Modelling Patient No-Shows

Aman Panwar, Hutan Vahdat, Krishnaveni Sompallae,

Lindsey Peters, Sayo Taiwo

Table of Contents

[Overview 2](#_Toc7806071)

[Dataset 2](#_Toc7806072)

[Description 2](#_Toc7806073)

[Pre-Processing 2](#_Toc7806074)

[Exploratory Analysis 3](#_Toc7806075)

[Models 5](#_Toc7806076)

[Decision Tree 5](#_Toc7806077)

[Logistic Regression 5](#_Toc7806078)

[SVM 6](#_Toc7806079)

[Random Forest 6](#_Toc7806080)

[AdaBoost 7](#_Toc7806081)

[Gradient Boost 7](#_Toc7806082)

[Neural Network 8](#_Toc7806083)

[Evaluation 8](#_Toc7806084)

[Model Comparison and Selection 8](#_Toc7806085)

[Business Perspective 10](#_Toc7806086)

[Technical Improvements 10](#_Toc7806087)

[Business Insights 10](#_Toc7806088)

[Important Factors 10](#_Toc7806089)

[Model Implementation 11](#_Toc7806090)

[Model improvements 11](#_Toc7806091)

# Overview

Patient no-shows are a very popular problem in the healthcare industry. Many studies have been done sighting no-show rates ranging from 5% to more than 30%. A patient no-show is bad for all involved parties. The healthcare organization loses revenue, the patient fails to receive treatment, and the community at large suffers from inefficient utilization of the healthcare system. Our goal is to use a dataset from Brazil to identify which appointments result in no-show (supervised classification problem). While the dataset is from Brazil, many human behaviors transcend borders. The insights gathered through this process can generally be applied to the global issue.

Before we begin modeling, we will walk you through the dataset description, preprocessing and exploratory analysis of our dataset. Exploratory analysis identifies features we expect to be significant factors within our models. Next, we will go through a variety of models designed to help solve classification problems. Because of the skewed nature of our dataset, we will use the area under the ROC curve (AUC) to evaluate and compare each model.

# Dataset

## Description

Data is obtained through Kaggle. It contains 110,527 instances and a breakdown of variables is below.

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| PatientID | Identification | Unique identifier for the patient |
| AppointmentID | Identification | Unique identifier for the appointment |
| Gender | Categorical | Male or Female |
| ScheduledDay | Date | Date the appointment was scheduled |
| AppointmentDay | Date | Date of the appointment |
| Age | Numerical | Age of patient |
| Neighborhood | Categorical | 81 distinct neighborhoods where the patients live |
| Scholarship | Indicator | Whether or not the patient is on a social welfare program |
| Hipertension | Indicator | Whether or not the patient has hypertension |
| Diabetes | Indicator | Whether or not the patient has diabetes |
| Alcoholisim | Indicator | Whether or not the patient is an alcoholic |
| Handicap | Numerical | The number of handicaps the patient has (0-4) |
| SMS\_received | Indicator | Whether or not the patient received a text reminder |
| No-show | Target (Categorical) | Whether or not the patient came to the appointment |

Because the dataset represents less than 2 months of appointments, we created a variable AppointmentLag measuring the number of days between when the appointment was scheduled and the actual date of the appointment. With the limited range of appointment dates, both AppointmentDay and ScheduledDay are excluded from modeling.

## Pre-Processing

Before modeling, our dataset was preprocessed as described below.

While there were no missing values, there were a few lines that exhibited values that did not make sense. One instance had an age value of -1, which was imputed with the median age of the dataset of 37. In addition, 5 instances had an AppointmentLag that was negative, indicating that the appointment was scheduled after it was over. These instances were imputed with 0.

The two identifiers, PatientID and AppointmentID were ignored along with AppointmentDay and ScheduledDay.

Gender and Neighbourhood were converted from categorical to indicator variable, and our numerical attributes (Age, Handicap, and AppointmentLag) were rescaled to 0-1.

# Exploratory Analysis

A basic exploratory analysis was performed on the dataset using Tableau. The no-show rate of the dataset is 20.2%. While not all plots created for exploratory analysis, some of the more interesting ones are plotted below.

