Prediction Errors

Gaurav Sood

September 1, 2018

Say you want to measure the how often people visit pornographic domains over some period. To measure that, you build a model to predict whether or not a domain hosts pornography. And let's assume that for the chosen classification threshold, the False Positive rate (FP) is 10% and the False Negative rate (FN) is 7%. Here below, we discuss some of the concerns with using scores from such a model and discuss ways to address the issues.

Let's get some notation out of the way. Let's say that we have n users and that we can iterate over them using i. Let's denote the total number of unique domains—domains visited by any of the n users at least once during the observation window—by k. And let's use j to iterate over the domains. Let's denote the number of visits to domain j by user i by $c_{ij} = 0, 1, 2, ...$. And let's denote the total number of unique domains a person visits ($\sum(c_{ij} == 1)$) using t_i. Lastly, let's denote predicted labels about whether or not each domain hosts pornography by p, so we have $p_1, ..., p_j, ..., p_k$.

Let's start with a simple point. Say there are 5 domains with p: $1_1, 1_2, 1_3, 1_4, 1_5$. Let's say user one visits the first three sites once and let's say that user two visits all five sites once. Given 10% of the predictions are false positives, the total measurement error in user one's score $= 3 * .10$ and the total measurement error in user two's score $= 5 * .10$. The general point is that total false positives increase as a function of predicted $1s$. And the total number of false negative increase as the number of predicted $0s$. More generally, the total error for user i is:

$$\sum_{1}^{k} c_{ij} * (p_j == 1) * (FP) - c_{ij} * (p_j == 0) * (FN) \tag{1}$$

Formalizing clarifies three simple things. First, the net error is a function of $FP - FN$. Second, even when the share of visits to pornographic domains is the same, the larger the number of domains (t_i) a person visits, greater the error in their score (total number of visits to pornographic domains). Third, when c_{ij} are right-skewed, e.g., browsing data, errors in the right tail can be very costly. Concretely, misclassifying domains that people visit a lot can be super expensive—it may even change inferences wholesale.

One way to speak to the first two issues is to use different probability cutoffs for classification. Different probability cutoffs generate different FN and FP rates and allow us a way to provide bounds for the inferences.

The third point cuts deeper. To address the issue, one could tweak the cost function of the domain level model such that the cost of each error is proportional to usage. But given the skew, it would put a metric ton of weight on the features of too few domains. And that may mean that the performance of the model is pretty bad. A better, simpler solution may be to use the labels from the dataset used in training. Labeled datasets like the Shallalist cover a vast majority of the heavily visited domains. And using labels from the training set means that we are saved from the most costly errors. Doing so also means that we aren't doing the silly thing of introducing (adding to) measurement error for cases where we have little measurement error.

If still concerned about errors, one could download the top 1M domains from Alexa, take the difference from the original labeled dataset, and for the remaining domains that are also in the universe of domains you are analyzing, use some reputable web service to get the category of content hosted by the domain.