Lead Scoring Case Study

By-

Agam Kushwaha

Aditya Mangani

Business Objective

- 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. When these people fill up a form providing their email address or phone number, they are classified to be a lead.
- The company wants to help them in selecting the most promising leads, i.e. the leads that are most likely to convert into paying customers.
- To build a logistic regression model to assign a lead score value between 0 and 100 to each of the leads which can be used by the company to target potential leads.

Data Reading and Understanding

- **Initial Steps**
  - Imported libraries
  - Checked top few rows
  - Checked Shape
  - Data Types Missing Values
  - Statistical Parameters etc.

	Prospect ID	Lead Number	Lead Origin	Lead Source	Do Not Email	Do Not Call	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Last Activity	Country	Specialization	How did you hear about X Education	What yo curre occupatic
0	7927b2df-8bba-4d29-b9a2-b6e0beafe620	660737	API	Olark Chat	No	No	0	0.0	0	0.0	Page Visited on Website	NaN	Select	Select	Unemploye
1	2a272436-5132-4136-86fa-dcc88c88f482	660728	API	Organic Search	No	No	0	5.0	674	2.5	Email Opened	India	Select	Select	Unemploye
2	8cc8c611-a219-4f35-ad23-fdfd2656bd8a	660727	Landing Page Submission	Direct Traffic	No	No	1	2.0	1532	2.0	Email Opened	India	Business Administration	Select	Stude
3	0cc2df48-7cf4-4e39-9de9-19797f9b38cc	660719	Landing Page Submission	Direct Traffic	No	No	0	1.0	305	1.0	Unreachable	India	Media and Advertising	Word Of Mouth	Unemploye
4	3256f628-e534-4826-9d63-4a8b88782852	660681	Landing Page Submission	Google	No	No	1	2.0	1428	1.0	Converted to Lead	India	Select	Other	Unemploye

Data Cleaning

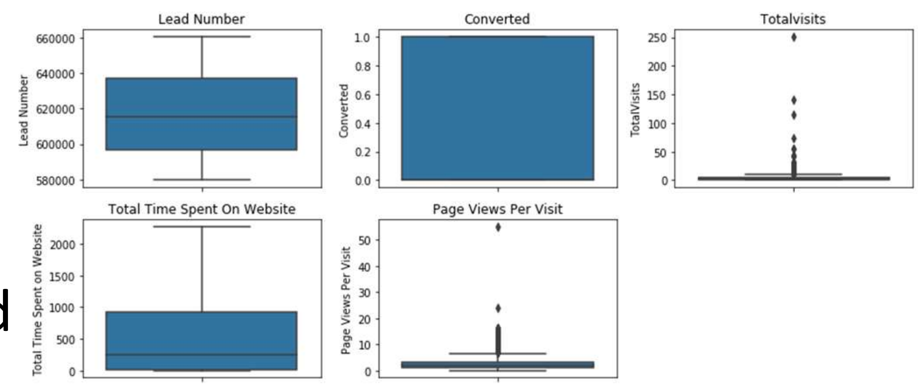
- We found that a lot of records contain the value "Select". It means the customer didn't select any option provided in the form.
- Replaced "Select" with NAN
- Removed the columns which have missing value more than 30%
- Imputed missing value in the below columns with their mode:
  - What matters most to you in choosing a course
  - What is your current occupation
  - Country
- 3 columns were left with missing values around 1%. Dropped NAN values.
- Checked number of Unique Values in each column

Data Cleaning contd…

- Checked Unique Values present in each column
- Removed below columns as data was being presented by single value:
  - Magazine
  - X Education Forums
  - Receive More Updates About Our Courses
  - Update me on Supply Chain Content
  - Get updates on DM Content
  - I agree to pay the amount through cheque
- Removed below columns which have 'No' values greater than 99%:
  - Do Not Email
  - Do Not Call
  - Search
  - Newspaper Article
  - Newspaper
  - Digital Advertisement
  - Through Recommendations
  - What matters most to you in choosing a course

Outlier Treatment

- Checked outliers using percentiles and Boxplot
- The columns **TotalVisits** and **Page Views Per Visit have Outliers** had outliers
- Treated them using IQR Method




```
Q1 = leads_data['TotalVisits'].quantile(0.25)
Q3 = leads_data['TotalVisits'].quantile(0.75)
IQR = Q3 - Q1
leads_data=leads_data.loc[(leads_data['TotalVisits'] >= Q1 - 1.5*IQR) & (leads_data['TotalVisits'] <= Q3 + 1.5*IQR)]

Q1 = leads_data['Page Views Per Visit'].quantile(0.25)
Q3 = leads_data['Page Views Per Visit'].quantile(0.75)
IQR = Q3 - Q1
leads_data=leads_data.loc[(leads_data['Page Views Per Visit'] >= Q1 - 1.5*IQR) & (leads_data['Page Views Per Visit'] <= Q3 + 1
```

