Lead Scoring Case -Study

Presented by

Zuber Nagani

Problem Statement and Solution approach

The company requires to build a model wherein we need to assign a lead score to each of the leads such that the customers with the higher lead score have a higher conversion chance and the customers with the lower lead score have a lower conversion chance. **Solution approach-**

Step 1: Importing and Cleaning Data – Perform EDA

Step-2 Create Dummy Variables for Categorical Variable

Step 3 - Test Train Split

Step-4 Scaling Data Using Standard Scaler

Step-5 Eliminate Highly Correlated Data before Model Building

Step 6: Feature Selection Using RFE

Step 7 Model Building

Step-8 Metrics beyond simply accuracy

Step 9: Plotting the ROC Curve

Step 10A: Finding Optimal Cutoff Point

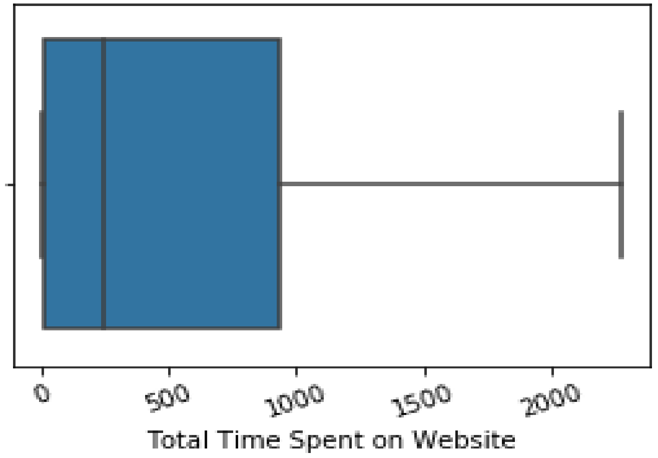
Step-10B (Optional Step) Precision and Recall

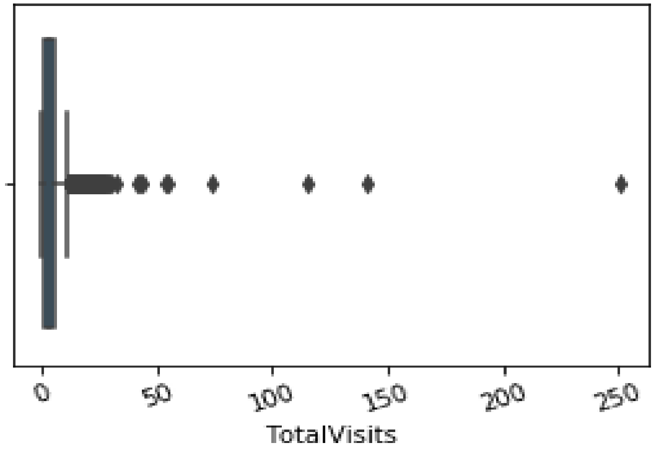
Step 11: Making predictions on the test set

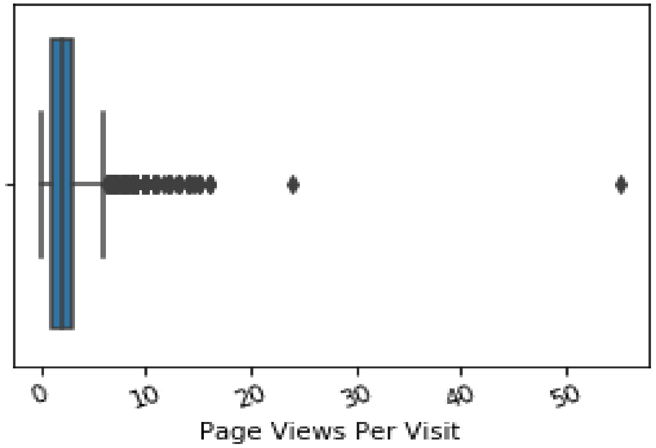
Data Cleaning and Imputation –

Outliers Treatment -

- Outlier Treatment - In the Numerical Variables, Outlier were present.
- Capping – We capped the values using Quantiles (0.01 and 0.99) in order to get rid of outliers.