A screenshot of text

Description generated with very high confidence The no-show rate by age tends to be higher than the population average for patients younger than 45. It should be noted that the variation in rate increases drastically in patients older than 80 years old. This is due to having less datapoints for these ages.

A close up of text on a white background

Description generated with high confidence

Patients who scheduled their appointment within a week tend to show more often than those who schedule further out. This is especially true for patients who have same day schedule (lag time = 0). This variable will likely be useful in models.

Patients with a chronic illness are more likely to show up for their appointment. This is true for those with hypertension (left) and diabetes (right).

A screenshot of a cell phone

Description generated with high confidenceA screenshot of a cell phone

Description generated with very high confidenceA screenshot of a cell phone

Description generated with high confidence

The scholarship variable indicates that those patients on a social welfare program are more likely to miss their appointment than those who are not. Essentially, this is an indicator for patients with limited financial resources. Other variables may be able to be introduced to further differentiate the socio-economic status of patients.

We hypothesized that patients who received text reminders would be more likely to show up to their appointment, however our initial analysis did not show this (left). Upon further exploration we found that patients who scheduled their appointment within 3 days did not receive text reminders. Once appointments with this lag time were excluded, the rate for those with reminders are lower than those without (right).

A close up of text on a white background

Description generated with high confidenceA screenshot of a cell phone

Description generated with high confidence

# Models

## Decision Tree

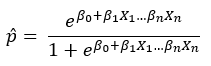
Decision trees provide some advantages over other modeling techniques. Simplicity and impactful features can become clear through a decision tree model. They also provide the potential insights into performing feature selection.

For this dataset, the most impactful features were: Appointment Lag, Age, and Neighborhood\_Santos.Dumont. These were the features the dataset was split in that provided the most information gain regarding the target variable of whether a patient would no-show or not.

The model that returned the highest AUC had the lowest complexity as noted below. When complexity was tuned to 0, the AUC increased from 0.5 to 0.65.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model Number | Seed | Min Split | Min Bucket | Max Depth | Complexity | AUC |
| 1 | 42 | 0 | 1 | 30 | 0 | 0.65 |
| 2 | 42 | 0 | 1 | 30 | 0.001 | 0.5 |
| 3 | 42 | 0 | 1 | 30 | 0.01 | 0.5 |
| 4 | 42 | 0 | 1 | 30 | 0.05 | 0.5 |
| 5 | 42 | 0 | 1 | 30 | 0.1 | 0.5 |
| 6 | 42 | 0 | 1 | 30 | 0.5 | 0.5 |

## Logistic Regression

Logistic regression in sensible in that it can output probabilities based off features built into the model. Logistic regression models are useful in that they predict values between 0 and 1 corresponding to the probability of the positive class occurring.

Our estimated regression equation would look like:

Where Beta values equate to the coefficients from the output. β0 equates to the coefficient of the intercept, and β1...βn correspond to the coefficients of the features built into the model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **Coefficients** | **Std. Error** | **z value** | **p value** |
| Intercept | -1.84413 | 1.13242 | -1.628 | 0.103423 |
| Age | -0.79665 | 0.05655 | -14.089 | < 2e-16 |
| Appointment\_Lag | 4.46413 | 0.11124 | 40.131 | < 2e-16 |
| SMS\_received | 0.35315 | 0.02033 | 17.372 | < 2e-16 |
| Scholarship | 0.15618 | 0.03036 | 5.144 | 0.000000269 |
| Hypertension | -0.12104 | 0.03061 | -3.954 | 0.000076709 |
| Alcoholism | 0.18105 | 0.05458 | 3.317 | 0.000909 |
| Diabetes | 0.12339 | 0.04147 | 2.976 | 0.002923 |

7 features in this model were statistically significant with p values of less than 0.05. The features are as followed: Age, Appointment Lag, SMS received, Scholarship, Hypertension, Alcoholism, and Diabetes.

This model resulted in an AUC of .66.

## SVM

Support-Vector Machines are models that have utility in scenarios where there is not an obvious boundary between the classes in a target variable. The objective is to maximize the margins and minimize hinge loss. By placing the support vectors as close as possible to each cluster of classes this objective is achieved.

