UPGRAD LEAD SCORE CASE STUDY

BY:

VINAYAK RANE

AMAR RANA

RAMYA

CONTENTS

PROBLEM STATEMENT

- X Education sells online courses to its customers
- Company wants to increase the number of leads to join the courses
- Company is looking to smoothen the process of identifying potential leads (Hot leads)
- Company wishes to call only those leads who are potentially hot leads and hence needs to save time for other productive task

Business Objective

- Lead wants to 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.
- The CEO, in particular, has given a ballpark of the target lead conversion rate to be around 80%
- the company reaches its target for a quarter before the deadline. During this time, the company wants the sales team to focus on some new work as well. So during this time, the company's aim is to not make phone calls unless it's extremely necessary, i.e. they want to minimize the rate of useless phone calls.

METHODOLOGY FOR MODEL PREPARATION

- Data Cleaning, imputing and understanding the data
- To check null values , ‘Select’ data and to find a solution to deal with such values
- To check outliers in the data
- Exploratory data Analysis
- Creation of Dummy variables for categorical columns
- Scaling of numerical variables
- Building Logistic Regression Model
- Model evaluation using confusion matrix, precision, recall, specificity.

DATA CLEANING & IMPUTATION

Total columns at initial = 37

Columns such as ‘City’, ‘Country’, ’Prospect Id’, ’Lead number’ are eliminated as there serve no enhancement in analysis

Eliminating all the ‘Asymmetric’ features as these contain more than 50% of null values

Reducing the data by removing all the rows which contain the ‘Select’ values in columns such as ‘Lead Profile’, ’Specialization’ and 'How did you hear about X Education'

Imbalance Ratio (convert_0/convert_1) = 0.96

At the end we left with 12 columns and 4535 rows for EDA

Exploratory Data Analysis (EDA)

- ‘TotalVisits’ has high co-relation with ‘Page Views Per Visit’
- ‘Total time spent on Website’ has a direct correlation with ‘Converted’ which is a target column

In [36]:
```
lead_data.corr()
```
Out[36]:

	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit
Converted	1.000000	-0.002933	0.336092	-0.098751
TotalVisits	-0.002933	1.000000	0.113488	0.407609
Total Time Spent on Website	0.336092	0.113488	1.000000	0.186492
Page Views Per Visit	-0.098751	0.407609	0.186492	1.000000

In [37]:
```
#Checking the correlation among varibles
plt.figure(figsize=(10,8))
sns.heatmap(lead_data.corr(),annot = True)
plt.show()
```

Exploratory Data Analysis (EDA) - Conti

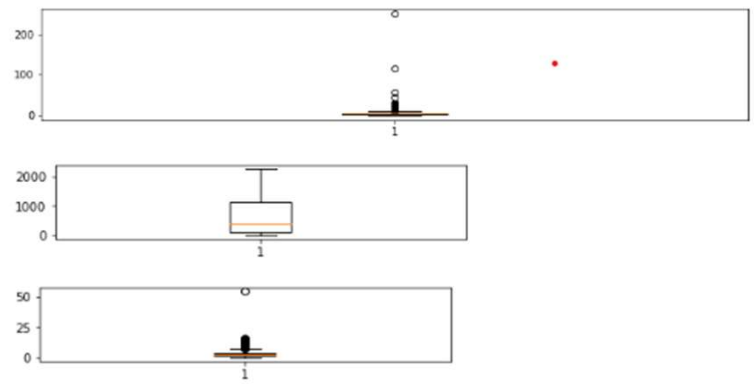
In the boxplots,
we can see there are not much of the outliers in the numerical cols which can affect our observations

```python
#checking the outliers
plt.figure(figsize = (12,7))

plt.subplot(3,1,1)
plt.boxplot(x = 'TotalVisits', data = lead_data)
plt.show()

plt.subplot(3,1,2)
plt.boxplot(x = 'Total Time Spent on Website', data = lead_data)
plt.show()

plt.subplot(3,1,3)
plt.boxplot(x = 'Page Views Per Visit', data = lead_data)
plt.show()
```



DUMMY VARIABLES SELECTION

Following are the categorical variable which are considered for creating dummy variables

