**Lead Scoring Case Study Summary**

**Problem statement: -**

An education company named X Education sells online courses to industry professionals. The company requires Data Scientist to build a model and play with Leads generated from various sources. Finally, derive a hot leads and convert such potential leads to confirmed opportunity (leads). The target lead conversion rate to be around 80% (by CEO).

**Resolution:**

The best practice is always to start with hot leads i.e. leads that have higher probability of getting converted. This will not only result in higher conversion ratio but also effective use of time. Time spent on nurturing hot leads can be increased whereas time spent on leads with low score (cold leads) can be minimized.

Determining hot and cold leads can be done by using a logistic regression model. We will build a logistic regression model and assign lead score to each lead by using the meta data provided for each lead.

1. [Step 1: Import important libraries](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#1)
   * Read Leads data into dataframe
   * Quick review of dataframe
   * Shape of Leads dataframe
   * Check for conversion rate in dataframe

Basic steps to import necessary libraries. Review dataframe.

1. [Step 2: Analyze data and prepare data](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#2)
   * Check for missing values
   * Drop unwanted columns
   * Fill NA for null values

Following columns contain more than 30% null values initially:

* Tags
* Lead Quality
* Asymmetrique Activity Index
* Asymmetrique Profile Index
* Asymmetrique Activity Score
* Asymmetrique Profile Score

So, we remove all the NULL values with some relevant text and dropped the columns who has values Null values more than 30%

1. [Step 3: Model Building](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#3)
   * Relating all the categorical variables to Converted
   * Univariate Analysis
   * Bivariate Analysis: Categorical variables
   * Checking correlation
   * Outlier Analysis
   * Create dummy variables

Univariate Analysis – Outliers

* Univariate analysis revealed data distribution and outliers in ‘Leads’ data. Key columns where outliers were identified are:-

1. TotalVisits
2. Page Views Per Visit

* Inter Quantile Range (IQR) method has been used to treat outliers in the data.

We decided to remove the outliers

Bivariate Analysis: Categorical variables

* ‘Converted’ column has been chosen as target variable. So, bivariate analysis of important variables has been performed with respect to the target variable.
* Unemployed showing interest so, on next batch has higher chances of getting converted.
* Lead originated through “Landing Page Submission” and “API” has high possibility of getting converted.
* Lead belongs to Email Opened; SMS Sent converts more than any other sources.

Binary Variables Encoding:

* Variables which have binary (Yes/No) values have been encoding with 1 and 0.
* 1 denotes Yes whereas 0 denotes No.

Correlation of the variables:

* From the heatmap, we can see that there are some variables having very high correlation with respect to positive and negative.

Create Dummy Variables:

* Independent variables as dummy variables allows easy interpretation and calculation of the odds ratios, which increases the stability and significance of the coefficients.
* Dummy variables have been created for following columns:

1. Lead Origin
2. Lead Source
3. Last Activity
4. Specialization
5. What is your current occupation
6. Do Not Email
7. A free copy of Mastering The Interview
8. [Step 4: Data Preparation for Modeling](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#4)
   * Train Test split
   * Feature Scaling

The modified ‘Leads’ dataset has been split into Train and test dataset in the ratio 70:30.

Train dataset has been used to train the model whereas Test dataset has been used to evaluate the model

It is important to have all variables on the same scale in order to avoid the dominance of variables with high magnitude in the model.

“MinMaxScaler” function has been used to scale the data for modeling which brings all the data points into a standard.

1. [Step 5: Model Building](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#5)
   * Create a function for model building
   * Assessing the model with StatsModels
   * Prediction
   * Create confusion metrics
   * Metrics beyond simply accuracy
   * Plotting the ROC Curve
   * Finding Optimal Cutoff Point
   * Plot accuracy sensitivity and specificity
   * Observation (Train Data)
   * Precision and Recall
   * Precision and recall tradeoff
   * Making predictions on the test set
   * Observation (Test Data)

**Using logistic Regression:**

* Generalized Linear Model (GLM) from StatsModels library has been used to build the Logistic Regression Model.
* Initially, the model was built using 82 features present in X\_train dataset.
* Most of the features were found to be insignificant. So, we needed to perform feature selection technique.

**Feature Selection using Recursive Feature Elimination (RFE):**

* RFE is an optimization technique for finding the best performing subset of features. It is based on the idea of repeatedly constructing a model and choosing either the best (based on coefficients), setting the feature aside and then repeating the process with the rest of the features. This process is applied until all the features in the dataset are exhausted. Features are then ranked according to when they were eliminated.
* We ran RFE to identify top 15 features for further model building process.
* Insignificant features were dropped one by one after checking the P-value and Variance Inflation Factor (VIF).
* Accepted P-value should be kept below 0.05 and VIF should be less than 5.

**Final Model and Interpretation:**

* Final model contains 13 most important features which satisfy all the selection criteria.
* Lead score having conversion probability greater than 0.35 are being predicted as “Converted”.
* Using this probability threshold value (0.35), the leads from the test dataset have been predicted whether they would get converted or not.
* Confusion matrix with cut-off 0.35 has been created to calculate evaluation metrics.
* Confusion matrix: [[3109 773]

[ 442 1943]]

* Evaluation metrics:

Accuracy: 81% approx.

Sensitivity: 81% approx.

Specificity: 80% approx.

Precision: 77% approx.

**Evaluation Metrics:**

* Receiver Operating Characteristics (ROC) Curve:
  + By determining the Area Under the Curve(AUC) of the ROC curve, the goodness of the model is determined.
  + Since the ROC curve is close to the upper left part of the graph, it means this model is a very good model.
  + The value of AUC for our model is 0.87.
* Plot accuracy sensitivity and specificity:
  + Tradeoff between sensitivity and accuracy can be observed (cutoff = 0.35).
* Precision and Recall plot:
  + Ideal cutoff of 0.43 is observed from recall and precision plot.
* We will use both the cutoff and evaluate results for further predictions.

1. [Step 6: Final Observation](http://localhost:8888/notebooks/Documents/Python%20Scripts/Machine%20Learning%20-%20I/Lead%20Scoring%20Assignment/LeadScore_V1.ipynb#6)

Let us compare the values obtained for Train & Test:

Train Data:

* Accuracy : 80.61%
* Sensitivity : 81.47%
* Specificity : 80.08%

Test Data:

* Accuracy : 80.15%
* Sensitivity : 81.08%
* Specificity : 79.59%

**Conclusion and Recommendations:**

* Followings are top features that contribute to decision which mean the conversion probability of a lead increases with increase in values of these features:
  + Lead Origin
  + Last Notable Activity
  + Last Activity
  + What is your current occupation
* Top three categories that contribute to decision
  + Lead Origin ==> Lead Add Form
  + Total Time Spent on Website
  + Lead Source\_Direct Traffic

**The Model seems to predict the Conversion Rate very well.**

**As per CEO, the target lead conversion rate to be around 80% and we should be able to give the CEO confidence in making good calls based on this model**