2018

Medical Appointment No-Shows in Brazil

PREDICTIVE ANALYSIS USING SPARK'S MLLIB

BY CRAIG BARBISAN

UNIVERSITY OF TORONTO – SCHOOL OF CONTINUING STUDIES | SCS-3252

Contents

Introduction

"If I have made an appointment with you, I owe you punctuality, I have no right to throw away your time, if I do my own."

- *Richard Cecil, poet*

As populations grow, so will the need for the resources required to support and care for the population. Among the more critical resources necessary to help reach life expectancy and beyond, is access to health care resources, particularly in times of illness. While health care is currently available to many populations throughout the world, the value ascribed to this resource does not always align with its relative importance. As the data presented in this report will show, medical appointments are not always kept by patients. This type of behaviour not only can put the patient's health at risk, but it can create artificial backlogs in the operations of medical clinics, limiting other patient's access to healthcare as a downstream effect.

But why do patients miss appointments, given the potential consequences posed to themselves and others if they do? Appointments are missed for all sorts of reasons. Appointments may be forgotten by the patient or made suddenly too inconvenient to keep. They may be cancelled because the patient has anxiety about the appointment and decides it is too frightening a proposition. Perhaps the patient is so sick, that travelling to the appointment becomes out of the question. Or perhaps the patient attempted to keep the appointment, but it just wasn't possible to get there due to weather, transportation issues or conflicting priorities. And none of those scenarios even consider cases where the missed appointment was not the fault of the patient at all! Maybe the appointment was recorded incorrectly by the clinic staff when it was scheduled. Maybe the clinic had to reschedule the appointment but forgot to notify the patient. Or maybe a cancellation was requested by the patient or the clinic and that wasn't captured correctly either. These are all just some examples of the circumstances that could lead to a missed appointment, but the net result is the same. Time is wasted, doctors are inconvenienced, and patients may not receive advice or treatment that could be critically time-sensitive.

It would be desirable to find solutions that would minimize the likelihood of medical appointments being missed. However, there are many variables described in these scenarios that are not necessarily subject to control. There isn't anything a medical clinic can do to control the traffic or the weather. If the patient is going to miss the appointment due to anxiety or simply because he's being inconsiderate, the clinic won't know that until the appointment has already been missed. But if the clinic was given the ability to predict that the patient won't show up, then measures could be taken to reduce the disruption to the clinic; and to reschedule the appointment while ensuring that the impact to patient outcomes is minimized. Regardless of who is at fault for the missed appointment, mutual benefits could be achieved by knowing in advance that the missed appointment will likely occur.

There is more to this investigation than just discussing ways to ensure patients keep their appointments, or to suggest ways for clinics to better handle no-show scenarios. Data is required to better understand the factors which influence the likelihood of a missed medical appointment, as are tools to process and analyze that data. The data which supported this analysis was sourced from a website called Kaggle (http://www.kaggle.com). Kaggle is an online platform which allows its users to post their data sets and share their data with other users, which allows them to collaborate on problems and derive insights from the data. The particular data set selected was titled "Medical Appointment No Shows" (Hoppen, 2017) and contained data for approximately 110,000 medical appointments scheduled to occur in medical clinics across Brazil between April 2016 and June 2016. The limited geography and timeframe represented in this dataset, may bias the results and limit its usefulness outside of that context. For example, if weather were a significant factor to our model, or proximity of the appointment date to a holiday, then the results may be different from country to country. The weather in Canada is obviously not going to be similar to the weather in Brazil. Similarly, Canada and Brazil do not observe all of the same holidays, so while there are 2 holidays from the end April to the beginning of June in Brazil, there is only 1 holiday during that time period in Canada.

Data on its own is not usually sufficient to perform any sort of formal analysis, particularly if the dataset qualifies as "big data". Even in our moderately sized set of 110,000 records, it is still far too large to consume and derive meaning from through manual inspection. Tools are required for tasks such as inspection, cleaning, filtering, grouping, aggregating and reporting. For this analysis, there were several tools selected to help complete the job:

- **Databricks Community Edition** – a cloud-based big data platform which provides an AWS micro-cluster, a cluster manager and a notebook environment.
- **Apache Spark** – a cluster computing framework upon which libraries are built including SQL and Dataframes, MLlib, GraphX and Spark Streaming.
- **Scala programming language** – a general purpose programming language which is both object-oriented and functional, and uses a Java like syntax.
- **Spark MLlib** – A collection of algorithms and utilities used to build machine learning models; available for several programming languages including Scala.
- **Spark ML Pipeline API** – a Spark MLlib utility built on top of DataFrames, to help create and tune practical ML pipelines.

