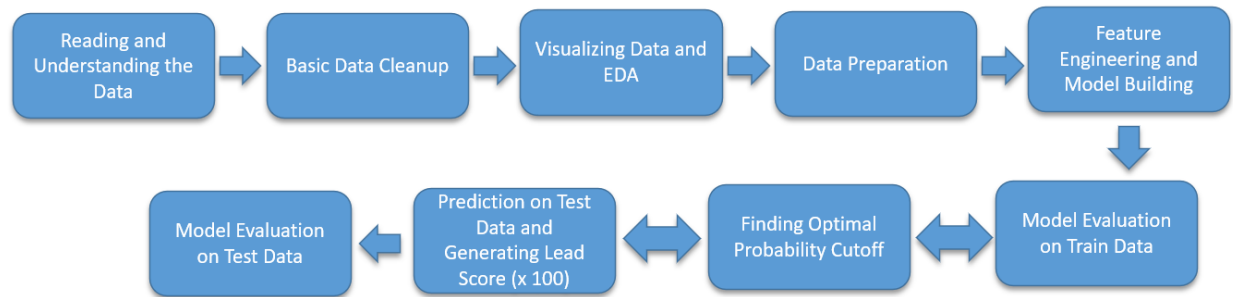


# Summary Report

---



## 1. Reading and Understanding the Data:

Initial data with 9240 records in leads.csv file has 37 columns which include 30 categorical and 7 numerical columns are available.

## 2. Basic Data Clean up:

- As 'Select' is not a valid class, we can conclude that the Select might be the default value set in the form dropdowns. We replaced 'Select' with NaN.
- Columns having only one unique value does not have any variance, hence we dropped these columns.
- Dropped the columns having more than 40% missing value.
- Created new buckets/bins for the categorical variables having very high numbers of classes with few datapoints.
- Performed missing value treatment using **Business Understanding**. For **Specialization** and **Occupation** NaN values are replaced with a new category **Not Disclosed**.
- Renamed some column names to simpler names for convenience during EDA and Model building.

## 3. Visualizing Data and EDA

- Box Plot of TotalVisits, Total Time Spent on Website, Page Views Per Visit.
- Pair Plot of all Numeric variables.
- Count Plot of different categorical variables with Converted as label.

Based on the plot we derived inferences and mentioned that in the PPT and the Jupyter Notebook.

#### 4. Data Preparation:

- **Outlier Treatment:** By observing box plot and calculating different percentile values, identified 2.8% of total data (< 5%) as outliers and removed those rows.
- **Train-Test Split:** Dataset has been split into Train and Test in 70:30 ratio.
- **Missing Value Imputation (Statistical Imputation):** Calculated median, mode on Train dataset. Used that value to impute missing values in Train and Test Dataset. Performed Mode Imputation for Categorical columns and Median imputation for Numeric variables.
- **Categorical Variables Encoding:**
  - Columns having binary classes replaced with 0,1
  - Dummy variables (with drop\_first=True) have been created for categorical columns having more than 2 classes.
- **Performed MinMax Scaling** on Train data(other than dummy).
- **Performed Variance Thresholding**, removed columns having lower variance than threshold=.001
- **Created correlation heatmap** and dropped variables having higher correlations.

#### 5. Feature Engineering and Model Building

- RFE has been used to get top 16 features and built 1<sup>st</sup> LogisticRegression model.
- Then manually eliminated the features one by one. Total 7 models were built and after each model building p-values of all beta-coefficients and VIFs have been checked simultaneously, identified feature has been excluded in next model. Accepted p-value is lower than .05 and VIF < 5.
- Checked Overall model accuracy, Confusion Matrix after each new model, to understand how the new model is performing in compared to the previous one.

#### 6. Prediction & Model Evaluation: (on Training data with cutoff .5)

- Model 7 has been used to predict the probability on training dataset and then used .5 as probability cut off to calculate our target (0 or 1).
- Calculated different evaluation metrics as below:

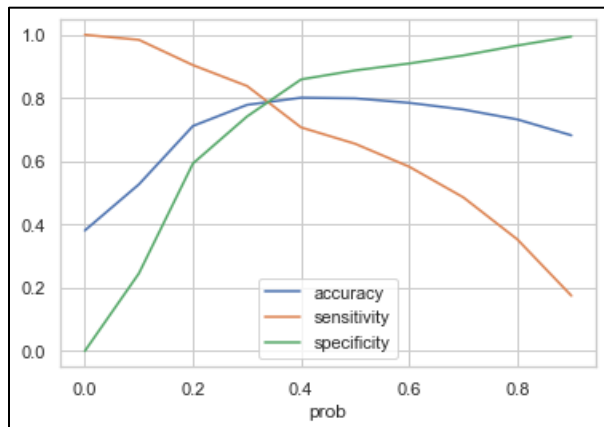
```
Overall model accuracy: 0.7989821882951654
Sensitivity / Recall: 0.6554025865665415
Specificity: 0.887432536622976
False Positive Rate: 0.1125674633770239
Positive Predictive Value: 0.7819810851169736
Positive Predictive Value: 0.8069642439822389
```

```
Confusion Matrix:
True Negative: 3453      False Positive: 438
False Negative: 826      True Positive: 1571

Overall model accuracy: 0.7989821882951654
```

## 7. Finding Optimal Probability cutoff & Evaluating on Train Data

- Calculated specificity, sensitivity, and accuracy for our model for different cut-off probabilities and then plotted that in below graph. From the graph we got optimal probability cutoff = .32.



## 8. Prediction on Test Data & Generating Lead Score

- Performed MinMax Scaling on Test Data (only Transform) and kept only those columns which are present as predictor variables for final model.
- Using Model 7 we calculated the probability on Test dataset and used cutoff = .32 to predict the target (0,1). Created a column **Lead Score** (between 0 to 100) by doing **prob\*100**. A higher score means hot lead, lower score implies cold lead.

## 9. Model Evaluation on Test data & Interpretation

Calculated evaluation metrics on test data.

```
Model Evaluation Metrics on Test dataset
#####
Confusion Matrix:
True Negative: 1258      False Positive: 402
False Negative: 203      True Positive: 832

Overall model accuracy: 0.7755102040816326
Sensitivity / Recall: 0.8038647342995169
Specificity: 0.7578313253012048
False Positive Rate: 0.2421686746987952
Positive Predictive Value: 0.6742301458670988
Positive Predictive Value: 0.8610540725530459
```

Top 3 variables which contribute most towards the probability of a lead getting converted:

- **Total Time Spent on Website**
- **What is your current occupation (Working Professional)**
- **Lead origin (Other)**