Exploring Medical Appointment No-Shows With Logistic Regression and Association Rule Mining

Daniel Pham
University of California, Riverside
ledanielpham@gmail.com

Viktoriya Nikiforova
University of California, Riverside
vniki001@ucr.edu

Introduction

Any industry that involves the process of scheduling appointments has the common experience of dealing with no-shows. Whether it's restaurant reservations, haircuts, or doctors' offices, it's a problem that both disrespect the time of the office and takes away from another's client's opportunity to be seen. Some medical offices in Brazil have saved data on their patients, detailing whether or not the patient had shown up to the appointment along with other combinations of characteristics. By using data mining methods, it might be possible to find patterns between the patients to potentially explain why they didn't show up or predict future patients' decisions. The two methods being utilized in this analysis will be logarithmic regression and association rule mining.

CCS Concept

• Supervised Learning • Logistic Regression
• Association Rule Mining

Related Works

In Andrew Ng's CS229 regression notes, he goes over the principles on how logistic regression is used and works in practice. He derives the cost function and gradient descent formula that is used to find the optimal minimal value for weights in logistic regression [1]. These formulas are then used for the logistic regression implementation method in this work.

Thabet Slimani and Amor Lazzez's paper on association rule mining explains the various types of association patterns available as data mining techniques. The paper offers association mining as a way to describe relationships between the items through the lens of frequent patterns. This reinforces the decision that the no-show question could be answered using this method. According to the paper, the specificity of the method will be, frequent item-set mining done within single-dimensions with boolean association [2].

Proposed Method

The main question being asked is whether or not if an individual will miss their scheduled appointment. Since the output of the problem is binary in nature, logistic regression is chosen as one of the two methods to be implemented. Logistic regression is a classifier that is used for its predictive analysis to predict a binary outcome from a set of independent variables. In other words, find which one of the binary classes that the sample belongs to by using its features. Weights are assigned to each feature and are calculated during training through gradient descent. Logistic regression returns a probability as its output by using the logistic sigmoid function, which is then mapped between two classes. Generally, probabilities that are 0.5 or over are returned as the positive class, the class we are interested in, while probabilities under 0.5 are returned as the negative class. In terms of logistic regression for the question being asked by the problem, we are looking to create a model that can predict future cases on whether a patient will miss their scheduled appointment or not. If logistic regression is deemed to be a good model for the dataset, we then take a look at the assigned weight for each feature to see what is attributing to the medical appointment no-shows.

The last of our proposed methods is association rule mining. The premise of this technique would be to find frequent patterns within the features associated with the patients that did not show up to their appointments [2]. The idea behind the utilization of this method is to see what features these patients have in common. In this analysis, the features were treated as items in a basket. Its purpose is to find a way to explain the relationships between the items through the search for items frequently found together in the set. The various descriptors that characterize the patients are treated as individual values, so a patient is made up of a combination of features. The goal would be to focus on our specific case of patients that are characterized by not showing up. We don't care about any frequent pattern, we're looking for patterns that only include that main feature. Once those are isolated, we would figure out the qualities most frequently associated with that feature, to understand which ones could explain the problem at hand. After the patterns are developed, then they must be analyzed to make sure the features are actually dependent on each other. This means that even though one item is frequently found alongside the other, it doesn't inherently mean they have any effect on each other. The item could just be a feature that happens to be largely sampled in the data set, meaning that it's just a frequent item for both patients who showed up and those who didn't.

Logistic Regression Evaluation

The logistic regression model is trained by using Andrew Ng's derivation of the cross-entropy equation as the cost function. The provided dataset was split into 2 separate sets: the training set with 80% of the samples and the test set with the remaining 20% samples of the dataset. The training set is used to train the model through gradient descent while the test set is used to test if the model is overfitting to the data. The training and test set accuracy for the dataset are roughly 79.7% and 78.8%, respectively. Since the test accuracy is very similar to the training accuracy, this means that our model is not overfitting to the dataset. Scikit-learn's logistic regression library was then implemented by using the same training and test set. This was done in order to see how our model compares to scikit-learn's model. It had an accuracy of 78.7%. Since our model has similar accuracy to Scikit-learn's, that means the logistic regression model was implemented correctly.

