Lead scoring Case Study

Manaswini / Srivatsa

DSC 43 – April 22 Batch

Problem Statement

- **Business problem**
  X Education needs help to select the most promising leads, i.e. the leads that are most likely to convert into paying customers.

- **Converted to Data Science Problem**
- 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

Analysis Approach

- Explore the provided Dataset (EDA)
    - Identified Missing Data
    - Imputed wherever Appropriate
    - Performed Bivariate Analysis
        - – relating each of the Columns with "Coverted" (Target Variable)
    - Thus Identify the relevant set of Columns for Further usage in Regression
- Prepared Data for Logistic Regression
    - Created Appropriate Dummy variables for Categorical variables
    - Scaling the Numerical variables
- Build the Model
    - Use RFE to identify the set of relevant Columns
    - Recursively Build are refine the model
        - Use VIF and P-Value to eliminate unnecessary Attributes
- Model Evaluation
    - Predicted the Y (Target Variable) – Conversion Probability
    - Determined – Accuracy / Specificity / Sensitivity / Precision & Recall
- Evaluated against the Test Set
- Concluded on the Outcome

EDA – Missing Values

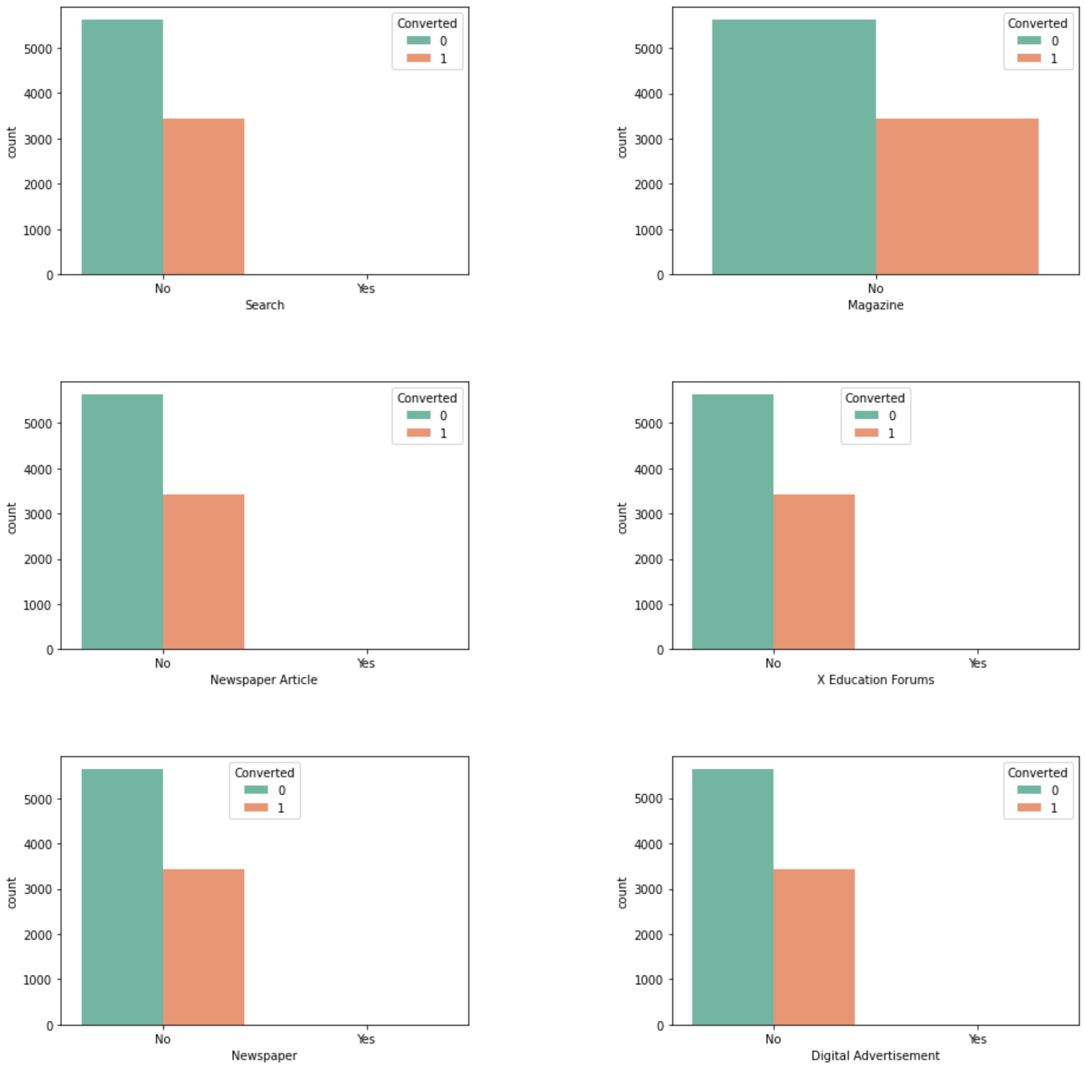
Some Observations

- Initial Data set with 37 Columns and 9240 Rows
- All Columns with greater than 45% of values missing was dropped
- In Column: What matters most to you in choosing a course – Better Career Prospects formed 99% of the values – hence dropped
- In Column: What is your current occupation – 85% was Unemployed – hence used the same for Imputing missing values
- In Column: Country 95% of values was India – hence dropped column
- Replaced All "Select" across table with Null values
- Grouped Missing values for Specialization to Others

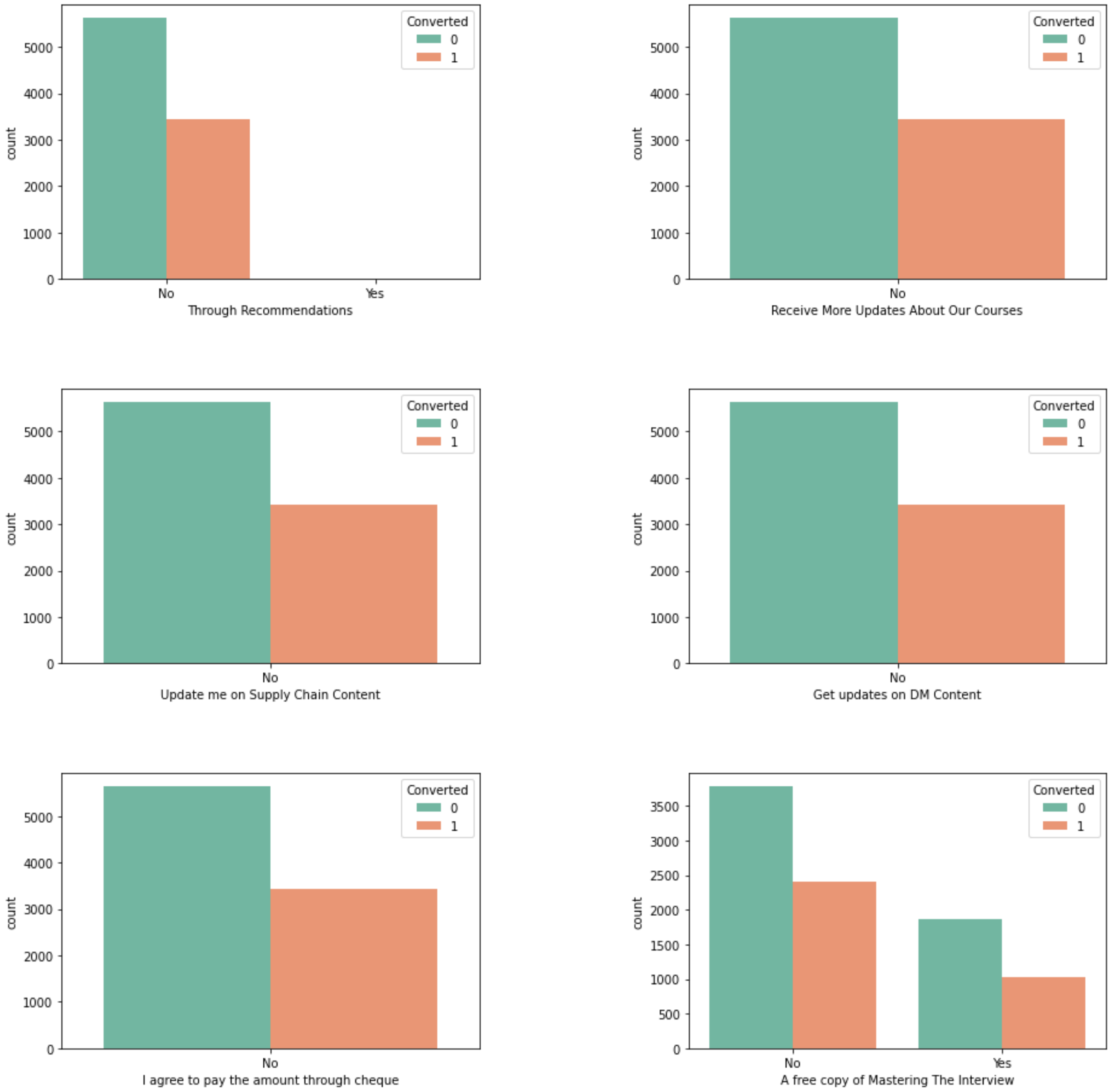
  Imputed Missing City Values with "Mumbai"