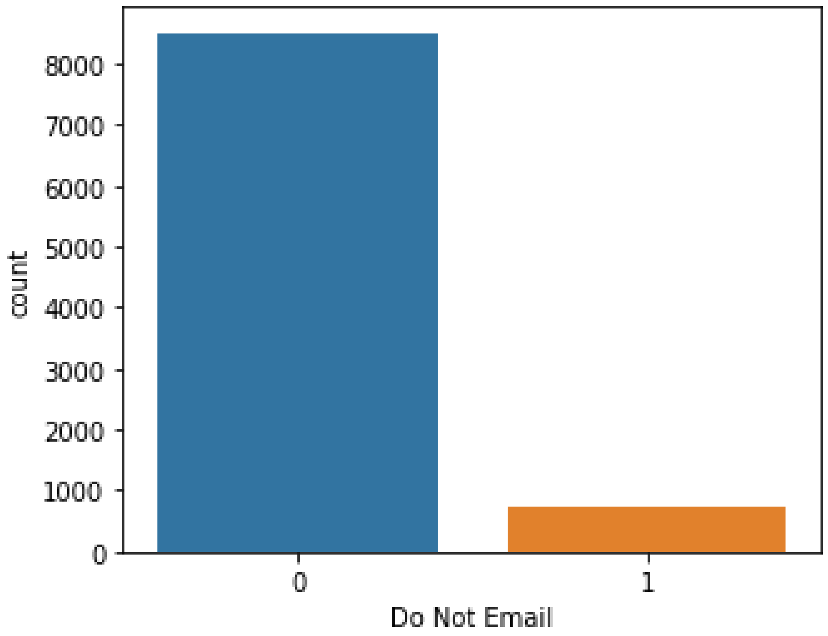


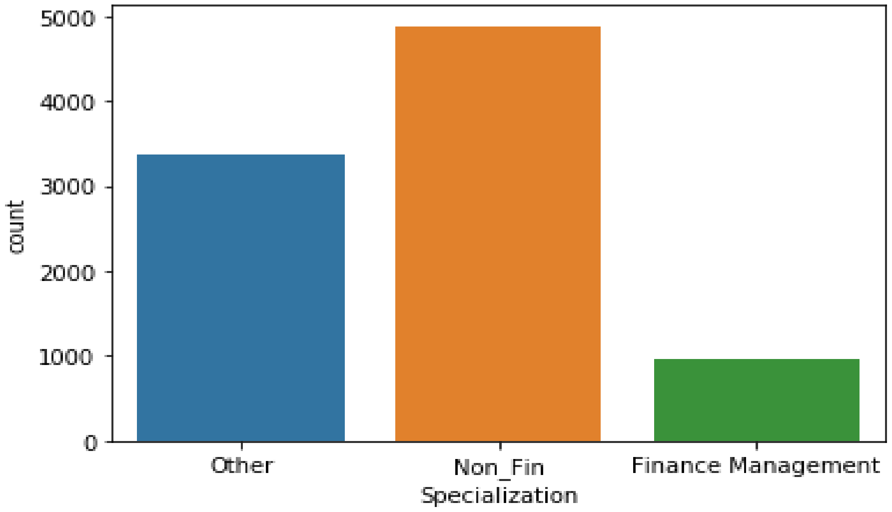


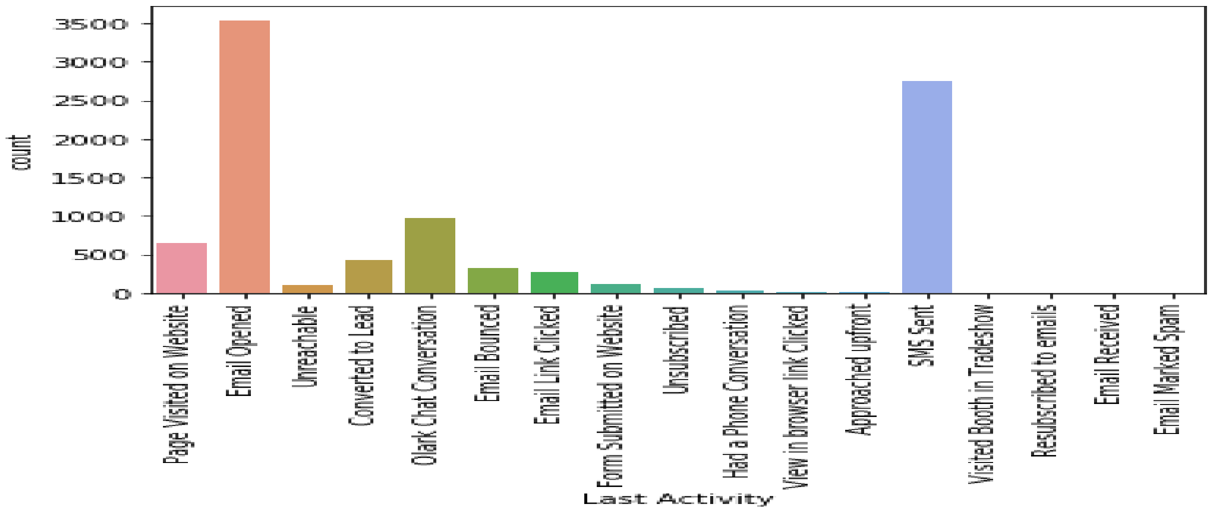
Data Cleaning and Imputation –

Categorical Variables- Checked the skewness of Data

- There are many categorical variables in which data was heavily skewed.
- Club the data categories into single category such as “Other” to minimize the number of dummy variables for a single attribute.







- Also, we dropped some of the columns that are highly skewed. (Like – 90%-10%) because it wont help in any new finding.

