Insights & Summary

Reading and Understanding Data :

Observations

- A large number of columns have null values. Those columns should ideally be dropped
- Prospect ID and Lead Number both serve the same purpose. They are both unique identifiers. We will drop Prospect ID
- Column names are just too long. We will modify the column names
- Few categorical columns have "Select" in their entries. Those select are essentially null values because select appears when someone does not select anything from the dropdown.

Data Cleaning

Rename column names

- Long column names make analysis tiring as one has to always refer to column names. Also has impact on charts created later on
- Ideally, we should follow python's preferred Snakecase nomenclature.

Replace "Select" category with null values

- There are 4 columns that contains Select, which are effectively null values. We are going to make that change.

Handle null values and sales generated columns

- Given there are a number of columns with very high number of null entries, let's calculate the percentage of null values in each column, and take a decision from there.
- Furthermore, we can also drop Sales generated columns because those are the data entries that are made after the sales team has connected with the

student. Those data have no bearing to the purpose of our model ie. providing lead score. The columns are

- tags
- lead_quality
- all asymmetrique columns
- last_activity
- last_notable_activity

Observation:

As can be seen, there are quite a few columns with high number of missing data. Since there are no ways to get data back from reliable sources, we can drop all those columns that have missing values > 40%.

Drop columns that have null values > 40% or Sales generated columns

Observations

There are five columns that still have high null values: country, specialization, occupation, course_selection_reason, and city. We will look at them individually to see what can be done.

Country column

Observation

The distribution of the data is very heavily skewed, with India + null values = 97% of the total. It is safe to drop this column.

course_selection_reason column

- The distribution of the data is very heavily skewed, with Better career prospects + null values = approx 100% of the total. It is safe to drop this column.

occupation column

Observation

For occupation, we can first combine categories, and then impute proportionally to maintain the distribution and not introduce bias.

specialization column

Observation

For specialization, we can first combine categories based on the course type, and then impute proportionally to maintain the distribution and not introduce bias.

city column

Observations

We will categorize cities based on logical decisions and impute proportionatel.

Handle categorical columns with low number of missing values and low representation of categories

In this step, we will go through the rest of the categorical columns one by one and

- Merge categories that have low representation
- Impute the missing values

Observation

As can be seen from the above output, the categorical columns (i.e. number of unique values > 2) are:

- lead_origin

- lead_source

Handle Binary columns

- Drop those columns that have significant data imbalance
- Drop all those columns that have only 1 unique entry

Observation

- The following columns can be dropped as they have just 1 unique values
    - magazine
    - course_updates
    - supply_chain_content_updates
    - dm_content_updates
    - cheque_payment

Let's now check the data imbalance for the rest of the columns

Observations

Because of heavy data imbalance, we can drop the following columns as well

- do_not_call
- search
- newspaper_article
- x_education_forums
- newspaper
- digital_advertisement
- through_recommendations

Handle Numerical columns

lead_number column: change datatype

lead_number column is a unique identifier for each leads. Therefore, aggregations won't be of any relevance. We should change it to object.

total_visits column

For this column, we need to handle the missing values, and can convert the datatype to integer since visits can't be decimal.

page_views_per_visit column

Handle missing values

Exploratory Data Analysis

Numerical columns

Observations

- High peaks and skewed data. There might be a possibility of outliers. We will check them next.

Heatmap

Observations:

No significant correlation such that columns can be dropped.

Check for outliers

Observations

- Looking at both the box plots and the statistics, there are upper bound outliers in both total_visits and page_views_per_visit columns. We can also see that the data can be capped at 99 percentile.

Specialization

- Most of the speciliazation taken are management.

Occupation

- Unemployed users are the most significant leads.

City

- Mumbai in particular and Maharashtra in general dominates the lead. This is likely due to the fact that the courses are based in Mumbai.

Creating dummy variable for categorical columns

- Categorical columns are: lead_origin, lead_source, specialization, occupation, city.

Outliers Treatment

As we can see, we were able to significantly reduce the number of outliers by capping.

Lets create empty lists of categorical columns and numerical columns, then we will review the columns one by one and see what needs to be done with each of them.

we'll leave this column as is for now, and add this to a new list of binary categorical variables.

Considering the values and outliers here, Let's cap the values at 96th percentile i.e. 10

Also, we will impute the null values with 0 here, we could impute this with median but considering the values might not have been tracked because they haven't logged on to the site, and 0 is also the mode of the column.

Let's also create a function to cap the outliers if needed in future.

Since we also look at the boxplots and percentiles for numberical numbers, lets create a similar details function to include these two pieces of information.

There are 27 percent null values in this column, and other values are mostly India, so this column would not be that useful,
we could create a binary column using this like 'India' if there weren't so many null values here. but considering the null values, lets drop this column.

There are too many null values in this as well, and the most values are also not selected I think, because the option says default value 'Select', lets drop this column as well.

All these columns have high imbalance, and mostly have only one value i.e. No. So it doesn't make sense to keep these column,
Lets delete these column.

Do not email column

details('Do Not Email')
we'll leave this column as is for now, and add this to a new list of binary categorical variables.

There are 27 percent null values in this column, and other values are mostly India, so this column would not be that useful, we could create a binary column using this like 'India' if there weren't so many null values here. but considering the null values, lets drop this column.

We Can append the Do Not Call Column to the list of Columns to be Dropped since > 90% is of only one Value.

Numerical Attributes Analysis

We can see presence of outliers here

Since there are no major Outliers for the above variable we don't do any Outlier Treatment for this above Column

Check for Page Views Per Visit:

Inference

Median for converted and not converted leads are the close. Nothing conclusive can be said on the basis of Total Visits.

Inference

- Leads spending more time on the website are more likely to be converted.
- Website should be made more engaging to make leads spend more time..
- Median for converted and unconverted leads is the same. Nothing can be said specifically for lead conversion from Page Views Per Visi.
- There are no missing values in the columns to be analysed further.

Model Building using Stats Model & RFE:

There is a high correlation between two variables so we drop the variable with the higher valued VIF value.

So the Values all seem to be in order so now, Moving on to derive the Probabilities, Lead Score, Predictions on Train Data:

Observation:

After running the model on the Test Data these are the figures we obtain:

Accuracy : 72.40% **Sensitivity :** 89.26**%** **Specificity :** 70.92%

Final Observation:

Let us compare the values obtained for Train & Test:

Train Data:

Accuracy : 72.40% **Sensitivity :** 89.26% **Specificity :** 70.92%

Test Data:

Accuracy : 72.48% **Sensitivity :** 89.98**% Specificity :** 70.86%