Data Preparation

- **Binary Variables**
  - The column "**A free copy of Mastering The Interview**" was the only column with the values **'Yes**' and **'No**'. Let's converted them in the form of 1 and 0:
- **Dummy Variables**
  - Created dummy variables for below columns:
    - Lead Origin
    - Lead Source
    - Last Activity
    - What is your current occupation
    - Last Notable Activity
  - Dropped original columns

Train-Test Split and Feature Scaling

- **Train-Test Split**
  - The original Dataframe was split into train and test dataset. The train dataset was used to train the model and test dataset was used to evaluate the model.
  - Used **train_test_split** method of SKLearn library.
- **Feature Scaling**
  - Scaling helps to make the data in standard range and interpretation becomes easy.
  - Standardization was used for scaling.
  - Used StandardScaler method of SKLearn library.

Feature Selection Using RFE

- **Recursive Feature Elimination**
  - RFE is a technique to get optimum features which are best performing and showing best relation with target feature.
  - Selected 15 features for RFE.
- **Columns supported by RFE**

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

col = X_train.columns[rfe.support_]
col
```

```
Index(['Total Time Spent on Website', 'Lead Origin_Lead Add Form',
       'Lead Source_Direct Traffic', 'Lead Source_Google',
       'Lead Source_Organic Search', 'Lead Source_Referral Sites',
       'Lead Source_Welingak Website', 'Last Activity_Converted to Lead',
       'Last Activity_Email Bounced', 'Last Activity_Had a Phone Conversation',
       'Last Activity_Olark Chat Conversation',
       'What is your current occupation_Housewife',
       'What is your current occupation_Working Professional',
       'Last Notable Activity_SMS Sent', 'Last Notable Activity_Unreachable'],
      dtype='object')
```

Model Building

- **Logistic Regression**
  - Used Generalized Linear Models (GLM) from StatsModels library
  - Model was built using RFE with 15 features.
  - Added a constant for training Data features
  - In first model, all VIFs were in significant range; however, p-value of the feature “What is your current occupation_Housewife” was very high.
  - Removed that feature and ran the model again.
  - In second model, all VIFs and p-values were in significant range.
  - Considered the Model 2 as stable model and used it for further analysis.

