On Predicting the Behavior of Online Customers

Amirpasha Shirazinia
Email: amirpasha.shirazinia@gmail.com

August 17, 2016

Abstract

In this report, we analyze the behavior of typical online customers with respect to accepting call-in offers provided by a website. For this purpose, we use a machine learning algorithm, and, in particular, the *logistic regression* model, and evaluates its efficiency.

1 Data Collection & Primarily Statistics

The dataset, in csv format, is provided by *Now Interact* which consists of actual online behaviors from a website with a number of different contact offers. The amount of data that we investigate adds up to 67 MB of storage. Table 1 shows some general statistics extracted from the dataset under consideration.

Table 1: Dataset statistics

# Unique customers	# Total visits	# Unique visits (%)	# Accepted offers (%)
298 745	1 742 450	197 903 (66%)	5 621 (1.188 %)

2 Model Characteristics

First and foremost, we should choose a proper *predictive model* to predict customer behavior. We adopt our model based on the outcome score (Contact variable) which is a binary 0/1-level. Hence, one simple choice is the binary *logistic regression* model. Note that, here, linear regression model might not provide an accurate performance due to the binary level outcome.

Our next task is the *feature selection* stage, i.e., what feature or predictor to choose in order to have significant impact on the outcome. For this purpose, we select three numeric features:

1. The total number of times a visitor visits the website (TotalVisits). This feature shows roughly how long a visitor spends time on her/his visit.
2. The number of clicks a typical customer makes before displaying a contact offer (ClicksBeforeThisDisplay). This feature can be thought of as page depth.
3. Number of previous contact offers during a session (PreviousDisplaysThisSession). This feature qualitatively shows the insistence of the website in contact offers.

Next, let us denote the mentioned features (predictors) by $\mathbf{x}_1$ (TotalVisits), $\mathbf{x}_2$ (ClicksBeforeThisDisplay) and $\mathbf{x}_3$ (PreviousDisplaysThisSession), where $\mathbf{x}_j \in \mathbb{R}^m$ $(j = 1, 2, 3)$ and $m = 298745$ is the number of training samples. Further, by $0 \leq p \leq 1$ we denote the probability of outcome score (Contact) being 1. Hence, the logistic regression follows the following equation model:

$$\text{logit}(p) \triangleq \ln\left(\frac{p}{1-p}\right) = \theta_0 + \theta_1\mathbf{x}_1 + \theta_2\mathbf{x}_2 + \theta_3\mathbf{x}_3 \tag{1}$$

Here, θ_j is a coefficient to be fitted (optimized) using the maximum likelihood estimation method. We will evaluate these coefficients and interpret them in the next section.

3 Model Evaluation and Interpretations

We use the software *R* in order to optimize the logistic regression model parameters via the generalized linear model. The following table summarizes the results of the optimization. Now, we describe each column of the table.

Table 2: Optimized coefficiets

Coefficients (θ_j)	Estimates ($\hat{\theta}_j$)	Standard deviation (σ_j)	Z-value	Odds ratio (OR)
θ_0 (Intercept)	-3.811969	0.026382	-144.491	0.02210461
θ_1 (TotalVisits)	-0.063383	0.005969	-10.619	0.93858434
θ_2 (ClicksBeforeThisDisplay)	0.022506	0.005225	4.308	1.02276108
θ_3 (PreviousDisplaysThisSession)	-0.149249	0.013469	-11.081	0.86135485

The first column represents the model coefficients to be optimized. The second column of the table provides the estimates of the fitted coefficients, where θ_0 is the intercept, and the third column gives the standard deviation of the estimated coefficients. In the fourth column, we provide the Z-value (aka Wald statistic) corresponding to each estimated coefficient, which is defined as follows:

$$Z_j = \frac{\hat{\theta}_j}{\sigma_j}. \tag{2}$$

As can be seen from Table 2, the magnitude of Z-values (either positive or negative) is relatively large, which indicate that string evidence that the feature variables matter in outcome level. In addition, it indicates that the feature variables are unremarkably correlated.

Another important factor characterizing our model is the Odds ratio (OR) in the last column of Table 2. In fact, it reveals the effect of a one-unit increase in the predictor variable on the outcome probability. Mathematically, for each predictor variable, it follows that

$$\mathrm{OR}_j = \frac{p}{1-p} = e^{\hat{\theta}_j}\ , \quad j = 0, 1, 2, 3. \tag{3}$$

Based on OR_j, we can give the following insights:

- With one increase in TotalVistis, the ratio between accepted offer and non-accepted offer is approximately 0.94.
- With one increase in ClicksBeforeThisDisplay, the ratio between accepted offer and non-accepted offer is approximately 1.023.
- With one increase in PreviousDisplaysThisSession, the ratio between accepted offer and non-accepted offer is approximately 0.86.

To sum up, with respect to customer behavior, having more TotalVisits and PreviousDisplaysThisSession decreases the success probability (i.e., accepted offer) to lower than 50%, and having more ClicksBeforeThisDisplay increases the success probability (i.e., accepted offer) to more than 50%. Among these variables, the effect of PreviousDisplaysThisSession is more crucial due to its value being farther away from 1.

These findings can be interpreted with regards to the state of a customer's mind. For example, a customer with very large *number of visits* or *previously-shown an offer* is not likely to accept an offer. Perhaps, psychologically, it is due to the fact that she/he is fed up with large number of contact offers. Thus, it is more likely for a customer to *accept an offer at early stage of her/his visit*, and as a result these *focused* customers are *worthier* (as time is treasure) to be invested on or engaged in.

Finally, the accuracy of our model can be tested using different methods (e.g., R^2 test, Hosmer-Lemeshow test, etc.), other than the Wald test, i.e., Z-value (or its probability). In addition, in order to test the accuracy of the method, this model should be tested against test data which was not the focus of this assignment.

4 Relevance and Business Impacts

Web analytics enables to track customers' browsing behavior and help understand the browsing pattern. Further, understanding the behavior/pattern allows us to predict where an online customer is in the purchase cycle.

The proposed predictive model can be utilized for several purposes based on the business need, where, here, the purpose was customer engagement in real-time. Such model, though simple, can *more or less* – based on our findings in the previous section – track the customers' behavior, and some interesting insights were highlighted in the previous section. The rational behind our model choice was its simplicity, and low number of variables to optimize. However, the question is: Is more or less enough or even good? Well, the answer to this question might not seem very straightforward.

To offer some insights, it should be noted that we only chose three features in our model, and in order to cover more generic business needs, one need to consider various types of predictors. For example, geographical data of an online customer or device type, e.g., desktop, laptop or mobile, visitor type, e.g., organic or reference, when to engage a customer, or even landing/exit page (A/B testing). These features are also used in Google Analytics. We can also employ more complex models such as decision trees or deep learning methods (such as neural networks). The latter, though complex, is a powerful tool for making predictions as well.

5 Final Notes

This report has been completed through the following steps:

1. Importing the dataset, choosing the right predictive model and analyzing of the model (using R)[1].

2. Completing the report and finding the future works and business impacts.

Each part was carried out in less than 2 hours, and in total this report was completed in about 4 hours. The assignment was very-well explained and the tasks were to the point, and could be addressed without any major difficulties.

[1] The codes are available upon request