Data Cleaning and Imputation –

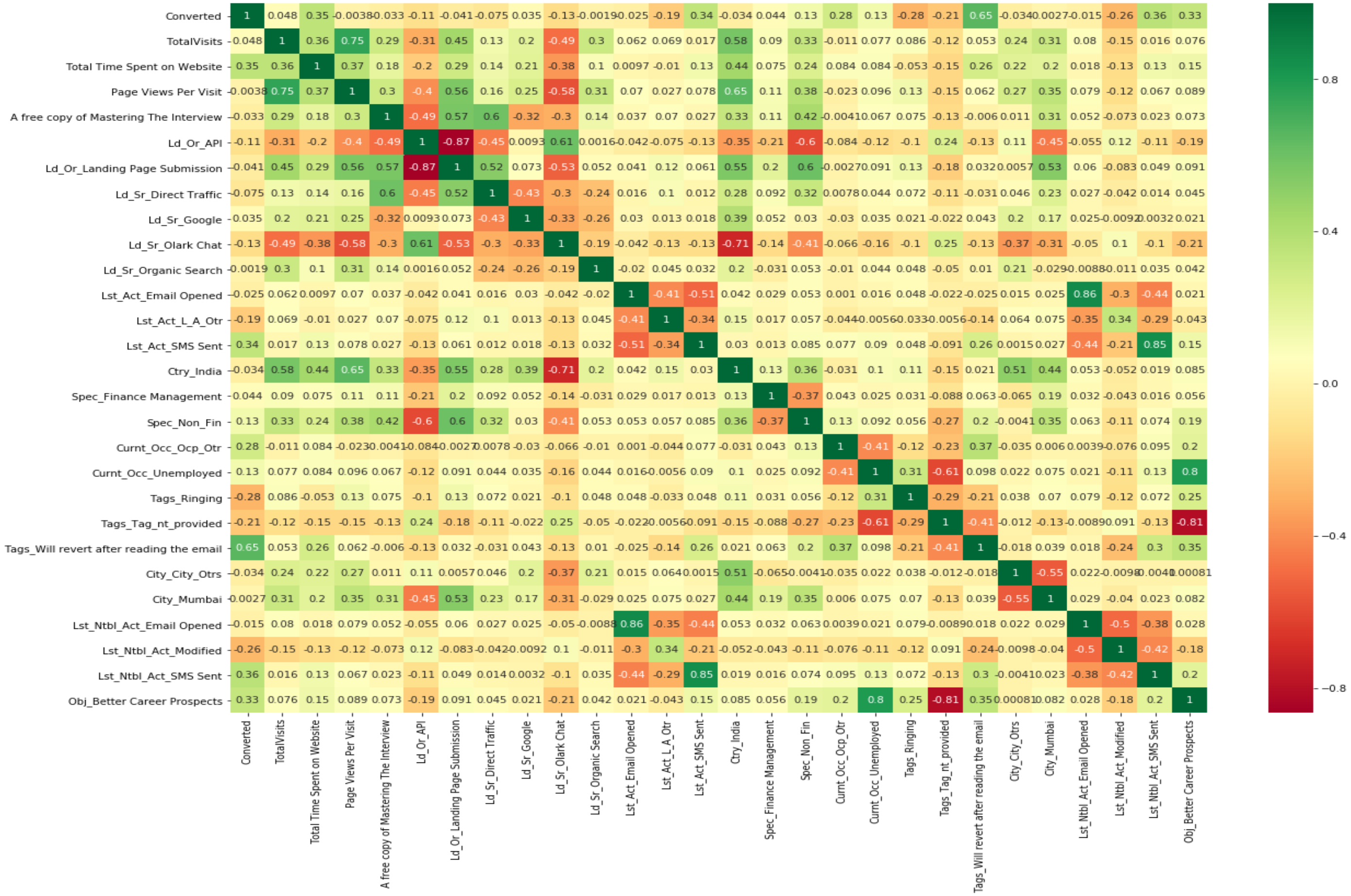
Summary

- Capped Outliers by using quantiles.
- Dropped the columns, having more that 50% null values.
- Imputed null values with appropriate method such as by using mean, median and mode and other metric.
- Replaced “Select” value with NULL and handled thereafter.
- Created new attribute (like “Others”) in columns if the values are highly skewed.
- Created Dummy variables for all categorical variables and replaced the original columns.

Test-Train Split

Divided the data set into Test and Train Dataset

- Split Dependent and Independent variables
- Scale data using Standard Scaler
- Drop high correlated data manually before moving to RFE.



Model Building

Logistic Regression

What is achieved by the exercise?

- No multicollinearity . (Less VIF) • High significance. (Less P Values)
- High Accuracy.
- Less Complexity. (Eliminate as much feature as possible without compromising much with Accuracy)
- Decent Trade-Off between TPR and FPR (Here we considered higher TPR given FPR is not too high)

Model Building Contd..

Logistic Regression

- Created Model-1 with 15 Features select by RFE as output.
- Overall 7 Model are made iteratively to find most highly significant features to predict the conversion rate.
- For instance, Model-1 metrices are given.
- **Accuracy:** 88% - Accuracy defines the number of correct predictions out of all predictions made. It Means model are 88% accurate of predicting the actually converted customers.