When executing SVM models, a variety of issues arose. Increasing complexities and/or increasing degree (in the case of polynomial kernels) led to models that could not be executed. The solution here was to remove the neighborhood feature since it was categoric with 81 levels.

Other issues when running SVM models pertained to specific kernels. Linear and Laplacian kernels could not execute due to shortage of memory.

The output of the SVM model is outlined below (none of these models included the neighborhood variable).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Number | Seed | Kernel | Degree | Options | AUC |
| 1 | 42 | Polynomial | 1 | C = 0.01 | 0.54 |
| 2 | 42 | Polynomial | 1 | C = 0.1 | 0.56 |
| 3 | 42 | Polynomial | 1 | C = 0.001 | 0.54 |
| 4 | 42 | RBF |  | C = 5 | 0.49 |
| 5 | 42 | RBF |  | C = 10 | 0.51 |

## Random Forest

Random Forest models are useful in that they provide flexibility and improvement upon a simple decision tree. Simply stated, the more trees in the forest the more robust the model is. Random forest models take a random sample of selected data and creates decision trees from them. The best solution is then derived by a ‘voting’ mechanism from all the decision trees. These models also provide insights into which features are more impactful. The parameters that were used to the determine the best model are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Model Number | No. of Variables | No. of Trees | AUC |
| 1 | 4 | 400 | 0.6845 |
| 2 | 4 | 500 | 0.6877 |
| 3 | 4 | 600 | 0.6895 |
| 4 | 4 | 800 | 0.6913 |
| 5 | 6 | 400 | 0.7048 |
| 6 | 6 | 800 | 0.7071 |
| 7 | 8 | 600 | 0.7141 |
| 8 | 8 | 800 | 0.7141 |

Model number 7 and 8 gave the same performance but the 7th model was selected as the best model because it used relatively a smaller number of trees which reduces the computation complexity and time.

The variable importance table was used to select the variables with higher predictive powers. Based on this table, the variables that had higher predictive power are appointmentlag, age, sms-received, scholarship (financial assistance) and alcoholism.

|  |  |
| --- | --- |
| **Variable** | **Mean Decrease Accuracy** |
| AppointmentLag | 108.87 |
| Age | 55.09 |
| SMS\_received | 38.58 |
| Scholarship | 34.34 |
| Alcoholism | 31.39 |

## AdaBoost

Boosting methods have increased in popularity of late. Using an iterative ‘ensemble’ approach they can often provide better results than other modeling approaches. As the model goes through its iterations, it minimizes error. This is evident in our case, as denoted by the AUC value of the model. Up to this point, AdaBoost provides the highest AUC over the prior models. The parameters that were used to the determine the best model are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Model Number | Iterations | Complexity | AUC |
| 1 | 50 | 0.01 | 0.7284 |
| 2 | 50 | 0.1 | 0.7101 |
| 3 | 50 | 1 | 0.5 |
| 4 | 50 | 0.005 | 0.7304 |
| 5 | 50 | 0.0025 | 0.731 |
| 6 | 50 | 0 | 0.7083 |

Model 5 returns the best performance, with an AUC of 0.731.

The variables used in predicting are shown below and are ranked in decreasing order of importance:

* Alcoholism
* Diabetes
* Hypertension
* Age
* Handicap
* AppointmentLag
* Scholarship
* SMS\_received

## Gradient Boost

Gradient Boost models are another example of an ensemble modeling method. Gradient boost requires several components. It works by optimizing the loss function, leveraging a ‘weak learner’, and using an additive model to ultimately minimize the loss function. Decision trees are the ‘weak learner’ in this modeling method that provide informative splits that typically lead to better models.

|  |  |  |
| --- | --- | --- |
| Model Number | Max Depth | AUC |
| 1 | 6 | 0.7349 |
| 2 | 10 | 0.7313 |
| 3 | 4 | 0.7349 |
| 4 | 3 | 0.7333 |
| 5 | 20 | 0.7217 |
| 6 | 5 | 0.7349 |
| 7 | 7 | 0.7351 |

Model 7 returns the best performance, with an AUC of 0.7351

The variables used in predicting are shown below and are ranked in decreasing order of importance:

* AppointmentLag
* Age
* Gender
* SMS\_received
* Scholarship
* Hypertension
* Handicap
* Alcoholism
* Diabetes

## Neural Network

Neural networks are interesting models in that they try to replicate processes involved in the human mind. Artificial neuron models are based on neurons. These neurons take inputs which have associated weights which modify the strength of each input. The neuron then adds together all the inputs and calculates the output to be passed on.

