

# Using RandomForest to Predict Medical Appointment No-shows



Adeline Ong [Follow](#)

Nov 19, 2019 · 5 min read



Photo by Adhy Savala on Unsplash

No-shows, or patients who miss their scheduled appointments are common and costly to healthcare institutions. A US study found that up to 30% of patients miss their appointments, and \$150 billion is lost every year because of them.

Identifying potential no-shows can help healthcare institutions pursue targeted interventions (e.g. reminder phone calls, double-book an appointment slot) to reduce

no-shows and financial loss.

## Dataset

The Kaggle dataset comprised 110k appointments records from public healthcare institutions in a Brazilian city. The appointments occurred across a 6-week period in 2016.

Here's a summary of the dataset's features:

| Target                      | Identifiers                | Patient Information                   | Appointment Information                                    | Location of Clinic               |
|-----------------------------|----------------------------|---------------------------------------|--|----------------------------------|
| No show<br>(Binary: Yes/No) | Patient ID<br>(Non-unique) | Age<br>(Range: -1 to 115, continuous) | Schedule Day & Time<br>(Format: YYYY-MM-DD<br>TZHH-MM-SSZ) | Neighbourhood<br>(81 categories) |
|                             | Appointment ID (Unique)    | Gender<br>(Binary: Male, Female)      | Appointment Day<br>(Format: YYYY-MM-DD<br>TZHH-MM-SSZ)     |                                  |
|                             |                            | Hypertension (Binary: 1,0)            |  |                                  |
|                             |                            | Diabetes (Binary: 1,0)                |  |                                  |
|                             |                            | Alcoholism (Binary: 1,0)              |  |                                  |
|                             |                            | Handicap (Range: 0 to 4)              |  |                                  |
|                             |                            | Scholarship (Binary: 1,0)             |  |                                  |
|                             |                            | SMS received (Binary: 1,0)            |  |                                  |

Table summarising the dataset's original variables

Patient IDs were non-unique, which suggests that the same patient had multiple appointments during the 6-week period. To avoid data leakage (which would occur if the same patient's data was used for validation and testing), we will only include a patient's latest appointment in our models.

## Data Cleaning

There were 2 main cleaning steps:

1. Binary coding (1,0) no-shows, gender and handicap. I assumed that handicap was suppose to be binary because of its description on Kaggle.
2. Dropping observations that had logical inconsistencies, such as negative ages, and when scheduled dates were after appointment dates.

# Feature Engineering

Feature engineering was used to recode datetime features and retain information captured from previous appointments (those prior to the latest).



List of original dataset features and engineered features. Colours indicate the features from which the engineered features were created from.

Prior no-shows and prior appointments were calculated by subtracting 1 from each patient's total number of no-shows and appointments respectively.

Scheduled date and appointment date were split into day of week (DoW) and day of month. Scheduled time was also split into hour of the day.


Day difference was the difference in terms of days between the scheduled date and appointment date.

Total conditions was the sum of hypertension, diabetes, handicap and alcoholism. In other words, it indicates the number of conditions a patient suffers from.

## Feature Selection

Features were selected based on their Information Value (IV), which ranks and scores features based on how well they predict the target.

| SELECTED FEATURES |          |
|-------------------|----------|
|                   | IV Score |
| Day difference    | .53      |
| Prior no shows    | .38      |
| SMS               | .10      |
| Age               | .07      |
| Scheduled hour    | .03      |
| Hypertension      | .02      |
| Scheduled day     | .02      |

 EXCLUDED  
ID features  
Redundant features  
Neighbourhood  
Low IV score (<.02)

Selected features and their Information Value (IV) scores.

Features with IV scores  $<0.02$  (the threshold for very poor predictors) were dropped. Using a conservative threshold helps ensure that useful features are not prematurely excluded from the model.

As it turns out, most features were very poor predictors of the target.

In addition, ID features, redundant features and neighbourhood were removed. Neighbourhood was removed because I was unable to find additional information about them, and including 80 dummy variables in the model was undesirable.

About 62k observations remained after data cleaning and feature selection. 20% of these were no-shows.