- Since the variable 'Ctry_India' has the highest VIF. We dropped the column, again built the model.

Dep. Variable:	Converted	No. Observations:	6372
Model:	GLM	Df Residuals:	6356
Model Family:	Binomial	Df Model:	15
Link Function:	logit	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-1783.3
Date:	Sun, 31 May 2020	Deviance:	3566.6
Time:	23:37:13	Pearson chi2:	8.80e+03
No. Iterations:	7		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.7762	0.218	-3.563	0.000	-1.203	-0.349
TotalVisits	0.5057	0.064	7.876	0.000	0.380	0.632
Total Time Spent on Website	1.0342	0.050	20.508	0.000	0.935	1.133
Page Views Per Visit	-0.6009	0.077	-7.797	0.000	-0.752	-0.450
Ld_Sr_Direct Traffic	-2.1339	0.195	-10.969	0.000	-2.515	-1.753
Ld_Sr_Google	-1.8463	0.197	-9.370	0.000	-2.233	-1.460
Ld_Sr_Olark Chat	-2.3180	0.181	-12.800	0.000	-2.673	-1.963
Ld_Sr_Organic Search	-2.0626	0.224	-9.204	0.000	-2.502	-1.623
Lst_Act_L_A_Otr	-0.6501	0.114	-5.693	0.000	-0.874	-0.426
Ctry_India	-0.9189	0.164	-5.586	0.000	-1.241	-0.597
Curnt_Occ_Ocp_Otr	0.7045	0.174	4.048	0.000	0.363	1.046

	Features	VIF
8	Ctry_India	12.03
4	Ld_Sr_Google	7.00
3	Ld_Sr_Direct Traffic	5.64
14	Obj_Better Career Prospects	5.50
6	Ld_Sr_Organic Search	3.41
11	Tags_Tag_nt_provided	3.09
2	Page Views Per Visit	2.97
5	Ld_Sr_Olark Chat	2.49
0	TotalVisits	2.42
12	Tags_Will revert after reading the email	2.06
13	Lst_Ntbl_Act_SMS Sent	1.66
10	Tags_Ringing	1.51
7	Lst_Act_L_A_Otr	1.47
1	Total Time Spent on Website	1.36
9	Curnt_Occ_Ocp_Otr	1.31

Model Building Contd..