The model that net the best AUC had 3 hidden layer nodes. Neurons are passed from layer to layer with varying inputs and weights. The activations in one layer then impact the activations in the layer that come after.

The hidden layers decompose the initial input. Pieces of the initial input are interpreted via the hidden layers to help build the neural network similar to how the human mind processes information and learns. The inputs from the varying layers then all combine to help provide an ideally accurate model depending on the inputs.

The output of the Neural Network model is outlined below:

|  |  |  |  |
| --- | --- | --- | --- |
| Model Number | Seed | Hidden Layer Nodes | AUC |
| 1 | 42 | 0 | 0.6609 |
| 2 | 42 | 1 | 0.666 |
| 3 | 42 | 2 | 0.6803 |
| 4 | 42 | 3 | 0.7209 |
| 5 | 42 | 4 | 0.7176 |
| 6 | 42 | 5 | 0.7122 |
| 7 | 42 | 6 | 0.71 |
| 8 | 42 | 7 | 0.714 |
| 9 | 42 | 8 | 0.7157 |

# Evaluation

## Model Comparison and Selection

The models were evaluated by using AUC performance metric because:

1. Our no-show appointments dataset is unbalanced

2. Model operating conditions are not known

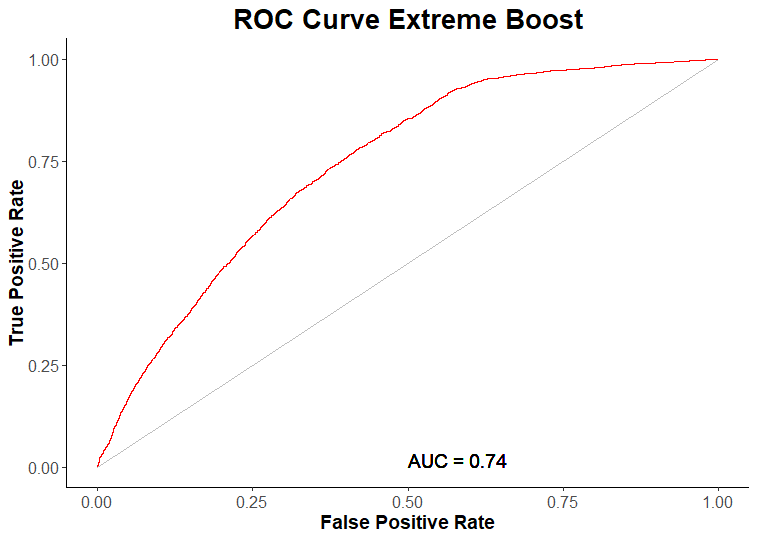
AUC is used when a single number is needed to summarize the model performance or when nothing is known about operating conditions.

Accuracy is the proportion of true results (both true positive and true negative) among the total population. It gives information about how well our model correctly classifies a positive as positive and negative as negative cases. Accuracy doesn’t provide information about incorrect classifications such as false positives and false negatives. Hence, we preferred to use AUC as a performance metric for our model evaluation.

Below is a summary table of the different classification models tested as well as the best AUC achieved for each model.

|  |  |
| --- | --- |
| Models | AUC |
| Decision Tree | 0.65 |
| Logistic Regression | 0.66 |
| Support Vector Machine | 0.54 |
| Random Forest | 0.71 |
| Adaptive Boost | 0.73 |
| Gradient Boost | 0.74 |
| Neural Network | 0.72 |

Certain modeling techniques (such as random forest) rely on simple averages in the ensemble. Boosting, essentially keeps adding new models to provide a more accurate estimate of the target variable. Our best selected Gradient Boost model (AUC 0.74) was evaluated against the Test (0.7422) and validation dataset (0.7351). A plot of the ROC curve of this model is below.