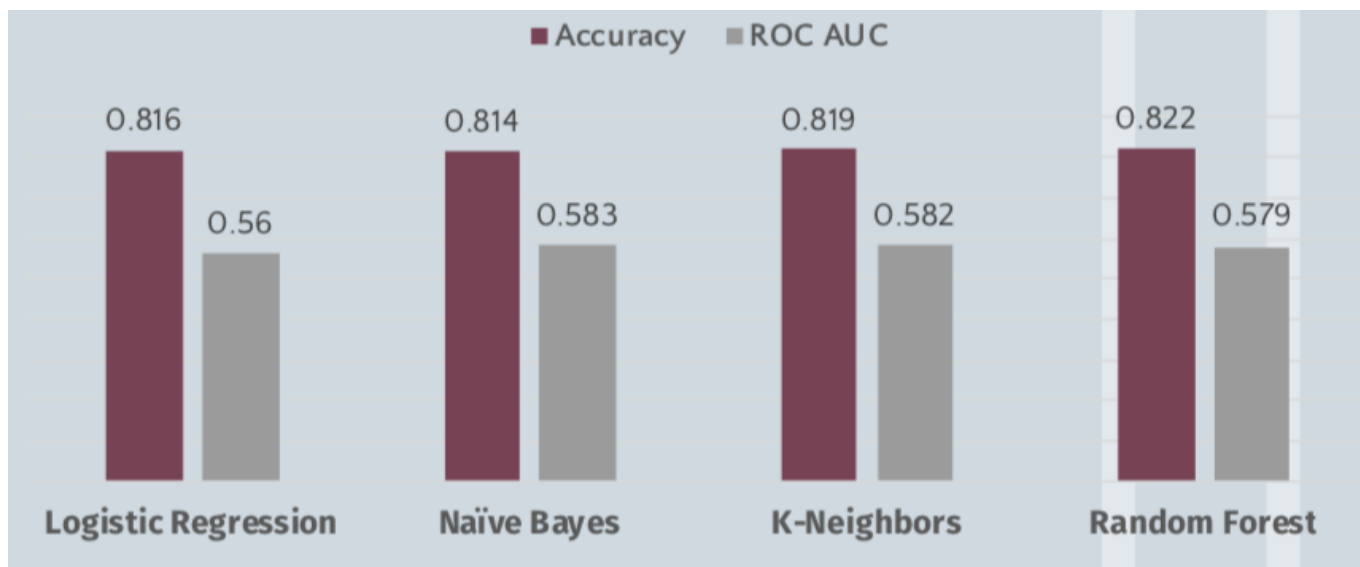
## 5-Fold Cross-Validation

The dataset was split into 20% test and 80% cross-validation data sets.

Using 5 training-validation folds, the data was fitted on four supervised learning algorithms:

1. Logistic Regression
2. Naïve Bayes
3. K-Nearest Neighbour (KNN)
4. Random Forest

RandomCVSearch (5 folds, 10 iterations, 50 fits) was used to tune the hyperparameters for KNN and Random Forest.



The models were evaluated based on their mean accuracy and Receiver Operating Characteristic Area Under the Curve (ROC AUC) scores.