Logistic Regression

- After the iterative Feature elimination exercise, for 7th model we achieved –
- No multicollinearity (VIF score less than 2 here which is pretty good)
- Simple Model (left with 9 features and every feature is significant)
- High Accuracy (Not compromised much in feature elimination. Overall Model Accuracy comes out to be 86%)

Model Building Contd..

Dep. Variable:	Converted	No. Observations:	6372
Model:	GLM	Df Residuals:	6362
Model Family:	Binomial	Df Model:	9
Link Function:	logit	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-1978.3
Date:	Sun, 31 May 2020	Deviance:	3956.7
Time:	23:37:14	Pearson chi2:	7.85e+03
No. Iterations:	7		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	1.1846	0.138	8.581	0.000	0.914	1.455
Total Time Spent on Website	0.9932	0.047	21.110	0.000	0.901	1.085
Ld_Sr_Direct Traffic	-3.0726	0.165	-18.609	0.000	-3.396	-2.749
Ld_Sr_Google	-2.7877	0.160	-17.440	0.000	-3.101	-2.474
Ld_Sr_Olark Chat	-2.1771	0.158	-13.813	0.000	-2.486	-1.868
Ld_Sr_Organic Search	-2.9580	0.186	-15.864	0.000	-3.324	-2.593
Lst_Act_L_A_Otr	-0.6330	0.109	-5.786	0.000	-0.847	-0.419
Tags_Ringing	-3.3150	0.239	-13.866	0.000	-3.784	-2.846
Tags_Will revert after reading the email	4.2976	0.172	24.996	0.000	3.961	4.635
Lst_Ntbl_Act_SMS Sent	1.8270	0.106	17.174	0.000	1.619	2.036

	Features	VIF
8	Lst_Ntbl_Act_SMS Sent	1.56
7	Tags_Will revert after reading the email	1.49
1	Ld_Sr_Direct Traffic	1.46
2	Ld_Sr_Google	1.45
5	Lst_Act_L_A_Otr	1.42
0	Total Time Spent on Website	1.27
6	Tags_Ringing	1.25
3	Ld_Sr_Olark Chat	1.21
4	Ld_Sr_Organic Search	1.21

Logistic Regression

- **Accuracy** - It defines the ability to differentiate the converted and non converted customers correctly. Accuracy is the proportion of actual converted and actual non-converted in all evaluated cases.
- **Sensitivity** -. It is the proportion of actual Converted customers that are correctly predicted as Converted by a model.
- **Specificity** - It is the proportion of Non-Converted customers that are correctly predicted as NonConverted by a model.
- **FPR** – It is the number of Non-Converted Customers incorrectly identified as Converted.
- **Positive predictive value** – The proportion of actual converted customers as converted
- **Negative predictive value** - The proportion of non-converted customers as non-converted.