## Business Perspective

There is a clear need to reduce no shows in health care settings. The CRISP-DM methodology dictates that a thorough business understanding takes place. Followed by data understanding, data prep, modeling, evaluation, and deployment.

The evaluation of this prediction model would correspond to following healthcare or business situations.

1. Identify patients who are likely to no-show
2. Schedule only those appointments in which the patient will arrive

Situation 1 is where our model will provide the best benefit. If hospitals knew which patients needed the most intervention, they could reach out to those patients in an attempt to alter their no-show behavior.

Situation 2 is likely not feasible even with a model that could predict correctly every time. There are typically laws in place regarding what healthcare organizations can and cannot do. Even if not a law, refusing to put a patient on the schedule because a model predicted that they would not come to their appointment would cause great public dissatisfaction.

## Technical Improvements

The models are likely to be improved via cross-validation. This would ensure that all of the data is used for prediction and that subsets of the data are not heavily influencing the final model. Also, a cost matrix would be the best way to evaluate models. Because costs will differ throughout healthcare organizations (especially organizations within different countries), this was not a possibility for our model. It’s highly suggested that organizations doing similar analysis on their patient population should devote research on what different scenarios cost.

# Business Insights

## Important Factors

The Gradient Boost model was able to identify the important features in predicting the no-show appointments. Using these insights, we can make recommendations to decrease the number of no-show appointments.

* *SMS Received:* As evident from exploratory and modeling analysis, text reminders likely reduce the no-show rate. To further reduce the no-show rate, we recommend a need to automate SMS approach to ensure all patients receive text reminders. Perhaps requesting multiple cells numbers to text to add to likelihood patient shows up for appointments.
* *Appointment Lag-Time:* The predicted model observed that recently scheduled appointments have less likely no-shows. We suggest that appointments scheduled further than a week in advance be confirmed by the patient a week before the appointment. Patients who do not confirm may be prompted to reschedule and their appointment slot can be filled by other patients.
* *Age:* From the predicted model, it was evident that young people are more likely to miss appointments. Hence, educating the young people about the why not showing up to an appointment is a problem. It is quite possible that people may at least cancel in advance if they know that their absence is noticed and a concern.

## Model Implementation

Considering a what-if scenario to use our model results to double book appointments for likely no- show patients:

1. If our model predicts correctly, healthcare resources are not wasted
2. If the model predicts incorrectly, healthcare staff might not be able to attend all the scheduled patients which results in more stress on staff and dissatisfaction of patients.

Because our model has an AUC of .74, it does differentiate the two classes well, but not without overlap. It would not be ideal to implement the model in this way without further improvement.

## Model improvements

This model may be improved by the introduction of additional variables. Variables we recommend are as follows:

* *Patient Wait-Time*: while knowing the wait time of each appointment would cause a data leakage problem, data could be gathered on patient-specific historical wait-times. Research suggests that no-shows are at times attributed to dissatisfaction regarding time spent waiting. Having this variable would allow us to see if patients who no-show more frequently also have larger wait times historically. Given this information, we may be able to justify whether the risk of extending wait-times due to double booking would offset the benefit of not having unfilled appointment slots.
* *Patient Proximity:* the original dataset includes patient neighborhood and some neighborhoods did see variation in no-show rate when compared to others. Replacing this neighborhood information with the actual distance patients have to travel to their appointments may provide better insight as to whether distance or travel impacts no-shows.
* *Insurance Type:* we observed a difference in no-show rate between patients who are part of a government welfare program vs. those who are not. Adding in different insurance types may allow us to further differentiate the financial impact the appointment has across the patient population.
* *Appointment Visit Type:* The reason for the appointment may provide additional information to differentiate patients who show vs. no-show. For instance, perhaps patients are more likely to miss an annual appointment rather than an appointment that requires a procedure.

In addition to the proposed variables above, our data included less than 2 months’ worth of data. Gathering data that spans at least a year would allow us to use month as a variable. It’s possible that time of year impacts no-shows.