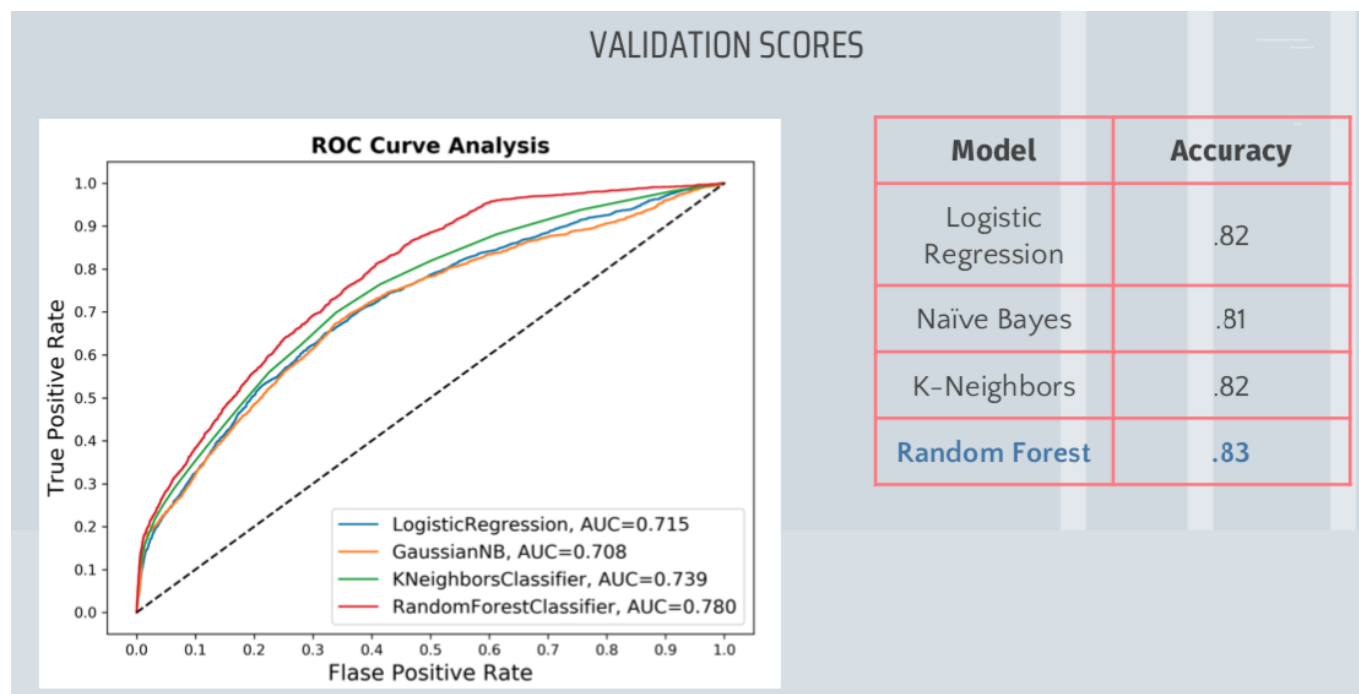
Unfortunately, the models' performance were similar, and I was unable to choose a best model.

*Note: I did not use F1 score because it was dependent upon the chosen classification threshold, which I intended to tune after model selection. I used ROC AUC score and accuracy because I wanted the best model regardless of threshold.*

## Validation Using More Data.

As cross-validation was not useful for model selection, the 80% used for training and validation was re-split into 60% training and 20% validation. (Previously cross-validation was using 64% for training, and 16% for validation for each fold). Hopefully

having a larger validation set would make the differences among the models more obvious.



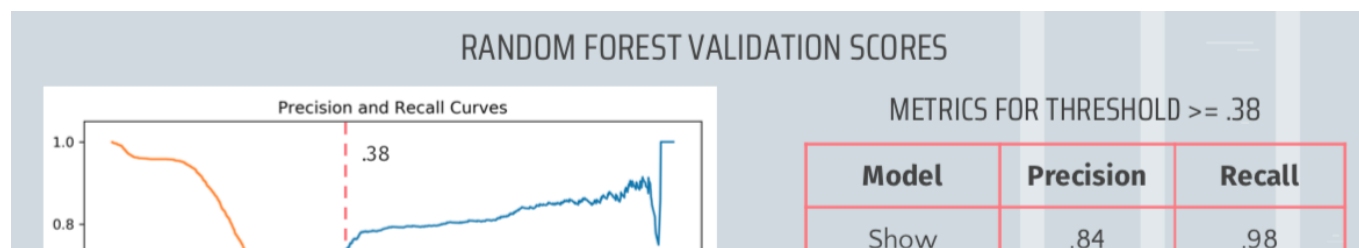
Graph depicting models' ROC curves, and table of models' accuracy scores

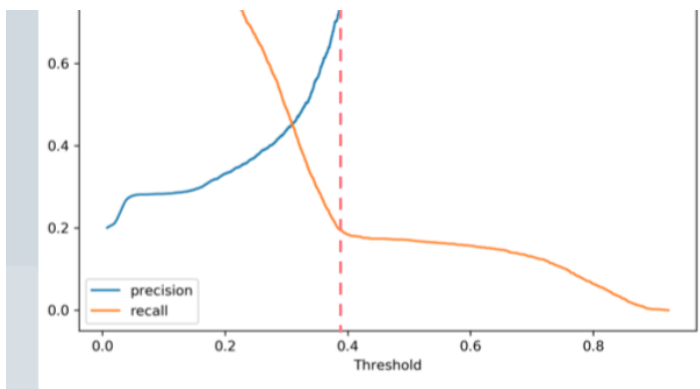
From the ROC Curve Analysis, we can see that RandomForest outperformed the other algorithms. Its accuracy was also comparable to the others. We'll use RandomForest as our model going forward.

