

UPGRAD LEAD SCORE CASE STUDY

BY:

VINAYAK RANE

AMAR RANA

RAMYA SHANKAR

A solid orange horizontal bar spanning the width of the slide at the bottom.

CONTENTS

- Problem Statement
- Business Objective
- Methodology for model preparation
- Data Cleaning/Imputation
- Exploratory Data Analysis
- Dummy Variables selection
- Train-Test Split
- Model Building
- Model Evaluation – Specificity/Sensitivity/Precision/Recall
- Business recommendation

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'

At the end we left with 12 columns and 6373 rows for EDA

Exploratory Data Analysis (EDA)

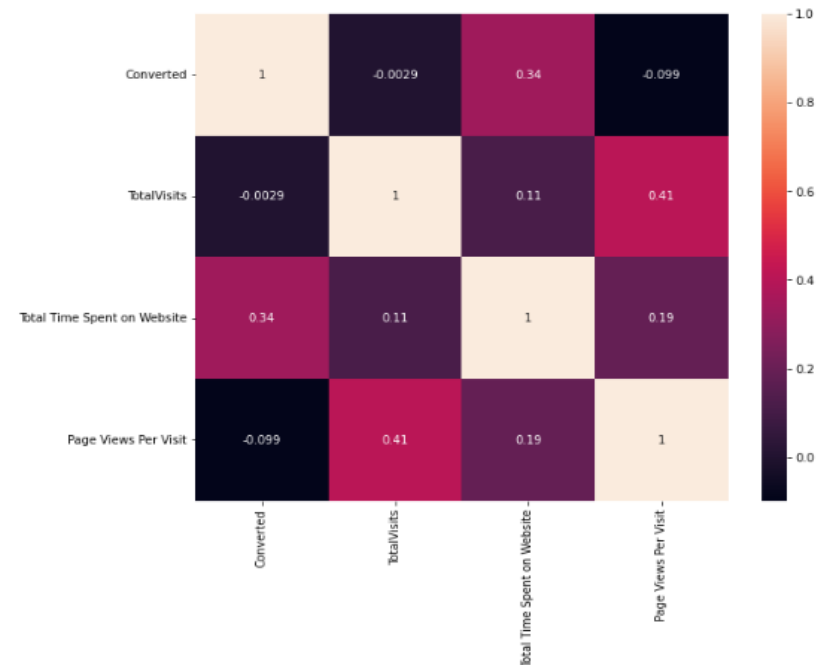
- 'TotalVisits' has high co-relation with 'Page Views Per Visit'
- 'Total time spent on Website' has a direct correlation with 'Converted' which is a target column

