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

   Ans: Based on the coefficient values from below list of variables contributing to the final model, the following are the top three variables that contribute most towards the probability of a lead getting converted. These are also all categorical variables.

   ✓ Lead Origin: Has 2 sub categories Lead Add Form, Lead Import contributing most in lead conversion.

   ✓ Last Activity: Has 3 sub-categories SMS_Sent, Unsubscribed, others contributing most in lead conversion.

   ✓ Lead Source: Has 2 sub-categories Welingak Website , Olark Chat contributing to Lead Conversion.

```
Lead Origin_Lead Add Form                        3.908569
Last Activity_Others                             2.281693
Lead Source_Welingak Website                     2.012467
Lead Origin_Lead Import                          1.477128
const                                            1.471697
Last Activity_Unsubscribed                       1.435453
Last Activity_SMS Sent                           1.294636
Lead Source_Olark Chat                           1.238324
Total Time Spent on Website                      1.127928
Last Notable Activity_Modified                  -0.896926
Last Activity_Olark Chat Conversation           -0.970622
Do Not Email_Yes                                -1.704308
What is your current occupation_Student         -2.272949
What is your current occupation_Unemployed      -2.686516
dtype: float64
```

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: Based on the highest coefficient values from the screen shot in the question above, the following are the top three categorical/dummy variables that should be focused the most in order to increase the probability of lead conversion :

- ✓ Lead Origin_Lead Add Form
- ✓ Last Activity_Others
- ✓ Lead Source_Welingak Website

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.

Solution:
The Business can follow the below strategy during the 2 months period when they hire 10 interns. Now they have 2723 customers with the convert to predict flag helping them to target the customers to call and nurture to convert.

When "convert to predict" flag is set to 1 – means this customer is a hot lead.

This strategy will pull up more customers as target customers as we have lowered the cut-off value to 0.1 this will bring more customers under the Hot Leads bucket..

To help business look at these numbers we have built a reusable code block in the python file wherin the business can adjust the cut-off values as below.

```
build_model_cutoff(X_train[drop_col], y_train, X_test[drop_col], y_test, cutoff=0.1)
```

```
------------------Result of training data--------------------
    Prospect ID  Converted  Convert_Probability  Convert_predicted  Lead_Score
0          3009          0            0.198519                  1          20
1          1012          0            0.301167                  1          30
2          9226          0            0.358675                  1          36
3          4750          1            0.873619                  1          87
4          7987          1            0.811317                  1          81
------------------Result of test data--------------------
    Prospect ID  Converted  Convert_Probability  Convert_predicted  Lead_Score
0          3271          0            0.130989                  1          13
1          1490          1            0.973415                  1          97
2          7936          0            0.112644                  1          11
3          4216          1            0.769955                  1          77
4          3830          0            0.132646                  1          13
-----------------Model Evaluation Metrics--------------------
Confusion Matrix :
 [[ 602 1132]
 [  37  952]]
Accuracy : 0.570694087403599
Sensitivity : 0.9625884732052579
Specificity : 0.34717416378316035
Precision : 0.45681381957773515
```

	Prospect ID	Converted	Convert_Probability	Convert_predicted	Lead_Score
0	3271	0	0.130989	1	13
1	1490	1	0.973415	1	97
2	7936	0	0.112644	1	11
3	4216	1	0.769955	1	77
4	3830	0	0.132646	1	13
...	...	...	...	...	...
2718	850	0	0.198056	1	20
2719	2879	0	0.130657	1	13
2720	6501	1	0.904977	1	90
2721	7155	0	0.114224	1	11
2722	376	0	0.087525	0	9

2723 rows × 5 columns

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.

Solution:
The Business can follow the below strategy during the times when the company reaches its target for a quarter before the deadline.
As the company wants the sales team to focus on some new work as well, we have set the cut-off value higher which gives us leads on top most category, with convert to predict flag helping them to target the customers to call and nurture to convert.

When "convert to predict" flag is set to 1 – means this customer is a hot lead.

This strategy will pull up less customers as target customers as we have increased the cut-off value. This will have less number of customers under the Hot Leads bucket.

To help business look at these numbers we have built a reusable code block in the python file wherein the business can adjust the cut-off values as below.

```
build_model_cutoff(X_train[drop_col], y_train, X_test[drop_col], y_test, cutoff=0.9)
```

```
------------------Result of training data-------------------
    Prospect ID  Converted  Convert_Probability  Convert_predicted  Lead_Score
0          3009          0             0.198519                  0          20
1          1012          0             0.301167                  0          30
2          9226          0             0.358675                  0          36
3          4750          1             0.873619                  0          87
4          7987          1             0.811317                  0          81
------------------Result of test data-------------------
    Prospect ID  Converted  Convert_Probability  Convert_predicted  Lead_Score
0          3271          0             0.130989                  0          13
1          1490          1             0.973415                  1          97
2          7936          0             0.112644                  0          11
3          4216          1             0.769955                  0          77
4          3830          0             0.132646                  0          13
------------------Model Evaluation Metrics-------------------
Confusion Matrix :
 [[1719   15]
 [ 760  229]]
Accuracy : 0.7153874403231729
Sensitivity : 0.23154701718907988
Specificity : 0.9913494809688581
Precision : 0.9385245901639344
```

	Prospect ID	Converted	Convert_Probability	Convert_predicted	Lead_Score
0	3271	0	0.130989	0	13
1	1490	1	0.973415	1	97
2	7936	0	0.112644	0	11
3	4216	1	0.769955	0	77
4	3830	0	0.132646	0	13
...	...	...	...	...	...
2718	850	0	0.198056	0	20
2719	2879	0	0.130657	0	13
2720	6501	1	0.904977	1	90
2721	7155	0	0.114224	0	11
2722	376	0	0.087525	0	9

2723 rows × 5 columns