To evaluate the logistic regression model's effectiveness, we implemented cross-validation, confusion matrix, F-scores, and AUC score. Cross-validation was done over 10-folds and it had a minimum of 79.4%, a mean of 79.6%, and a maximum of 79.8%. This indicates that our model would accurately perform well in practice since it has similar accuracy. However, when we take a look at the model's confusion matrix, it shows that the model actually has some poor results and it is reflected in its F-scores. The confusion matrix for both our and scikit-learn's model has an F1-score of 0.88 for the negative class and 0.02 for the positive class. This means that the logistic regression model is doing a very good job of predicting the negative class and a very poor job of predicting the positive class. This is not good because we want to have a high F1- score in the positive class since we want to know why patients miss their scheduled appointments. Taking a look at precision and recall for the positive class, it was 0.28 and 0.01 for our model while scikit-learn's model was 0.25 and 0.01, respectively. These are very low values for the class that we are interested in. To also affirm this, the AUC score from the ROC curve is 0.501. Since the AUC value lies between 0.5 to 1, in which 0.5 denotes a bad classifier while 1.0 denotes an excellent classifier, a score of 0.501 suggests that there is no discrimination in our classifier. This indicates that our classifier is useless and has failed.

This could be a case in which the classifier does not have enough examples to learn the minority class properly since our dataset consists of 20% positive classes. So to see if this was the case, SMOTE sampling was implemented to oversample our positive class till it matches the total amount of negative classes then we repeat the same evaluation methods. Logistic regression with SMOTE sampling has an accuracy of roughly 67.1%. This is a lower score compared to our model's score. The cross-validation scores for minimum, mean, and maximum were 62.0%, 62.9%, and 63.6%, respectively. Those scores indicate that logistic regression through SMOTE sampling would have a decent performance for accuracy since the scores are a bit lower than the model's accuracy. But, when we look at the F1-scores for smote sampling, the negative class has a score of 0.77 while the positive class has a score of 0.42. SMOTE sampling definitely has improved our results for the positive class while it has slightly worsened for the negative class, which was 0.02 and 0.88, respectively. Finally, the AUC score from the ROC curve for SMOTE is 0.632. This suggests that the classifier has a poor performance but it is still better than our implemented classifier with normal sampling.

Association Rule Mining Evaluation

The quality of the evaluation of this method depended on two calculations. The first was the support behind the developed rules. This is a calculation of how frequently this item set is found in the entire set of items. Typically, this value is set to a minimum value so as to have a percentage that the rule has to be above for it to be accepted. Otherwise, the rule doesn't adhere to the "frequent" part of the frequent pattern mining. The second is the lift, which measures how interesting the rule is compared to the null data. This a measure of how reliant the items in the rules are to one another. By these qualifications, the use of association rule mining did work well to find confident patterns in the data set to explain patient "No-Shows".

Frequent items associated with the "No-Show" patients were found but when they were checked for lift, the majority of them were uninteresting. This means the lift was either 1, or very close to 1. The initial support was set to 0.05 but after the developed rules were concluded to be uninteresting, min-support wasn't set to a specific value, and instead, the lift was calculated for all the possible elements that could be associated with the patients. The results for the lift were sorted by ascending values and the extremes on both ends are concluded to be the elements that created the most interesting rules with "no-show"s. The issue that came about with these rules is that the interesting elements had very low values of support. This could be explained by examining the frequency of those specific elements. They tended to be values that were only found a small amount of time throughout the dataset, therefore created a large lift because only the no-show patients contained these elements. There are not enough samples of patients with the found elements to be able to confidently classify them as rules.

Discussion & Conclusions

From the experimental evaluation for logistic regression, we can conclude that logistic regression is a poor classifier for the dataset. This may be due to nonlinearities in the dataset that logistic regression is unable to properly capture or that there may be not enough features in the dataset for logistic regression to capture the relationship between the features and the binary classes. It is important to note that oversampling the minority class through SMOTE did improve the logistic regression model, but it was not enough to make it into a good enough classifier to be used applicably. Nevertheless, we are unable to answer the main question for the dataset in terms of logistic regression as it is deemed as a poor classifier for the dataset.

Based on the evaluation of association rule mining, it could be concluded that this method could not be used to make accurate predictions on account of the small support for the interesting rules found. Perhaps if more data could be collected on people with those specific elements, then there could be a more conclusive understanding of support. With its current set, this method could not explain or predict why patients were not showing up for their appointments with the data.

In the end, logistic regression and association rule mining are just poor model choices for the problem at hand. Perhaps the conclusions made here could be used for assisting future endeavors for this problem.

References

[1]. Andrew Ng. 2012. CS229 lecture notes.
http://cs229. stanford.edu/notes/cs229-notes5.pdf.

[2] Thabet Slimani and Amor Lazzez. 2014. Efficient Analysis of Pattern and Association Rule Mining Approaches. International Journal of Information Technology and Computer Science. Vol.6. MECS Publisher, 70-81. DOI:
http://doi.org/10.5815/ijitcs.2014.03.09