```
X_train_sm = sm.add_constant(X_train[col])
logm_2 = sm.GLM(y_train, X_train_sm, family = sm.families.Binomial()).fit()
logm_2.summary()
```

P-Values and VIFs

- Below are the VIFs and p-values of the features obtained in Model 2:

P-Values

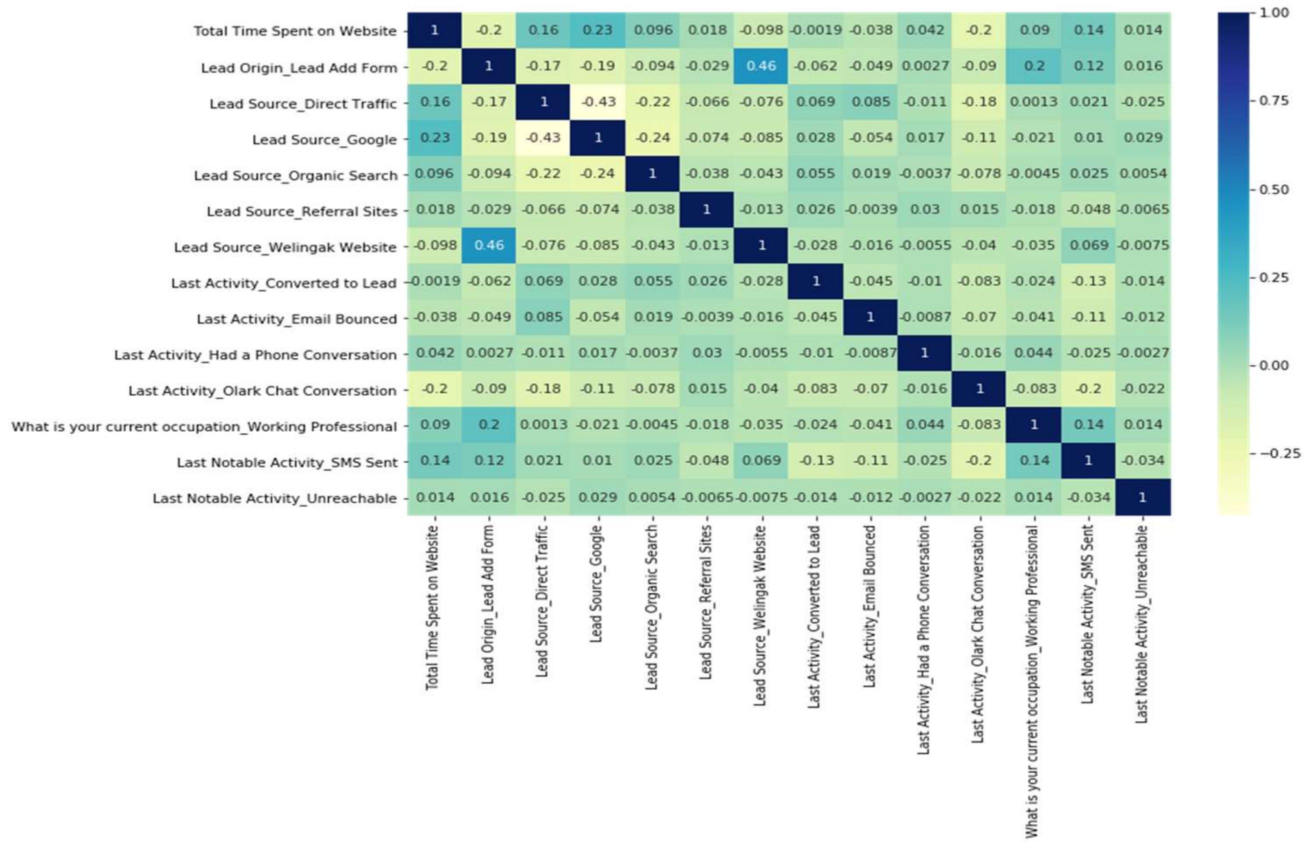
	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.1634	0.089	-1.829	0.067	-0.339	0.012
Total Time Spent on Website	1.1115	0.041	26.840	0.000	1.030	1.193
Lead Origin_Lead Add Form	2.6546	0.228	11.617	0.000	2.207	3.102
Lead Source_Direct Traffic	-1.4439	0.118	-12.248	0.000	-1.675	-1.213
Lead Source_Google	-1.1119	0.112	-9.913	0.000	-1.332	-0.892
Lead Source_Organic Search	-1.2551	0.142	-8.824	0.000	-1.534	-0.976
Lead Source_Referral Sites	-1.4198	0.381	-3.728	0.000	-2.166	-0.673
Lead Source_Welingak Website	2.6126	1.038	2.516	0.012	0.577	4.648
Last Activity_Converted to Lead	-1.2547	0.215	-5.826	0.000	-1.677	-0.833
Last Activity_Email Bounced	-2.1033	0.348	-6.046	0.000	-2.785	-1.421
Last Activity_Had a Phone Conversation	2.4248	1.155	2.100	0.036	0.162	4.688
Last Activity_Olark Chat Conversation	-1.6743	0.168	-9.971	0.000	-2.003	-1.345
What is your current occupation_Working Professional	2.8733	0.198	14.478	0.000	2.484	3.262
Last Notable Activity_SMS Sent	1.4862	0.083	17.955	0.000	1.324	1.648
Last Notable Activity_Unreachable	1.6225	0.560	2.896	0.004	0.525	2.720

VIF

Features	VIF
Lead Origin_Lead Add Form	1.51
Last Notable Activity_SMS Sent	1.38
Lead Source_Welingak Website	1.31
Lead Source_Google	1.25
Lead Source_Direct Traffic	1.24
Total Time Spent on Website	1.21
What is your current occupation_Working Profes...	1.18
Last Activity_Converted to Lead	1.11
Lead Source_Organic Search	1.10
Last Activity_Olark Chat Conversation	1.09
Last Activity_Email Bounced	1.07
Lead Source_Referral Sites	1.01
Last Activity_Had a Phone Conversation	1.01
Last Notable Activity_Unreachable	1.01

Data Visualization

- **Heat Map**
  - Below is the heatmap containing 14 features with very less multicollinearity.



Prediction of Conversion Probability

- Created a Dataframe with actual Leads and predicted probabilities

	Conversion	Conversion_Prob	Lead_ID
0	0	0.244526	5123
1	0	0.072237	6322
2	0	0.098727	3644
3	0	0.144717	3011
4	0	0.023779	8140

- Created new column “Predicted” with 1 if threshold probability > 0.5

	Conversion	Conversion_Prob	Lead_ID	Predicted
0	0	0.244526	5123	0
1	0	0.072237	6322	0
2	0	0.098727	3644	0
3	0	0.144717	3011	0
4	0	0.023779	8140	0

Training : Confusion Matrix and Accuracy

- **Confusion Matrix**
  - Below is command used and result