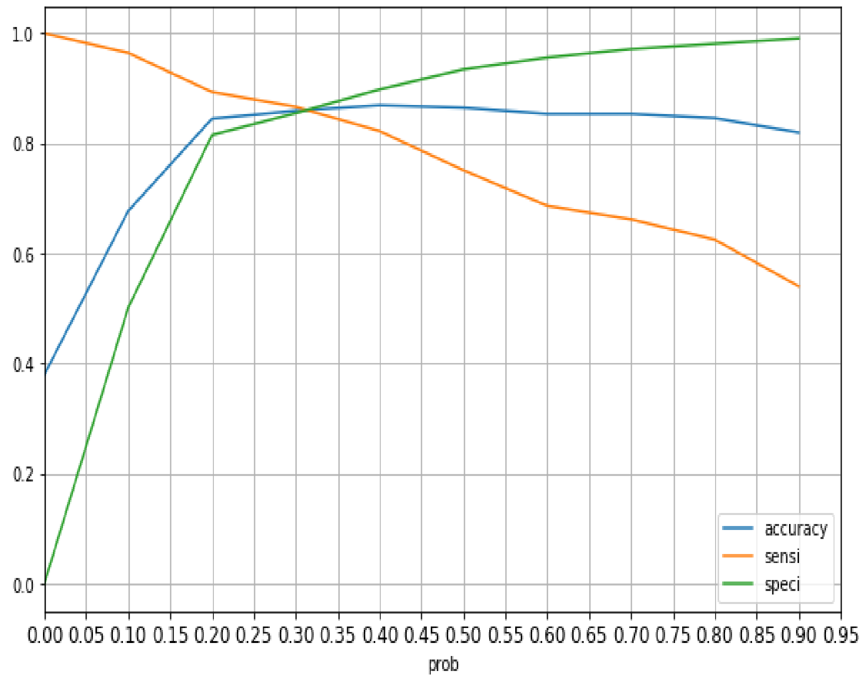
Optimal Cutoff Point (Train Model)

- 0.3 is taken as the optimum cutoff point as all the 3 metrices (Sensitivity, Specificity and Accuracy) seems to doing pretty good at this point.
- Accuracy – 85.9%

- Sensitivity – 86.6%
- Specificity – 85.5%
- FPR– 14.5%
- Positive predictive value – 78%
- Negative predictive value - 91%

ROC Curve and FPR-TPR Trade-Off (Train Model)

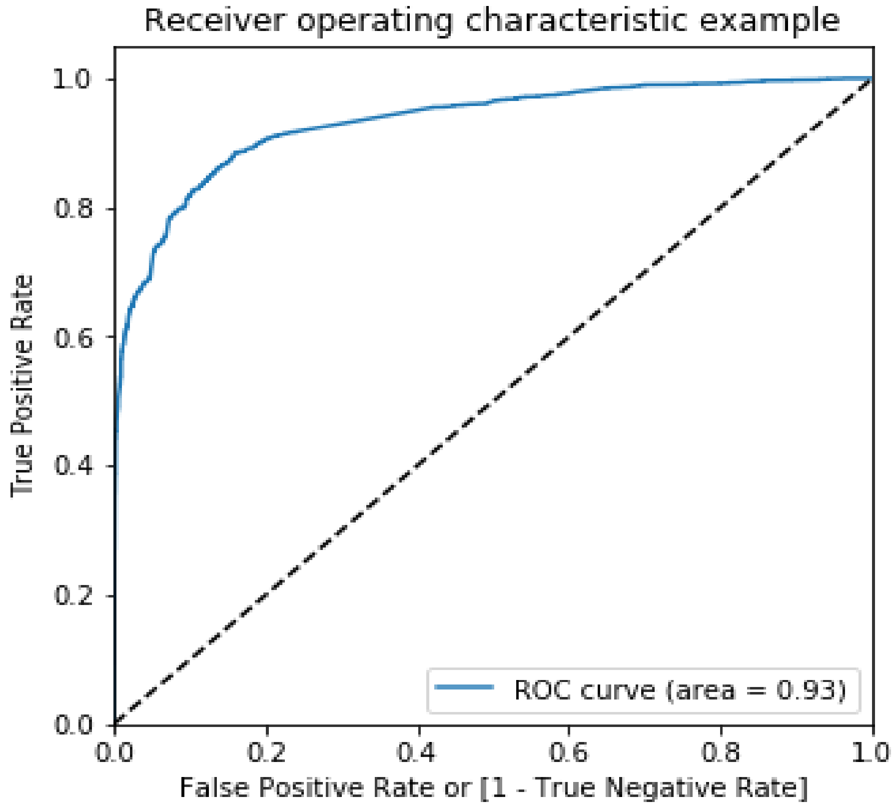
Probability	Accuracy	Sensitivity	Specificity
0.0%	38.0%	100.0%	0.0%
10.0%	67.7%	96.4%	50.1%
20.0%	84.5%	89.3%	81.5%
30.0%	**85.9%**	**86.6%**	**85.5%**
40.0%	86.9%	82.2%	89.8%
50.0%	86.5%	75.1%	93.4%
60.0%	85.4%	68.7%	95.6%
70.0%	85.4%	66.2%	97.1%
80.0%	84.6%	62.5%	98.1%
90.0%	82.0%	54.0%	99.1%



Here in the given case we should care about TPR (Sensitivity) reaches to 86.6% whereas FPR remains 14.5% for Probability Cut Off as 30%.