All Columns where Data was Skewed were Dropped

All Media Related



Other Columns



Model Building Process

```
from sklearn.linear_model import LogisticRegression
logreg = LogisticRegression()
```

```
from sklearn.feature_selection import RFE
```

```
rfe = RFE (logreg, step=15)
rfe = rfe.fit(X_train, y_train)
```

```
list(zip(X_train.columns, rfe.support_, rfe.ranking_))
```

```
[('Do Not Email', True, 1),
 ('TotalVisits', False, 3),
 ('Total Time Spent on Website', True, 1),
 ('Page Views Per Visit', False, 3),
 ('Lead Origin_Landing Page Submission', True, 1),
 ('Lead Origin_Lead Add Form', True, 1),
 ('Lead Origin_Lead Import', True, 1),
```

	coef	std err	z	P>|z|	[0.025	0.975]
const	1.0655	0.889	1.199	0.231	-0.676	2.807
Do Not Email	-1.6457	0.209	-7.877	0.000	-2.055	-1.236
Total Time Spent on Website	1.1111	0.041	27.039	0.000	1.031	1.192
Lead Origin_Landing Page Submission	-1.1229	0.131	-8.603	0.000	-1.379	-0.867
Lead Origin_Lead Add Form	1.4781	0.894	1.654	0.098	-0.274	3.230
Lead Origin_Lead Import	0.9052	0.477	1.898	0.058	-0.029	1.840
Lead Source_Olark Chat	1.1026	0.125	8.848	0.000	0.858	1.347
Lead Source_Reference	1.8623	0.918	2.029	0.042	0.064	3.661
Lead Source_Welingak Website	4.4162	1.150	3.840	0.000	2.162	6.670
Last Activity_Email Link Clicked	0.6789	0.412	1.649	0.099	-0.128	1.486

```
#Importing stats model package
import statsmodels.api as sm
X_train_sm = sm.add_constant(X_train)
logm1 = sm.GLM(y_train,X_train_sm, family = sm.families.Binor
result = logm1.fit()
result.summary()
```

Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	6351
Model:	GLM	**Df Residuals:**	6318
Model Family:	Binomial	**Df Model:**	32
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2568.9
Date:	Sun, 16 Oct 2022	**Deviance:**	5137.7
Time:	22:51:06	**Pearson chi2:**	6.42e+03
No. Iterations:	21	**Pseudo R-squ. (CS):**	0.4079
Covariance Type:	nonrobust		