```
confusion_train = metrics.confusion_matrix(y_train_pred_final['Conversion'], y_train_pred_final['Predicted'])
print(confusion_train)
```

```
[[3267  439]
 [ 695 1558]]
```

- **Accuracy**
  - Received 81% Accuracy in the model
  - Below is the command used

```
print(metrics.accuracy_score(y_train_pred_final['Conversion'], y_train_pred_final['Predicted']))
```

```
0.8096996140291995
```

Training : Metrics other than Accuracy

- Below are the other metrics:
  - Sensitivity
  - Specificity
  - False Positive Rate
  - Positive Predictive Value
  - Negative Predictive Value

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

Sensitivity:

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

0.6919662671992899

Specificity:

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

0.8815434430652995

False Postive Rate:

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

0.11845655693470049

Positive Predictive Value:

```
print (TP / float(TP+FP))
```

0.7802802802802803

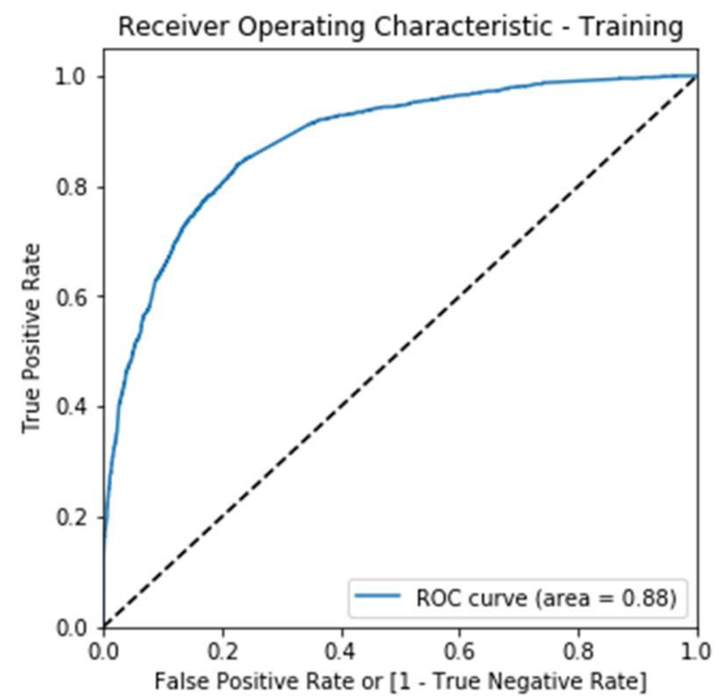
Negative Predictive Value:

```
print (TN / float(TN+ FN))
```

0.8247917192628125

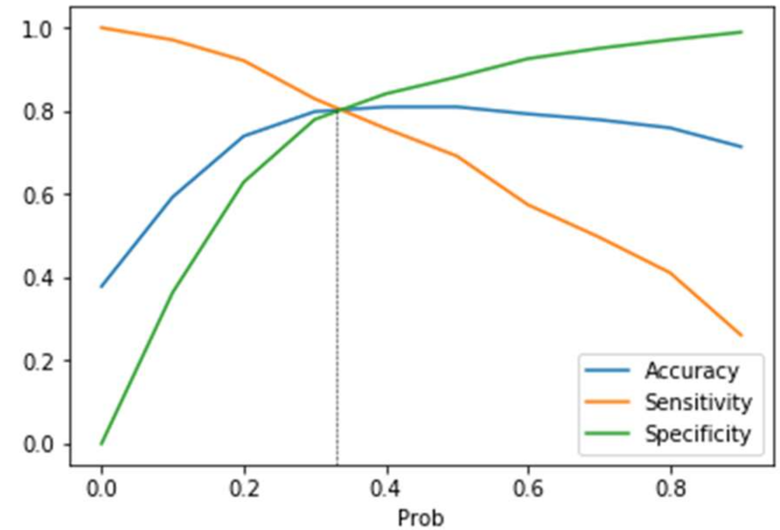
ROC Curve and AUC

- **Receiver Operating Characteristics**
  - ROC shows relation between True Positive Rate and False Positive Rate
  - Curve is going towards left upper section which shows True Positive Rate is high.
- **Area Under the Curve**
  - Goodness of the model is determined by AUC.
  - In the graph, it shows AUC Score as 88% which is good sign for model.



Optimal Probability Threshold

- **Accuracy, Sensitivity and Specificity Curve**
  - Calculated Accuracy, Sensitivity and Specificity for different probability cutoffs and plotted them.
  - The intersection point of Accuracy, Sensitivity and Specificity gives optimal value of probability threshold.
  - As per the graph shown. We got threshold probability as 0.33.



Making Prediction on Test Set

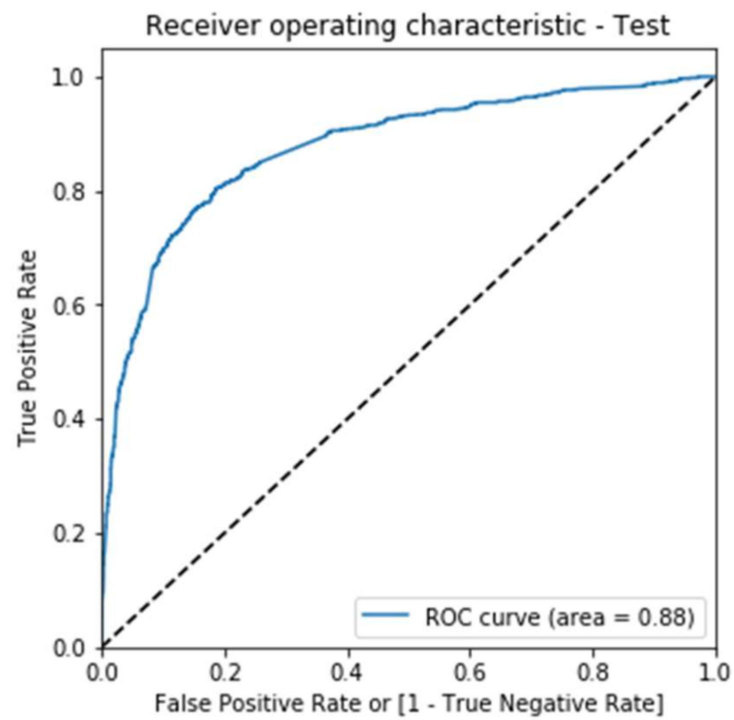
- Scaled numerical columns
- Took RFE supported columns in Test data set
- Added predicted probabilities to the leads
- Leads from the test data were predicted with threshold value of 0.33
- Created a final dataset

	Lead_ID	Converted	Conversion_Prob	Final_Predicted
0	7358	0	0.060495	0
1	8398	0	0.769079	1
2	3472	0	0.127308	0
3	8673	1	0.297090	0
4	8053	1	0.864495	1

Model Validation on Test Data Set

- Plotted ROC curve with Test Data
- Similar to the Train Data, the curve is going towards left upper section which shows True Positive Rate is high.
- AUC Score is 88% which is matching with Train Data. It shows goodness of the model

Hence, we can say that our model is validated on Test Data set and producing the same result similar to Train Data set.



Test : Confusion Matrix and Accuracy

- **Confusion Matrix**
  - Below is command used and result:

```
confusion_test = metrics.confusion_matrix(y_test_pred_final['Converted'], y_test_pred_final['Final_Predicted'])
print(confusion_test)
```

```
[[1274  327]
 [ 177  776]]
```

- **Accuracy**
  - Received 80.2% Accuracy in the model which is close to accuracy of Train Data
  - Below is the command used

```
accuracy_score=metrics.accuracy_score(y_test_pred_final['Converted'], y_test_pred_final['Final_Predicted'])
print(accuracy_score)
```

```
0.8026624902114331
```

Test : Metrics other than Accuracy

- Below are the other metrics:
  - Sensitivity
  - Specificity
  - False Positive Rate
  - Positive Predictive Value
  - Negative Predictive Value

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

Sensitivity:

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

0.8142707240293809

Specificity:

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

0.7957526545908807

False Postive Rate:

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

0.2042473454091193

Positive Predictive Value:

```
print (TP / float(TP+FP))
```

0.7035358114233907

Negative Predictive Value:

```
print (TN / float(TN+ FN))
```

0.8780151619572708

Lead Score Calculation

- Lead score was calculated on entire data set (Train + Test)
- Below formula was used:

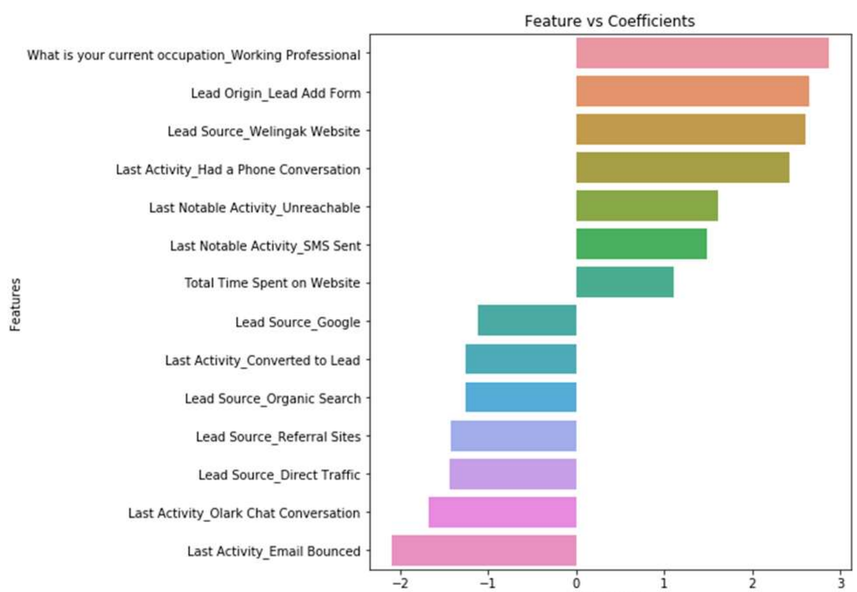
Lead Score = 100 * (Conversion Probability)

- Train and Test data was combined to entire Leads
- Higher the lead score, higher is the probability of a lead getting converted and vice versa
- Lead with Lead score greater than 33 has 1 in the Final_Predicted column because we chose the cut-off as 0.33.

	Lead_ID	Lead Number	Converted	Conversion_Prob	Final_Predicted	Lead_Score
0	0	660737	0	0.244526	0	24
1	1	660728	0	0.266255	0	27
2	2	660727	1	0.631987	1	63
3	3	660719	0	0.124306	0	12
4	4	660681	1	0.355913	1	36

Analysis of Feature Coefficients

- 14 features with their coefficients are shown in decreasing order in the right figure
- The same can be visualized through below Bar Plot:



	Features	Coefficients
11	What is your current occupation_Working Profes...	2.87
1	Lead Origin_Lead Add Form	2.65
6	Lead Source_Welingak Website	2.61
9	Last Activity_Had a Phone Conversation	2.42
13	Last Notable Activity_Unreachable	1.62
12	Last Notable Activity_SMS Sent	1.49
0	Total Time Spent on Website	1.11
3	Lead Source_Google	-1.11
7	Last Activity_Converted to Lead	-1.25
4	Lead Source_Organic Search	-1.26
5	Lead Source_Referral Sites	-1.42
2	Lead Source_Direct Traffic	-1.44
10	Last Activity_Olark Chat Conversation	-1.67
8	Last Activity_Email Bounced	-2.10

Conclusion

- As per the analysis, below are the features which help a lot to get successful Lead conversion:
- **Features with Positive Coefficient**
  - What is your current occupation_Working Professional
  - Lead Origin_Lead Add Form
  - Lead Source_Welingak Website
  - Last Activity_Had a Phone Conversation
  - Last Notable Activity_Unreachable
  - Last Notable Activity_SMS Sent
  - Total Time Spent on Website
- **Features with Negative Coefficient**
  - Lead Source_Google
  - Last Activity_Converted to Lead
  - Lead Source_Organic Search
  - Lead Source_Referral Sites
  - Lead Source_Direct Traffic
  - Last Activity_Olark Chat Conversation
  - Last Activity_Email Bounced

It means, below are the main 2 conclusion points:

- **The conversion probability increses with increase in value of the features with positive coefficient.**
- **The conversion probability increses with decrease in value of the features with negative coefficient.**