ROC is hugging Y axis which is good sign as Model is able to achieve high TPR with low FPR.

Probability	FPR (1-Specificity)	TPR(Sensitivity)
0.0%	100.0%	100.0%
10.0%	49.9%	96.4%
20.0%	18.5%	89.3%
30.0%	**14.5%**	**86.6%**
40.0%	10.2%	82.2%
50.0%	6.6%	75.1%
60.0%	4.4%	68.7%
70.0%	2.9%	66.2%
80.0%	1.9%	62.5%
90.0%	0.9%	54.0%
100.0%	0.0%	0.0%



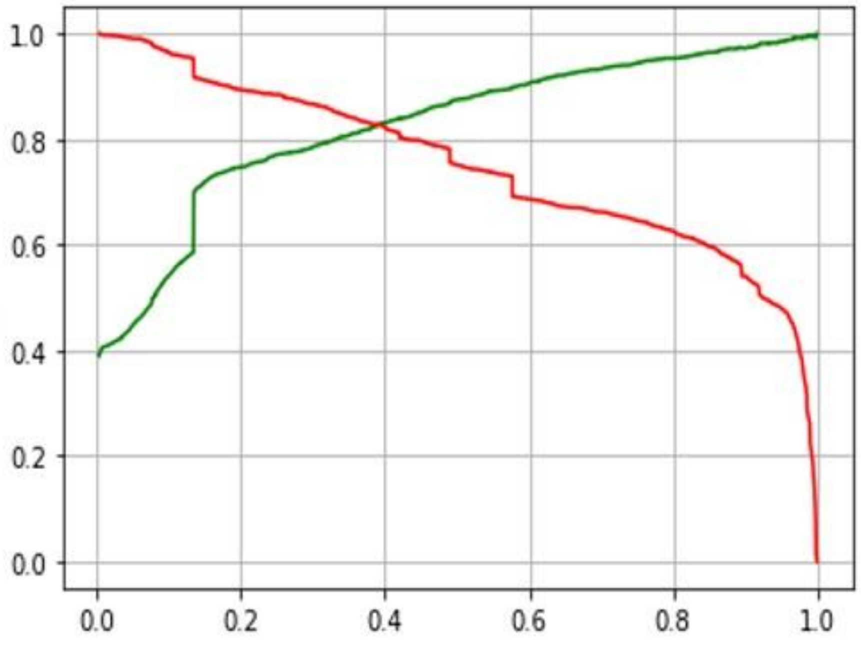
Precision and Recall (Train Model)

At 40% Probability we got decent Precision and Recall tradeoff.

Precision: Probability that a predicted Converted is actually a Converted.
Recall: Probability that an actual Converted case is predicted correctly. (same as sensitivity)
F1-score: Is useful when you want to look at the performance of precision and recall together.

Precision – Recall Trade Off

Probability	Precision	Recall	F1 Score
0%	38%	100%	55%
10%	54%	96%	69%
20%	75%	89%	81%
30%	79%	87%	82%
40%	**83%**	**82%**	**83%**
50%	88%	75%	81%
60%	91%	69%	78%
70%	93%	66%	77%
80%	95%	63%	75%
90%	97%	54%	69%