## Tuning Classification Thresholds

The classification thresholds for RandomForest were tuned based on the assumption that healthcare institutions would want to test out interventions before rolling them out on a mass scale. Consequently, precision was prioritised over recall.

I chose a threshold of 0.38, because it was about when the two curves tapered off. This threshold meant that no-shows would be correctly identified 74% of the time (precision), and about a quarter of all no-shows would be identified (recall).





|         |     |     |
|---------|-----|-----|
| No show | .74 | .25 |
|---------|-----|-----|

9

## Testing

The model was re-trained using 80% of the data and tested on the remaining 20%.

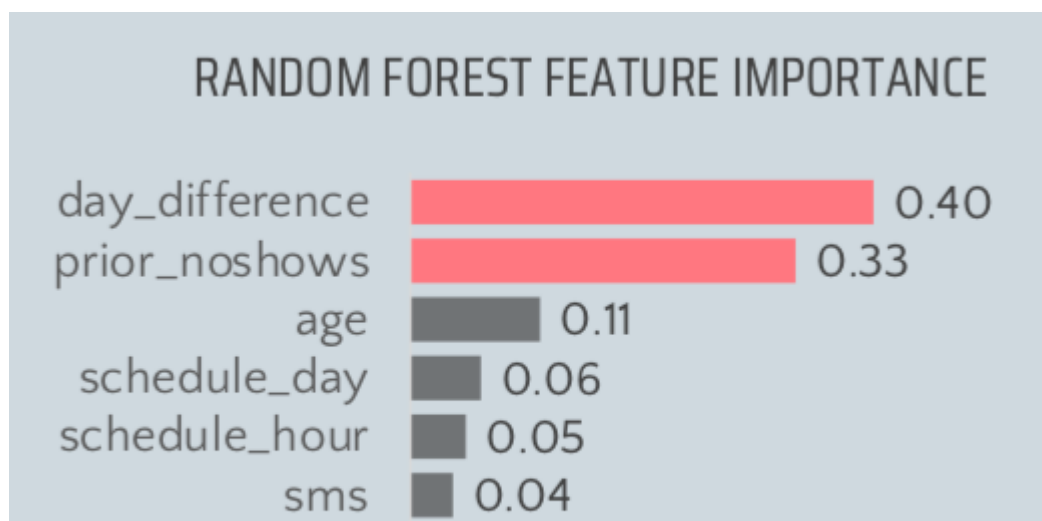
Test scores were similar to validation scores, which suggests that the model generalises well.

TEST SCORES FOR THRESHOLD  $\geq .38$

| Model   | Precision | Recall   | Accuracy |
|---------|-----------|----------|----------|
| Show    | .83 ▼.01  | .97 ▼.01 | .82      |
| No show | .68 ▼.06  | .21 ▼.04 |          |

▼ Points below validation scores

## A Closer Look at RandomForest





hypertension | 0.01

RandomForest's feature importance shows that day difference and prior no-shows were important in predicting no-shows.



Histograms depicting the differences in distributions for shows and no-shows, for prior no shows and day difference.

In particular, no-shows tended to have a larger difference between their scheduled and appointment days, and a history of not showing up.

Knowledge of feature importance is useful for choosing intervention measures. For instance, patients who scheduled their appointments early likely forgot about them. A



possible intervention would be to have more personal patient touch points (e.g. phone calls, since SMSes do not seem to be effective).

On the other hand, the reason might be more habitual for patients with a history of no-shows. If institutions are unable to convince these patients to show up, double-booking their slot could be a possible solution.

## Conclusion

In conclusion, it seems that no-shows can be predicted from patient information and appointment data. More information about the clinic's location (e.g. transport accessibility), type of care sought (e.g. primary, specialist), and the patient (e.g. education, income) would likely improve the model. The model could also benefit from a cost-benefit analysis of possible intervention measures to achieve a balance of precision and recall that would make the most business sense.

*\*\*\*Check out the codes on my GitHub*

[Machine Learning](#)[Data Science](#)[Brazil](#)[Health](#)[About](#) [Help](#) [Legal](#)

Get the Medium app