- 'Lead Origin’,
- 'Lead Source’,
- 'Do Not Email’,
- 'Last Activity’,
- 'Specialization’,
- 'What is your current occupation’,
- 'A free copy of Mastering The Interview’,
- 'Last Notable Activity'

TRAIN TEST SPLIT

The data is split in the ratio of 70 (Train) to 30 (test)

- Train data rows in total : 3174
- Test data rows in total : 1361

```
print(f"X_train shape {X_train.shape}\n")
print(f"X_test shape {X_test.shape}\n")
print(f"y_train shape {y_train.shape}\n")
print(f"y_test shape {y_test.shape}\n")
```

```
X_train shape (3174, 72)

X_test shape (1361, 72)

y_train shape (3174,)

y_test shape (1361,)
```

SCALING

Below are the numerical columns selected for scaling.

- 'TotalVisits'
- 'Total Time Spent on Website'
- 'Page Views Per Visit'

BEFORE

```
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	Lead Source_Direct Traffic	Sourc
2006	14.0	255	7.00	1	0	0	0	
5140	5.0	12	1.67	1	0	0	0	
7588	4.0	30	4.00	1	0	0	1	
5244	6.0	158	3.00	1	0	0	1	
8663	11.0	190	3.67	1	0	0	0	

AFTER

```
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	Lead Source_Direct Traffic	S
2006	1.604339	-0.648184	1.845831	1	0	0	0	
5140	0.111763	-1.076675	-0.588172	1	0	0	0	
7588	-0.054079	-1.044935	0.475848	1	0	0	1	
5244	0.277605	-0.819228	0.019187	1	0	0	1	
8663	1.106814	-0.762801	0.325150	1	0	0	0	

5 rows × 72 columns

MODEL BUILDING

- Model is build using Logistic Regression classification technique
- Columns are eliminated using Recursive Feature Elimination (RFE)
- Variance Inflation Factor and p-values are considered for further manual elimination of the columns
- Max limit for VIF is 5 and for p-value is 0.005
- Separate individual function for logistic model and Variance inflation Factor are written for the reusability
- Recursively perform RFE and VIF to get best feature at the end for building model

MODEL BUILDING - Conti

Generalized Linear Model Regression Results

Dep. Variable:	Converted	No. Observations:	3174
Model:	GLM	Df Residuals:	3165
Model Family:	Gaussian	Df Model:	8
Link Function:	identity	Scale:	0.15701
Method:	IRLS	Log-Likelihood:	-1561.0
Date:	Sun, 09 Jul 2023	Deviance:	496.95
Time:	22:17:29	Pearson chi2:	497.
No. Iterations:	3	Pseudo R-squ. (CS):	0.4481
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	0.5836	0.021	28.398	0.000	0.543	0.624
Total Time Spent on Website	0.1904	0.007	25.717	0.000	0.176	0.205
Lead Origin_Landing Page Submission	-0.2183	0.021	-10.359	0.000	-0.260	-0.177
Lead Source_Reference	0.3483	0.034	10.203	0.000	0.281	0.415
Lead Source_Welingak Website	0.5140	0.134	3.844	0.000	0.252	0.776
Do Not Email_Yes	-0.2320	0.027	-8.460	0.000	-0.286	-0.178
Last Activity_Converted to Lead	-0.1983	0.035	-5.608	0.000	-0.268	-0.129
Last Activity_SMS Sent	0.1537	0.015	10.184	0.000	0.124	0.183
What is your current occupation_Working Professional	0.3210	0.021	15.350	0.000	0.280	0.362

Out[97]:

	Features	VIF
1	Lead Origin_Landing Page Submission	1.596276
6	Last Activity_SMS Sent	1.524193
2	Lead Source_Reference	1.230836
7	What is your current occupation_Working Profes...	1.225357
0	Total Time Spent on Website	1.107498
4	Do Not Email_Yes	1.099861
5	Last Activity_Converted to Lead	1.074806
3	Lead Source_Welingak Website	1.003478

VIF values are pretty good. P values are also low

MODEL BUILDING - Conti

Prediction with cut off at 0.45 of final model is as below:

Out[101]:

	actual_lead_converted	Probability_of_conversion	predict_lead_converted
0	0	0.241891	0
1	0	0.160309	0
2	1	0.320020	0
3	0	0.362993	0
4	0	-0.011886	0
...	...	...	...
3169	1	0.438867	0
3170	0	0.513158	1
3171	1	0.650134	1
3172	1	0.464813	1
3173	1	1.043938	1

3174 rows × 3 columns

MODEL EVALUATION : Train data

Different measures are used to evaluate the model which includes

- Confusion Matri

```
#Lets check the confusin Matrix
conf_matrix = metrics.confusion_matrix(y_train_pred_df['actual_lead_converted'],
conf_matrix
```

```
array([[1230,  323],
       [ 296, 1325]], dtype=int64)
```

- Accuracy > ~ 80% which is quite good.

```
In [103]: #Accuracy Measure
          metrics.accuracy_score(y_train_pred_df['actual_lead_converted'],y_train_pred_df[

Out[103]: 0.8049779458097038
```

MODEL EVALUATION : Train data - Conti

- Sensitivity > ~ 81%
- Precision > ~ 80%
- Specificity > ~79%
- Recall > ~ 81%

```
In [104]: #calculating the sensitivity and specificity
          TP = conf_matrix[1,1]
          TN = conf_matrix[0,0]
          FP = conf_matrix[0,1]
          FN = conf_matrix[1,0]

In [105]: sensitivity = TP/(TP+FN)
          sensitivity

Out[105]: 0.8173966687230105

In [106]: specificity = TN/(TN+ FP)
          specificity

Out[106]: 0.7920154539600772

In [107]: precision = TP/(TP+FP)
          precision

Out[107]: 0.804004854368932

In [108]: recall = TP/(TP+FN)
          recall

Out[108]: 0.8173966687230105
```

We have got quite good values for sensitivity and specificity for threshold cut off at 0.45 with 80% of accuracy in train data which is quite good.

MODEL EVALUATION : Test data

Different measures are used to evaluate the model which includes

- Confusion Matri

```
In [118]: #confusion Matrix
          conf_matrix_test = metrics.confusion_matrix(y_test_pred_df['Actual_lead_converte
          conf_matrix_test

Out[118]: array([[503, 163],
                 [140, 555]], dtype=int64)
```

- Accuracy > ~ 77% which is quite good.

```
In [119]: #checking the accuracy of test data
          metrics.accuracy_score(y_test_pred_df['Actual_lead_converted'],y_test_pred_df['F

Out[119]: 0.7773695811903012
```

MODEL EVALUATION : Train data - Conti

- Sensitivity > ~ 81%
- Precision > ~ 80%
- Specificity > ~79%
- Recall > ~ 81%

```
In [120]: #calculating sensitivity and specificity
          TP = conf_matrix_test[1,1]
          TN = conf_matrix_test[0,0]
          FP = conf_matrix_test[0,1]
          FN = conf_matrix_test[1,0]

In [121]: #sensitivity
          sensitivity_test = TP/(TP+FN)
          sensitivity_test

Out[121]: 0.7985611510791367

In [122]: #specificity
          specificity_test = TN/(TN+FP)
          specificity_test

Out[122]: 0.7552552552552553

In [123]: precision = TP/(TP+FP)
          precision

Out[123]: 0.7729805013927576

In [124]: recall = TP/(TP+FN)
          recall

Out[124]: 0.7985611510791367
```

we obtained sensitivity of 80% and specificity of 75% with current logistic regression model which is quite satisfactory with 77.78% of accuracy.

Business recommendation & Conclusion

X Education can make use of the following points in order to convert their leads into successful leads:

- It is observed that those who working professionals are more prone to opt for the courses so business should focus on working professionals for lead
- Those who visits the website and spend considerable amount of time there, can be approached to convert them into successful leads
- Also, those are coming from source such as reference and Welingak website can also be taken into consideration for the successful lead, this applications should be treated as hot leads.
- Business can max the number of leads generated by direct traffic and google
- By closely looking at the data when the last activity is converted to Lead, then there are high chances of them getting converted into successful leads