“Predicting No-Shows for Medical Appointments”

- *Nilesh Pawar*

Setting

This project is inspired by the business context that a person makes a doctor appointment, receives the instructions and then either shows up or does not show up on the appointment date. A ‘no-show’ is a hindrance for doctors, patients, and hospital business in general. Patients who need an appointment earlier have to wait longer and the doctor loses time in which s/he could have seen more patients (unless the doctor can find someone to come in place of the no show). It disturbs the system and becomes problematic to the business, as it is happening 30% of time currently. With the aid of business analytics, I want to see if it is possible to predict an appointment to be a “no-show”.

Data

I used the dataset from Kaggle, a platform hosting public datasets. The specific file I used is called “Medical Appointment No Shows”, which contains a “No-show-Issue-Comma-300k.csv” file. The csv file has 300k medical appointments and its 15 variables (features) of each appointment registered. The most important variable, “Status”, indicates whether the patient shows up or not to the appointment. The others are standard information collected, such as appointment registration and appointment dates, day of appointment, sms reminder sent or not and finally, questions related to any medical history that can help such as Diabetes, Hypertension, etc.

Data cleaning was a major challenge of this project. While working on the dataset with 300,000 rows, I first subset it to different trials, or random sample of 10% to run analysis that would not take a long time. I corrected column names, converted the status variable, days of the week and gender to dummy variables. After multiple tests, I decided to drop variables that were less significant to the model. It was time-consuming to decide which variable to use and not to use.

I also decided to delete days of week from the analysis, because I did not see much correlation between them and the dependent variable of the model, and instead has kept a variable stating if it is a weekend, or not, in which it is possible to see important differences between the two categories.

Results

Within the analysis, I narrowed our findings into four particular features, including the age group, days to appointment, whether it is weekend or not and gender. Below you can see five tables with the percentage of show up and no show on every category.

Age Group (years)	**Show Up (%)**
0 - 16	66.93
17 - 30	62.60
31 - 50	68.64
51 - 70	75.90
> 70	77.37

Days to Appointment	**Show Up (%)**
0 - 7	73.72
8 - 30	66.61
31 - 90	64.85
91 - 180	73.47
181 - 365	91.67
> 365	0

Weekend or Not	**Show Up (%)**
Weekend	62.58
Weekday	69.74

Female or Not	**Show Up (%)**
Female	69.78
Male	69.78

The next table shows the list of accuracy scores I obtained from the five algorithms.

Algorithm	**Accuracy Score**
K-nearest neighbors	0.69533
Naive Bayes Classifier	0.655417
Decision Tree	0.572167
Support Vector Machine	0.692500
Random Forest	0.610833

Analysis and discussion

- *How is the data transformed into insights and the algorithms that were used in the process?*

Data Transformation

I transformed non-numerical data such as gender and show-up into dummy variables to include in our model. Next, through the 'Appointment Registration' and 'Appointment Date' values I obtained information such as Day of the Week, Days to Appointment and Time of the Day. These values were calculated as they are significant to determine whether a person shows up to their doctor's appointment or not. For example, some people may have the tendency to not show up on certain days of the week, and also if the days to the appointment is very long (2-3 weeks).

The next stage is to split the data into training and test data in order to train the algorithms and make a prediction on our test data.

Insights

Based on our results, I found the following:

1. Age Group

Elderly people (>51 years) have the highest show-up rates of over 75%. This could be because elderly people may have more serious reasons to visit the doctor, do not want to reschedule and wait longer, and also have more free time.

2. Days to Appointment

It is typical for people for a short appointment date (<7 days) to show up for appointments and I found this to be true in our result as well. The show-up rate drops beyond 7 days and increases significantly for very long days to appointment (>91 days). This could be due to the fear of possibly an even longer wait in-case the patient does not show up.

3. Higher on Weekdays

Comparing to the result for weekend (62.58%), weekdays have a higher rate (69.75%) of showing up. This could be resulted from people being lazy or occupied on weekends.

4. Same for Male and Female

I observed the same rate of showing for appointments for both gender.

Such a predictive model can help the healthcare facility to take appropriate measures to prepare for no-shows and avoid waste of time and resources. This would ensure an increase in efficiency and productivity for the facility.

- ***How well the algorithms perform?***

KNN with an accuracy rate of 69.5% performs the best. The rest perform well ranging from 57% to 69%. In general, the algorithms perform well.

- ***How did you choose which algorithms to execute?***

To conduct the analysis in this project, I chose five different predictive methodologies to obtain the best performer among all the models. The five models are each based on Nearest Neighbors, Naive Bayes classifiers, Decision Tree, Random Forest, and Support Vector Machine. The algorithms were selected due to their feature in predicting; they also have additional features that contribute to better models.

One of our algorithm, Nearest Neighbors, is chosen due to its simplicity and efficiency, as it finds previously defined training samples closest to the new point and predict accordingly. Its prediction capability and simplicity is something I look for dealing with this specific dataset. In addition, nearest neighbors are great with many classification and regression problems. Therefore, I first chose to work with this algorithm, which turned out to produce the highest accuracy.

As our second choice, Naive Bayes classifiers are used extensively in real-world cases, such as document classification. Since Naive Bayes classifiers can be very fast compared to other sophisticated methods, I adopted this method to process the analysis quickly and easily. Decision Tree is to create predictive model for a target variable by learning simple decision rules from the data features. I chose to use this algorithm as it is relatively simple to interpret, and it can handle both numerical and categorical data.

For a highly versatile machine learning method, Random Forest has many applications in terms of modeling regression and classification. Being able to handle many features, random forest is very

helpful to estimate important variables are important in the modeled data. Besides, I decided to also use random forest as I have experienced its strength in predictive modeling.

Support vector machines uses classification and regression analysis under supervised learning models. An SVM algorithm builds a model that is a representation of the examples as points in space, mapped as the separate categories are divided by a clear gap as widely as possible. In addition, SVMs can perform non-linear classification. The accuracy score I obtained from SVM model is the second highest.