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

```
Out[36]:
```

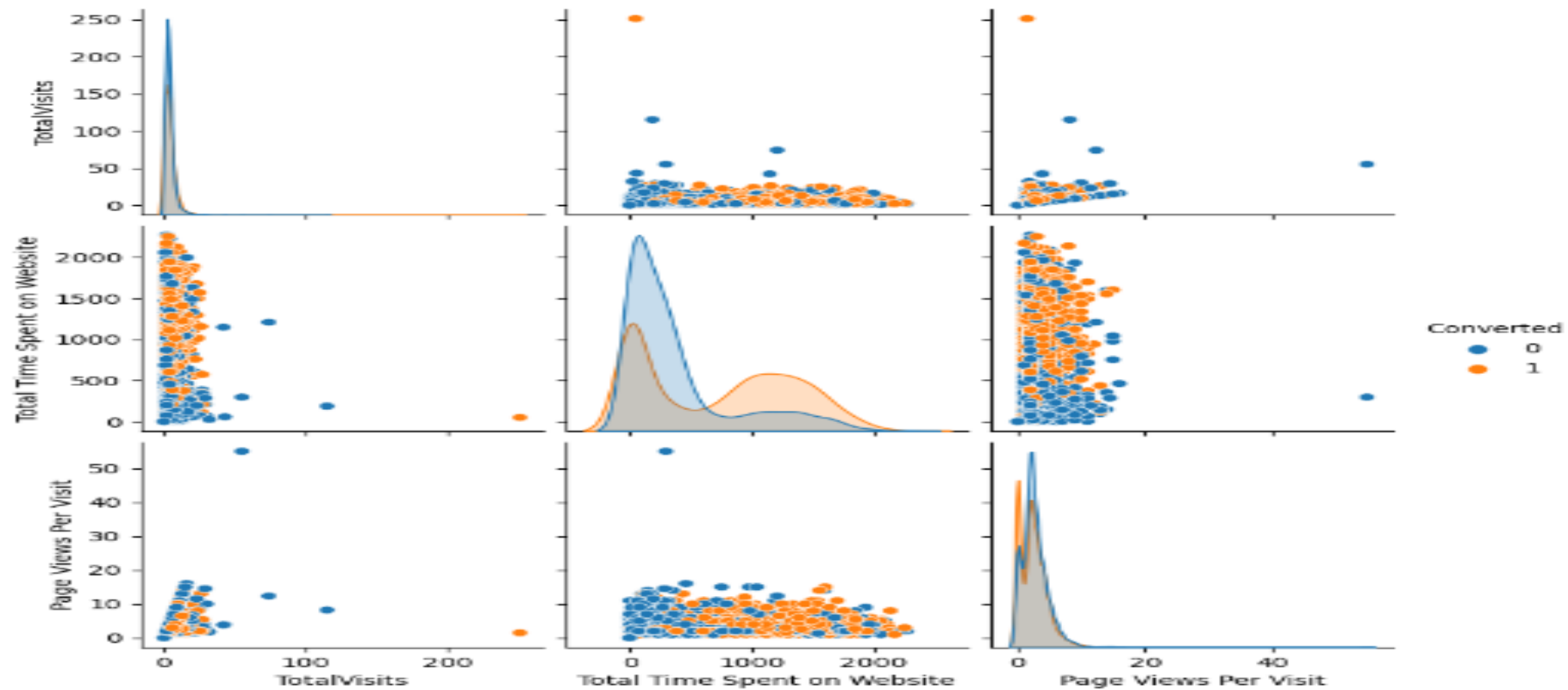
	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit
Converted	1.000000	-0.002933	0.336082	-0.098751
TotalVisits	-0.002933	1.000000	0.113488	0.407609
Total Time Spent on Website	0.336082	0.113488	1.000000	0.186492
Page Views Per Visit	-0.098751	0.407609	0.186492	1.000000

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



Exploratory Data Analysis (EDA) - Conti

```
xleads = leads_df[['TotalVisits', 'Total Time Spent on Website', 'Page Views Per Visit', 'Converted']]
sns.pairplot(xleads, diag_kind='kde', hue='Converted')
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	Origin_Landing Page Submission	Lead Origin_Lead Add Form	Lead Origin_Lead Import	Lead Source_Direct Traffic	Source
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	

5 rows x 9 columns

AFTER

```
x_train.head()
```

	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Origin_Landing Page Submission	Lead Origin_Lead Add Form	Lead Origin_Lead Import	Lead Source_Direct Traffic	Source
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 x 9 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 0.05
- 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:	4461
Model:	GLM	Df Residuals:	4449
Model Family:	Binomial	Df Model:	11
Link Function:	Logit	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-2079.1
Date:	Mon, 17 Jul 2023	Deviance:	4158.1
Time:	12:21:54	Pearson chi2:	4.80e+03
No. Iterations:	7	Pseudo R-squ. (CS):	0.3642
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	0.2040	0.196	1.043	0.297	-0.179	0.587
TotalVisits	11.1489	2.665	4.184	0.000	5.926	16.371
Total Time Spent on Website	4.4223	0.185	23.899	0.000	4.060	4.785
Lead Origin_Lead Add Form	4.2051	0.258	16.275	0.000	3.699	4.712
Lead Source_Olark Chat	1.4526	0.122	11.934	0.000	1.214	1.691
Lead Source_Welingak Website	2.1526	1.037	2.076	0.038	0.121	4.185
Do Not Email_Yes	-1.5037	0.193	-7.774	0.000	-1.883	-1.125
Last Activity_Had a Phone Conversation	2.7552	0.802	3.438	0.001	1.184	4.326
Last Activity_SMS Sent	1.1856	0.082	14.421	0.000	1.024	1.347
What is your current occupation_Student	-2.3578	0.281	-8.392	0.000	-2.908	-1.807
What is your current occupation_Unemployed	-2.5445	0.186	-13.699	0.000	-2.908	-2.180
Last Notable Activity_Unreachable	2.7846	0.807	3.449	0.001	1.202	4.367

	Features	VIF
9	What is your current occupation_Unemployed	2.82
1	Total Time Spent on Website	2.00
0	TotalVisits	1.54
7	Last Activity_SMS Sent	1.51
2	Lead Origin_Lead Add Form	1.45
3	Lead Source_Olark Chat	1.33
4	Lead Source_Welingak Website	1.30
5	Do Not Email_Yes	1.08
8	What is your current occupation_Student	1.06
6	Last Activity_Had a Phone Conversation	1.01
10	Last Notable Activity_Unreachable	1.01

MODEL EVALUATION : Train data

Different measures are used to evaluate the model which includes

- Confusion Matri

