Logistic Regression Case Study

By

Madhu & Rachit

17/11/2019

Business Objectives

X Education gets a lot of leads, its lead conversion rate is very poor. The company requires us to build a model wherein we need to assign a lead score to each of the leads such that the customers with higher lead score have a higher conversion chance and the customers with lower lead score have a lower conversion chance. The CEO, in particular, has given a ballpark of the target lead conversion rate to be around 80%.

Data

• **Source:** Upgrad given Data

• **Jupyter Notebook:** Lead Scoring Case Study by Madhu and Rachit

Data Files Provided

1. **'Leads.csv'** contains information about the client's leads from the past with around 9000 data points. This dataset consists of various attributes such as Lead Source, Total Time Spent on Website, Total Visits, Last Activity, etc. which may or may not be useful in ultimately deciding whether a lead will be converted or not. The target variable, in this case, is the column 'Converted' which tells whether a past lead was converted or not wherein 1 means it was converted and 0 means it wasn't converted.

2. **'Leads Data Dictionary.xlsx'** is data dictionary which describes the meanings of the columns. Some descriptions are given as NaN, it means they are self-explanatory.

3. All the variables description that belong to the Leads.csv are provided in the initial phase of the Jupyter notebook.

Packages used in the model

```
import numpy as np

import pandas as pd

from datetime import datetime as dt

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline
```

```
from sklearn.preprocessing import scale
from sklearn.datasets import fetch_mldata
from sklearn.decomposition import PCA
from sklearn.preprocessing import PowerTransformer
from sklearn import metrics
from sklearn.model_selection import train_test_split
import statsmodels.api as sm
from sklearn.linear_model import LogisticRegression
from sklearn.feature_selection import RFE
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.metrics import precision_score, recall_score
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import classification_report
from sklearn.model_selection import cross_val_score

import warnings
warnings.filterwarnings('ignore')

# settings to see the data. These are the settings to view the data in the jupyter notebook
pd.options.display.max_columns = None
pd.options.display.max_rows = None
pd.options.display.max_colwidth = 500
```

Data Dictionary

	Variables	Description
0	Prospect ID	A unique ID with which the customer is identified.
1	Lead Number	A lead number assigned to each lead procured.
2	Lead Origin	The origin identifier with which the customer was identified to be a lead. Includes API, Landing Page Submission, etc.

	Variables	Description
3	Lead Source	The source of the lead. Includes Google, Organic Search, Olark Chat, etc.
4	Do Not Email	An indicator variable selected by the customer wherein they select whether they want to be emailed about the course or not.
5	Do Not Call	An indicator variable selected by the customer wherein they select whether they want to be called about the course or not
6	Converted	The target variable. Indicates whether a lead has been successfully converted or not.
7	TotalVisits	The total number of visits made by the customer on the website.
8	Total Time Spent on Website	The total time spent by the customer on the website.
9	Page Views Per Visit	Average number of pages on the website viewed during the visits.
10	Last Activity	Last activity performed by the customer. Includes Email Opened, Olark Chat Conversation, etc.
11	Country	The country of the customer.
12	Specialization	The industry domain in which the customer worked before. Includes the level 'Select Specialization' which means the customer had not selected this option while filling the form.
13	How did you hear about X Education	The source from which the customer heard about X Education.
14	What is your current occupation	Indicates whether the customer is a student, unemployed or employed.
15	What matters most to you in choosing this course	An option selected by the customer indicating what is their main motto behind doing this course.
16	Search	Indicating whether the customer had seen the ad in any of the listed items.
17	Magazine	NaN
18	Newspaper Article	NaN
19	X Education Forums	NaN
20	Newspaper	NaN
21	Digital Advertisement	NaN
22	Through Recommendations	Indicates whether the customer came in through recommendations.
23	Receive More Updates About Our Courses	Indicates whether the customer chose to receive more updates about the courses.

	Variables	Description
24	Tags	Tags assigned to customers indicating the current status of the lead.
25	Lead Quality	Indicates the quality of lead based on the data and intuition the the employee who has been assigned to the lead.
26	Update me on Supply Chain Content	Indicates whether the customer wants updates on the Supply Chain Content.
27	Get updates on DM Content	Indicates whether the customer wants updates on the DM Content.
28	Lead Profile	A lead level assigned to each customer based on their profile.
29	City	The city of the customer.
30	Asymmetrique Activity Index	An index and score assigned to each customer based on their activity and their profile
31	Asymmetrique Profile Index	NaN
32	Asymmetrique Activity Score	NaN
33	Asymmetrique Profile Score	NaN
34	I agree to pay the amount through cheque	Indicates whether the customer has agreed to pay the amount through cheque or not.
35	a free copy of Mastering The Interview	Indicates whether the customer wants a free copy of 'Mastering the Interview' or not.
36	Last Notable Activity	The last notable activity performed by the student.

Data Preparation

We have used lot of techniques to prepare the data for the model. We removed duplicate rows and also dealt with the value “Select” by converting it to NaN. For more details on Data Preparation please refer the jupyter notebook.

Outlier Checks and Removals

Please refer to the jupyter notebook about how the outliers have been removed by visualizing the data through boxplots. Lower limit was 0.25 and higher was 0.75.

binary variables (Yes/No) to 0/1 Conversion

Please refer the jupyter notebook for the conversion of binary variables and the list of features being converted to binary is all present there.

Dummy Variable creation for the categorical variables

Please refer the jupyter notebook as how the dummy variables were created to make model. Previous columns were dropped as the new dummy variables have been created from them.