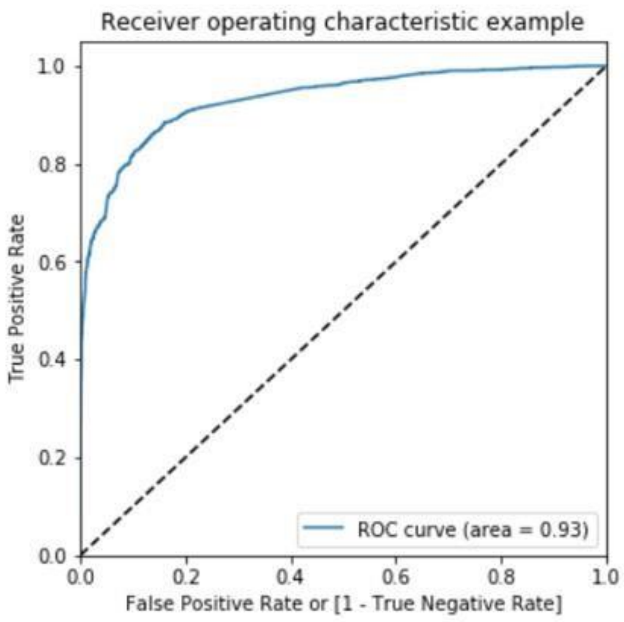


---Precision --- Recall

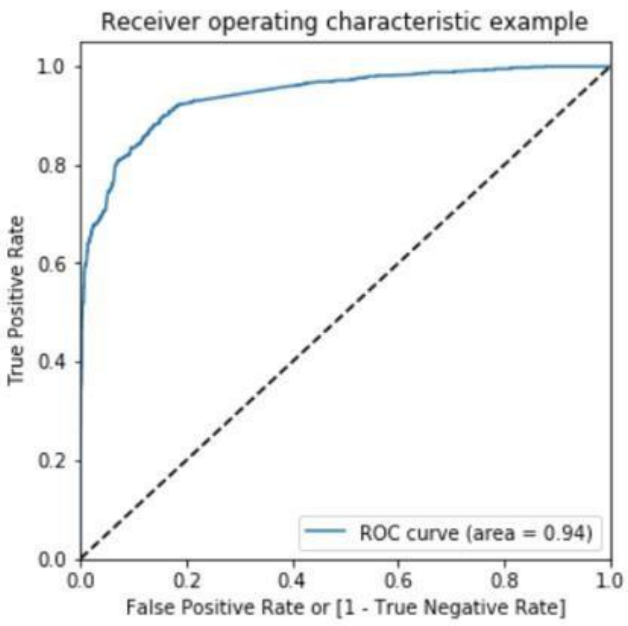
Train Vs. Test Model

	Probability	Accuracy	TPR (Sensitivity)	Specificity	FPR (1-Specificity)
Train Model	30.0%	85.9%	86.6%	85.5%	14.5%
Test Model	30.0%	86.70%	88.10%	85.90%	14.0%

ROC Curve



Train Model



Test Model

Most Significant Features

If we see the coefficients of all the features of final model, we would find that –

Features	Coeff
Tags_Will revert after reading the email	4.3
Lst_Ntbl_Act_SMS Sent	1.83
Total Time Spent on Website	0.99
Lst_Act_L_A_Otr	-0.63
Ld_Sr_Olark Chat	-2.18
Ld_Sr_Google	-2.79
Ld_Sr_Organic Search	-2.96
Ld_Sr_Direct Traffic	-3.07
Tags_Ringing	-3.31

How to interpret Positive Coefficients?

A positive coefficient simply implies that the probability that the event identified by the Dependent Variable happens increases as the value of the Independent Variable increases. In other words, when the value of the IV increases the probability increases. **How to interpret Negative Coefficients?**

A negative coefficient simply implies that the probability that the event identified by the Dependent Variable happens decreases as the value of the Independent Variable increases. In other words, when the value of the IV increases the probability decreases.

Recommendations

- Take Probability Cut-Off as 30% which will help to contact almost 86.6% (sensitivity) of the Leads with High Conversion chances.
- With the above call, the chance of having Lead as not converted is 14.5% (FPR) which company can bear as the focus is more on Converting as many leads as possible.

- With the given model the initial accuracy is doubled which can be further increased given the Company has an appetite to accept higher Non-Conversion Rate.
- If Company want the Leads to get converted then Leads with below features should be targeted-
- Tags_Will revert after reading the email
- Lst_Ntbl_Act_SMS Sent
- Total Time Spent on Website
- If Company want the Leads to get converted then Leads with below features should **not** be targeted-
- Lst_Act_L_A_Otr
- Ld_Sr_Olark Chat
- Ld_Sr_Google
- Ld_Sr_Organic Search
- Ld_Sr_Direct Traffic
- Tags_Ringing