```
confusion2 = metrics.confusion_matrix(y_train_pred_final.Converted, y_train_pred_final.final_predicted )
confusion2
```

```
|: array([[1823,  489],
        [ 444, 1705]], dtype=int64)
```

- Accuracy > ~ 79% which is quite good.

```
# Let's check the accuracy now
```

```
metrics.accuracy_score(y_train_pred_final.Converted, y_train_pred_final.final_predicted)
```

```
0.7908540685944856
```

MODEL EVALUATION : Train data - Conti

- Sensitivity > ~ 77%
- Precision > ~ 78%
- Specificity > ~78%
- Recall > ~ 77%

```
# Let's evaluate the other metrics as well
```

```
TP = confusion2[1,1] # true positive  
TN = confusion2[0,0] # true negatives  
FP = confusion2[0,1] # false positives  
FN = confusion2[1,0] # false negatives
```

```
# Calculate Precision
```

```
TP / (TP+FP)
```

```
0.784037558685446
```

```
# Calculate Recall
```

```
TP / (TP+FN)
```

```
0.7771056305258259
```

```
# Calculate sensitivity
```

```
TP / float(TP+FN)
```

```
0.7794759825327511
```

```
# Calculate specificity
```

```
TN / float(TN+FP)
```

```
0.7891566265060241
```

We have got quite good values for sensitivity and specificity for threshold cut off at 0.42 with 79% 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

```
confusion2 = metrics.confusion_matrix(y_pred_final['Converted'], y_pred_final.final_predicted )
confusion2
array([[801, 195],
       [213, 703]], dtype=int64)
```

- Accuracy > ~ 78% which is quite good.

```
# Let's check the overall accuracy
metrics.accuracy_score(y_pred_final['Converted'], y_pred_final.final_predicted)
0.7866108786610879
```


MODEL EVALUATION : Train data - Conti

- Sensitivity > ~ 76%
- Precision > ~ 78%
- Specificity > ~80%
- Recall > ~ 76%

```
TP = confusion2[1,1] # true positive  
TN = confusion2[0,0] # true negatives  
FP = confusion2[0,1] # false positives  
FN = confusion2[1,0] # false negatives
```

```
# Calculate Specificity
```

```
TN/(TN+FP)
```

```
0.8042168674698795
```

```
# Calculate Sensitivity
```

```
TP/(TP+FN)
```

```
0.767467248908297
```

```
# Calculate Precision
```

```
TP/(TP+FP)
```

```
0.7828507795100222
```

```
# Calculate Recall
```

```
TP/(TP+FN)
```

```
0.767467248908297
```

we obtained sensitivity of 76% and specificity of 80% with current logistic regression model which is quite satisfactory with 78.66% 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 visit the website and spend considerable amount of time there, can be approached to convert them into successful leads
- Also, those Source Olark Chat and Welingak Website can also be taken into consideration for the successful lead
- Those with Origin_Lead Add Form has chances to get convert into lead
- By closely looking at the data When the last activity Had a Phone Conversation, then there are high chances of them getting converted into successful leads
- It is observed that those who working professionals are more prone to opt for the courses
- Business can max the number of leads generated by direct traffic and google