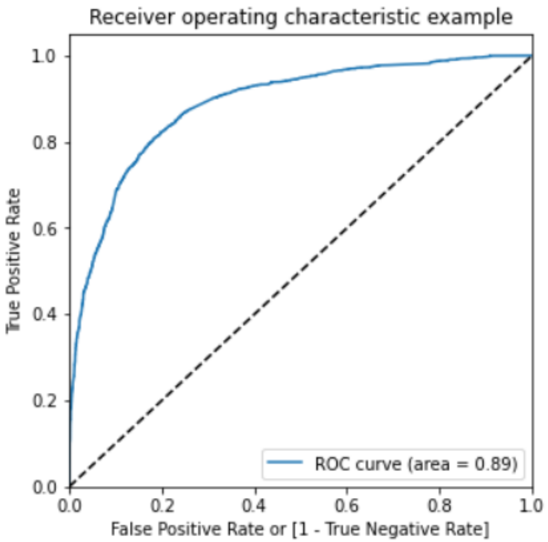
```
vif = pd.DataFrame()
vif['Features'] = X_train.columns
vif['VIF'] = [variance_inflation_factor(X_train.values, i)
vif['VIF'] = round(vif['VIF'], 2)
vif = vif.sort_values(by = "VIF", ascending = False)
vif
```

	Features	VIF
18	What is your current occupation_Unemployed	160.73
3	Lead Origin_Lead Add Form	62.72
25	Last Notable Activity_Modified	60.36
23	Last Notable Activity_Email Opened	56.96
6	Lead Source_Reference	48.15
29	Last Notable Activity_SMS Sent	46.14
7	Lead Source_Welingak Website	15.53

Selected Model Summary

Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	6351
Model:	GLM	**Df Residuals:**	6336
Model Family:	Binomial	**Df Model:**	14
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2603.8
Date:	Mon, 17 Oct 2022	**Deviance:**	5207.6
Time:	00:08:02	**Pearson chi2:**	6.54e+03
No. Iterations:	7	**Pseudo R-squ. (CS):**	0.4014
Covariance Type:	nonrobust		



The ROC curve has a good value of 0.89

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	-0.5723	0.172	-3.336	0.001	-0.908	-0.236
Do Not Email	-1.5505	0.185	-8.398	0.000	-1.912	-1.189
Total Time Spent on Website	1.1053	0.041	27.260	0.000	1.026	1.185
Lead Origin_Landing Page Submission	-1.1802	0.128	-9.235	0.000	-1.431	-0.930
Lead Source_Olark Chat	1.0626	0.122	8.744	0.000	0.824	1.301
Lead Source_Reference	3.3376	0.242	13.806	0.000	2.864	3.811
Lead Source_Welingak Website	5.8895	0.732	8.044	0.000	4.454	7.324
Last Activity_Email Opened	0.5679	0.130	4.352	0.000	0.312	0.824
Last Activity_Other_Activity	1.7413	0.241	7.238	0.000	1.270	2.213
Last Activity_Page Visited on Website	0.4204	0.178	2.363	0.018	0.072	0.769
Last Activity_SMS Sent	1.8180	0.131	13.913	0.000	1.562	2.074
Specialization_Others	-1.1979	0.125	-9.547	0.000	-1.444	-0.952
What is your current occupation_Working Professional	2.6404	0.197	13.411	0.000	2.255	3.026
Last Notable Activity_Modified	-0.8849	0.089	-9.928	0.000	-1.060	-0.710
Last Notable Activity_Olark Chat Conversation	-0.8796	0.346	-2.545	0.011	-1.557	-0.202

Lead Source (Wellingak Website and Reference)
And Current Occupation – Working Professional
have significant Positive Corelations

Outcome / Conclusion

FINAL OUTCOME

TRAIN SET Accuracy 81.24 | Sensitivity 80.9 | Specificity 81.4 | Precision 73.1 | Recall 80.9

TEST SET Accuracy 48.24 | Sensitivity 94.5 | Specificity 21.85 | Precision 40.8 | Recall 94.5

Conclusion : Though the Model Accuracy of the Test Set is Low - We Can still go ahead with the Above Model

The Sensitivity is High - That is Lead Conversion (Yes) - is correctly Predicted

The Recall is also Very High - That is Though there are some false Negatives True Positivity Rate is very high

Thus the Model will tend to Identify more than 80% of the Leads that can be converted correctly