	1.3072	0.072	18.070	0.000	1.165	1.449
What is your current occupation_other	1.4003	0.760	1.842	0.066	-0.090	2.890
What is your current occupation_working professional	2.7968	0.193	14.467	0.000	2.418	3.176
Last Notable Activity_unreachable	1.6871	0.610	2.766	0.006	0.492	2.883

9 and 10. Final Model and Model Evaluation

Now that we have the final set of features obtained by removing highly collinear ones, using RFE, inspecting p-values and VIF- we can build the final logistic regression model and evaluate its performance.

[658]:
```
# Check the overall accuracy
metrics.accuracy_score(y_pred_final['Converted'], y_pred_final.final_predicted)
```

[658]:
```
0.808666911494675
```

[659]:
```
# Creating confusion matrix
confusion2 = metrics.confusion_matrix(y_pred_final['Converted'], y_pred_final.final_predicted )
confusion2
```

[659]:
```
array([[1464,  280],
       [ 241,  738]], dtype=int64)
```

[660]:
```
# Substituting the value of true positive
TP = confusion2[1,1]
# Substituting the value of true negatives
TN = confusion2[0,0]
# Substituting the value of false positives
FP = confusion2[0,1]
# Substituting the value of false negatives
FN = confusion2[1,0]
```

[661]:
```
# Precision = TP / TP + FP
TP / (TP + FP)
```

[661]:
```
0.724950884086444
```

[662]:
```
#Recall = TP / TP + FN
TP / (TP + FN)
```

Plotting the ROC Curve

- shows tradeoff between sensitivity and specificity (increase in one will cause decrease in other).
- The closer the curve follows the y-axis and then the top border of the ROC space, means more area under the curve and the more accurate the test.
- The closer the curve comes to the 45-degree diagonal of the ROC space i.e. the reference line, means less area and the less accurate is the test.