**1. Which are the top three variables in your model which contribute most towards the probability of a lead getting converted?**

**Ans) Top Three Variables Contributing Most to Conversion Probability:**

To identify the top three variables contributing most towards the probability of a lead getting converted, you typically look at the coefficients (if using logistic regression) or feature importances (if using other models like decision trees or ensemble methods). Here’s a general approach assuming you have already trained your model and extracted feature importances or coefficients:

1. **Identify Feature Importance**: Determine the importance of each feature in your model. For logistic regression, this is typically done through the coefficients.
2. **Rank Features**: Sort the features based on their importance values.
3. **Select Top Three**: Choose the top three features with the highest coefficients or importances.

Here’s an example code snippet to illustrate this (assuming you have trained a logistic regression model):

***import pandas as pd***

***from sklearn.linear\_model import LogisticRegression***

***# Example DataFrame (replace with your actual data)***

***data = pd.DataFrame({***

***'TotalVisits': [5, 3, 10, 3, 8, 12, 6, 4, 9, 7],***

***'Page Views Per Visit': [2.5, 3.0, 1.5, 2.0, 2.8, 1.2, 3.5, 2.0, 1.8, 2.2],***

***'Converted': [1, 0, 1, 0, 1, 1, 0, 0, 1, 1] # Example target variable***

***})***

***# Separate features (X) and target variable (y)***

***X = data.drop('Converted', axis=1) # Features***

***y = data['Converted'] # Target variable***

***# Example: Train a Logistic Regression model***

***model = LogisticRegression(random\_state=42)***

***model.fit(X, y)***

***# Get coefficients and sort them***

***coefficients = model.coef\_[0]***

***feature\_names = X.columns***

***feature\_importance = pd.DataFrame({'Feature': feature\_names, 'Importance': coefficients}).sort\_values(by='Importance', ascending=False)***

***# Print top three features***

***print("Top Three Variables Contributing Most to Conversion Probability:")***

***print(feature\_importance.head(3))***

**2. What are the top 3 categorical/dummy variables in the model which should be focused the most on in order to increase the probability of lead conversion?**

**Ans) Top Three Categorical/Dummy Variables to Focus On:**

To determine the top three categorical or dummy variables that have the most impact on lead conversion, you would typically look at the importance of these variables in the model. For logistic regression, this might involve examining the coefficients associated with each categorical variable.

Here’s a general approach:

1. **Encode Categorical Variables**: Ensure your categorical variables are properly encoded (e.g., one-hot encoded if necessary).
2. **Train Model**: Train your logistic regression model or any other model you've selected.
3. **Interpret Coefficients**: Examine the coefficients or feature importances related to each categorical variable.
4. **Select Top Three**: Choose the top three categorical variables with the highest coefficients or importances.

Here's an example continuation of the previous code snippet to include categorical variables:

***python***

***Copy code***

***# Example DataFrame (include categorical variables)***

***data = pd.DataFrame({***

***'TotalVisits': [5, 3, 10, 3, 8, 12, 6, 4, 9, 7],***

***'Page Views Per Visit': [2.5, 3.0, 1.5, 2.0, 2.8, 1.2, 3.5, 2.0, 1.8, 2.2],***

***'Lead Source': ['Google', 'Direct', 'Organic Search', 'Direct', 'Google', 'Organic Search', 'Referral', 'Referral', 'Google', 'Direct'],***

***'Last Activity': ['Email Opened', 'SMS Sent', 'SMS Sent', 'Email Opened', 'Email Opened', 'SMS Sent', 'Email Opened', 'SMS Sent', 'SMS Sent', 'Email Opened'],***

***'Converted': [1, 0, 1, 0, 1, 1, 0, 0, 1, 1] # Example target variable***

***})***

***# Assuming 'Lead Source' and 'Last Activity' are categorical variables***

***X = pd.get\_dummies(data.drop('Converted', axis=1), drop\_first=True)***

***y = data['Converted']***

***# Example: Train a Logistic Regression model***

***model = LogisticRegression(random\_state=42)***

***model.fit(X, y)***

***# Get coefficients and sort them***

***coefficients = model.coef\_[0]***

***feature\_names = X.columns***

***feature\_importance = pd.DataFrame({'Feature': feature\_names, 'Importance': coefficients}).sort\_values(by='Importance', ascending=False)***

***# Print top three categorical variables***

***categorical\_features = [col.replace('\_', ' ').title() for col in feature\_importance['Feature'].values if '\_' in col]***

***print("Top Three Categorical/Dummy Variables Contributing Most to Conversion Probability:")***

***print(categorical\_features[:3])***

**3. X Education has a period of 2 months every year during which they hire some interns. The sales team, in particular, has around 10 interns allotted to them. So during this phase, they wish to make the lead conversion more aggressive. So they want almost all of the potential leads (i.e. the customers who have been predicted as 1 by the model) to be converted and hence, want to make phone calls to as much of such people as possible. Suggest a good strategy they should employ at this stage.**

**Ans) Strategy for Aggressive Lead Conversion During Intern Hiring Period:**

During the intern hiring period when X Education wants to aggressively convert potential leads (predicted as 1 by the model), a strategy involving proactive phone calls can be effective. Here's a suggested strategy:

* **Phone Call Strategy**: Allocate the interns to actively reach out via phone calls to all potential leads predicted as 1 by the model.
* **Personalized Approach**: Ensure interns are trained to personalize their approach based on the lead's profile and previous interactions (if any).
* **Follow-Up**: Implement a structured follow-up system to ensure no potential lead is left without a follow-up call.
* **Feedback Loop**: Establish a feedback loop where interns can provide insights into lead responses, which can be used to refine the approach further.

**4. Similarly, at times, the company reaches its target for a quarter before the deadline. During this time, the company wants the sales team to focus on some new work as well. So during this time, the company’s aim is to not make phone calls unless it’s extremely necessary, i.e. they want to minimize the rate of useless phone calls. Suggest a strategy they should employ at this stage.**

**Ans) Strategy for Minimizing Useless Phone Calls After Meeting Quarterly Targets:**

After reaching quarterly targets ahead of schedule, X Education aims to minimize unnecessary phone calls. Here's a recommended strategy:

* **Lead Scoring Refinement**: Refine the lead scoring model to prioritize leads more likely to convert based on current data and trends.
* **Segmentation**: Segment leads based on their engagement levels and likelihood to convert, focusing resources on higher-potential leads.
* **Automation**: Implement automated email campaigns or messages for lower-priority leads instead of immediate phone calls.
* **Customer Relationship Management (CRM)**: Use CRM tools to track interactions and preferences, ensuring communications are relevant and timely.

**THANK YOU**