Splitting the data for Test and Train

We have split the data in 7:3 ratio for test and train. For more details please refer the jupyter notebook.

Scaling

We have used the Yeo-Johnson PowerTransformer scaler for scaling the numerical variables, dummies are not scaled. `PowerTransformer` applies a power transformation to each feature to make the data more Gaussian-like. Currently, `PowerTransformer` implements the Yeo-Johnson and Box-Cox transforms. The power transform finds the optimal scaling factor to stabilize variance and minimize skewness through maximum likelihood estimation. By default, `PowerTransformer` also applies zero-mean, unit variance normalization to the transformed output. Note that Box-Cox can only be applied to strictly positive data. Income and number of households happen to be strictly positive, but if negative values are present the Yeo-Johnson transformed is to be preferred. Please refer the jupyter notebook for more details.

Lead Conversion Rate

We calculated the lead conversion rate and it turned out to be 38 percent.

Recursive Feature Elimination (RFE)

We have eliminated all the features that the RFE algorithm has told us.

Model Assessment with Stats Model

We have analyzed the remaining features with the stats model library of python. We will further check the p-value and VIFs of the particular features to make model more robust and stable.

Adding the probability column w.r.t. Lead ID and do our predictions

We now calculate the conversion probability with the help of model assessment and perform our predictions.

Checking the efficiency with Confusion Matrix

We check the efficiency of our model using the Confusion Matrix. Please refer to jupyter notebook for more details.

Checking the VIFs

Now we check for the VIFs of all the variables being used in the model. Luckily all the features are below 2 in value so the variables do not possess multicollinearity.

Removing the variables on the basis of high p-value according to the model assessment.

We do have few variables in our model who have high p-values and required to be removed one by one. So, we have performed those iterations to make a better model.

Our latest models' features are as follows:

All variables have p-value < 0.05.

All the features have very low VIF values, meaning, there is not much multicollinearity among the features as per heat map.

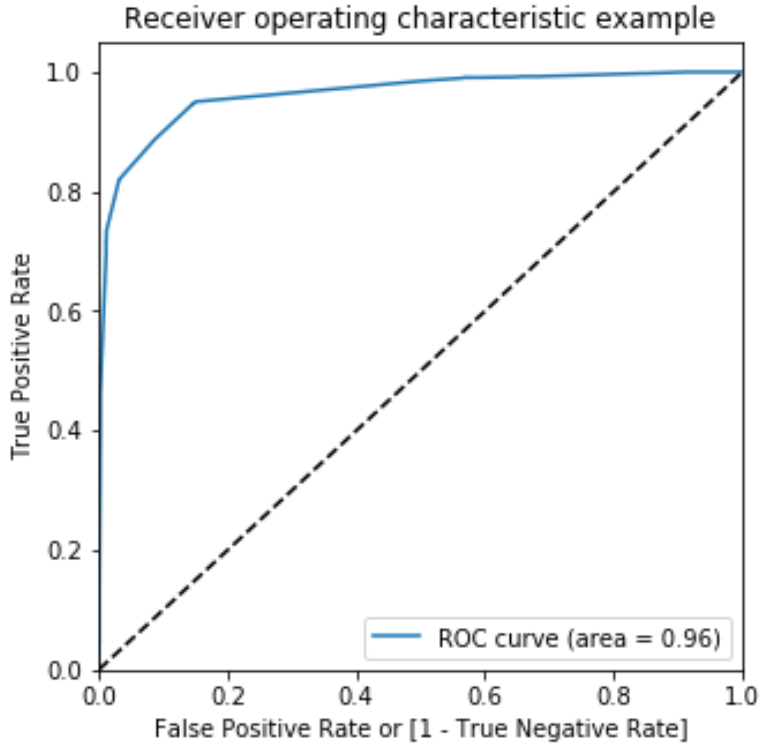
The overall accuracy of 0.9125 at a probability threshold of 0.05 is also very acceptable.

So, we need not drop any more variables and we can proceed with making predictions using this model. We will be using the efficiency matrices to know the performance of our model.

Calculating sensitivity and specificity

Please refer the jupyter notebook for the results sensitivity and specificity.

ROC Curve



We have got the above-mentioned ROC curve. Please refer to jupyter notebook for more details.

Finding the probability cut-off point

We have now created columns based on the probability of the respective leads, so that we can calculate the cut off based on the meeting point of the plots of accuracy, sensitivity and specificity. For more details please refer the jupyter notebook. 0.33 is the optimal cut off.

Precision and Recall

After plotting the value of precision and recall we get the optimal cut off point of 0.37. For more details please refer to the jupyter notebook.

Making predictions on the test set

We now calculate the respective matrix for calculating the efficiency of the model using the test set data. Please refer to the jupyter notebook for more details.

ROC Curve for test data set

We have drawn the ROC curve for test data set and found the area under the curve to be 0.97 which is an excellent category. For more details please refer to the jupyter notebook.

Lead Score for the complete dataset

We finally define the lead score on the basis of converted and predicted values and store in the dataframe. Please refer to the jupyter notebook for more details.

Important Features

We finally create the graph for the most important features in the model and publish the top 3 features. For more details please refer to the jupyter notebook.

Summary

By creating this logistic regression, we are able to predict about the lead that can be converted to a final monetary benefit to the company. In this case study we attached the probability factor to each and every lead according to the various features data we have. That probability factor is the main predictor which predicts the lead to be converted. The efficiency of this model is really good as per the ROC curve.