Objectives

Ultimately, this was a learning exercise to demonstrate the skills that were acquired throughout the “Big Data Management Systems and Tools” course. But there are real-world applications of this analysis which were touched on briefly in the introduction and will be discussed in further detail later in the report. Even though the veracity of the dataset provided by Kaggle cannot be confirmed, the exercise is still worthwhile because the machine learning pipeline being developed could easily be applied to a new dataset with the same features. If the new dataset were confirmed to be valid, then the pipeline would produce a usable model. The following is a brief discussion of the 3 objectives of this investigation:

1. To leverage available big data tools which facilitate analysis regardless of the size of the data set – The data set being analyzed would not reasonably full under the category of “big data”. With just over 110,000 records, it could be easily opened in Excel, SAS, MicroStrategy, or any other tool suited to moderately sized data sets. But the analytical tools selected for this investigation, would be suitable for much larger data sets. The tools selected were discussed in the introduction section and will not be repeated here.

2. To enhance the data set in ways that further our understanding of the phenomenon being observed – When analyzing data, it is sometimes necessary to “read between the lines”. We shouldn’t just analyze the features that are included in the data set we’re working with. It is possible to combine or transform features, to gain new insights into the problem that weren’t explicitly represented in the data. Similarly, it behooves us to filter out or ignore features that don’t provide any value to the analysis, or which can be demonstrated to be invalid. These processes are commonly referred to as feature extraction and feature selection respectively and are an important step in the workflow for supervised learning.

3. To leverage machine learning algorithms in the hopes of being able to predict similar behaviour in the future – Regression and classification algorithms can be trained with data to create predictive models of varying accuracy. This analysis will use both binary logistic regression and random forest classification algorithms and compare their results to select the most accurate model that can be created by our machine learning pipeline.

Dataset Features

In machine learning, a feature is an individual measurable property or characteristic of the phenomenon being observed. (Wikipedia - The Free Encyclopedia, 2018). In the case of this investigation, the phenomenon being observed is patients missing their medical appointments. Each appointment is represented as a new record in the dataset. We refer to the set of features explicitly contained within the dataset as the "raw features". Not all raw features necessarily add value to our analysis and those that don't are referred to as extraneous. Raw features can be combined and transformed to provide additional insights about the phenomenon being described, and we refer to those as engineered features or extracted features. The raw features provided in the dataset are as follows:

Raw Feature	Description
AppointmentID	A numeric identifier which uniquely identifies each appointment. This column was deemed extraneous and excluded from the model.
PatientID	A numeric identifier which uniquely identifies each patient. This column was deemed extraneous and excluded from the model.
Gender	Contains either M/F to indicate that the patient is male or female respectively.
ScheduledDay	The day that the patient registered the appointment. This field contains both the date and time.
AppointmentDay	The day of the actual appointment. This field contains the date only and logically should fall on or after the ScheduledDay.
Age	An integer field indicating the age, in years, of the patient.
Neighbourhood	The geographic neighbourhood in which the medical clinic is found.
Scholarship	A boolean field (1/0) which indicates whether the patient receives the Bolsa FamÍlia social welfare or not.
Hypertension	A boolean field (1/0) which indicates whether the patient suffers from hypertension or not.
Diabetes	A boolean field (1/0) which indicates whether the patient suffers from diabetes or not.
Alcoholism	A boolean field (1/0) which indicates whether the patient is alcoholic or not.
Handicap	A boolean field (1/0) which indicates whether the patient has a physical handicap or not.

SMS_received	A boolean field (1/0) which indicates whether the patient received SMS text message reminders about the appointment.
No-show	A boolean field (Yes/No) which indicates whether the patient missed the appointment or not.

There are many reasons why in our analysis, we would want to extract features from the data, over and above what exists in our initial dataset. Machine Learning algorithms expect their inputs to be numeric, however not all of the useful information in our dataset is numeric. For example, gender is represented as M or F for male or female respectively. These characters would be invalid inputs to our algorithm, so it is necessary to transform those characters to some numeric representation. It's also possible that certain features on their own aren't useful to the analysis, but together they provide a useful piece of information that can be extracted. ScheduledDay and Appointment day are date values and not suitable inputs to an ML algorithm. But by calculating the number of days between them, we arrive at a numeric feature that absolutely is worth considering in the context of our analysis. The longer it has been since the patient has scheduled the appointment, the more likely he/she is to have forgotten it! Here is a list of extracted features for our analysis along with the rationale for including them:

Extracted Feature	Description
GenderIndex	A numeric representation of the Gender raw feature, which is a categorical value. A patient who is male may be more or less likely to miss an appointment than a female patient or vice-versa.
AgeBucket	Although Age is already a numeric feature, it made sense to "bucket" values together, which effectively transforms it to a categorical value. The medical needs may not differ widely between a 12 year-old and a 13 year-old, but likely differ greatly between a 12 year-old and a 16 year-old. The analysis tries to account for this by bucketing the age category into age ranges.
Neighbourhood Index	A numeric representation of the Neighbourhood raw feature, which is a categorical value. Clinics in certain neighbourhoods may be more or less likely to miss an appointment, depending on the population that they serve, accessibility of the clinic, etc.
IssueCounter	A numeric feature which represents the total number of pre-existing medical concerns the patient is known to have, selected from the following list: hypertension, diabetes, alcoholism, physical handicap. The more medical issues a patient suffers from, the

	harder it may be to keep an appointment. Or perhaps the more likely they are to attend, given their need for medical attention.
DaysBetween	A numeric feature which represents the number of days between when the appointment was scheduled and when the appointment actually occurs.
ScheduledHour	A numeric feature which represents the hour of the day in which the appointment was recorded in the schedule (not when it is scheduled to occur). Appointments booked early in the day, or late in the day, may be more likely to be scheduled incorrectly given fatigue of the employee. Incorrectly scheduled appointments are likely to be missed.

Future Considerations

Features were considered as part of the analysis, that could not be included due to either time constraints, or limitations in the data that was available for analysis. It is possible that the introduction of these features could have resulted in a more accurate model, so they will be proposed now as potential future enhancements:

Weather

Weather can greatly influence a patient’s desire, or ability, to keep an appointment. By including features that describe weather conditions local to the neighbourhood of the medical clinic, weather can be factored into the model.

Weekends/Holidays

A patient may be less likely, or willing, to attend an appointment that falls on, or adjacent to, a weekend or holiday. These are periods of time when social plans can suddenly materialize and make travelling to or waiting for appointments seem much less palatable. A list of public holidays in Brazil in 2016 would be a pre-requisite to extract these types of features. No additional data would be necessary to support a similar analysis relative to weekends.

Booking Agent

Knowing who booked the appointment on behalf of the patient would be useful in the context of this analysis. A name wouldn’t necessarily be required. Any arbitrary unique identifier would be sufficient for the analysis while maintaining privacy of clinic staff. Assuming that not all staff members perform at the same level, then a weaker staff member may be more likely to errantly register an appointment and that would likely impact the model in a meaningful way.

Analysis

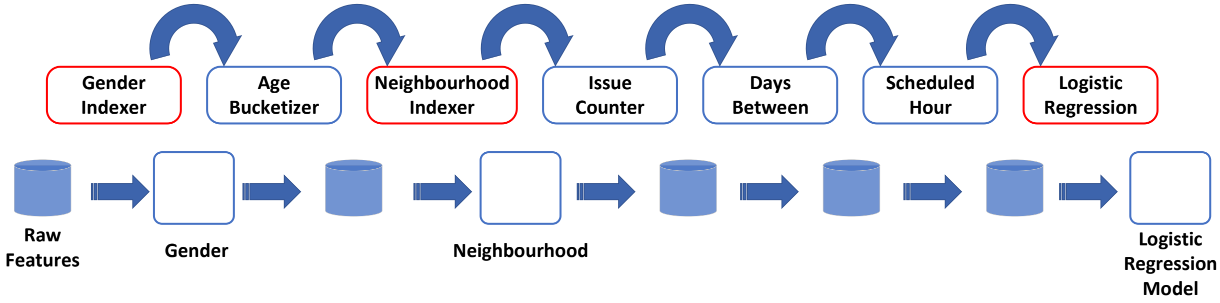


Figure 1 - Machine Learning Pipeline for this analysis. Note that the same pipeline was used for the random forest classification algorithm. Just the final estimator and its result would have been different.

Our stated objectives for this investigation included enhancement of the dataset to gain further insight into no-show appointments and the use of machine learning algorithms to predict when that scenario might occur. By constructing a machine learning pipeline using the Spark Pipeline API, we achieve both objectives. A pipeline is made up of several stages which combine to transform our raw features into an enhanced version of the initial dataset and train a machine learning model to predict the phenomenon being observed. Figure 1 depicts the pipeline that was developed to build the model for this investigation. Each stage of a pipeline is either a transformer (depicted as a blue box) or an estimator (depicted as a red box). Transformers take a dataset as input and they output a modified version of the same dataset. The models returned by an ML algorithm are transformers, because they take a dataset as input and return the same set with a prediction feature attached. Estimators take a dataset as input and return a transformer. Machine learning algorithms are estimators, but there are other estimators which can be included in the pipeline as depicted in Figure 1 (Apache Software Foundation, 2018).

Our model is intended to make a binary prediction, which represents whether a patient will miss their appointment or not. Logistic regression is a suitable algorithm for this purpose because it predicts the probability of an instance belonging to a default class, which can be snapped into a 0 or 1 classification (Jason Brownlee, 2016). The regression model assigns a probability to one of the classes, and we can round that probability to 1 or 0 accordingly, 1 meaning the appointment was missed and 0 meaning the appointment was kept.

For the sake of comparison, a second machine learning algorithm was also leveraged. The dataset and the pipeline remained the same, but the algorithm used was the Random Forest Classification algorithm. This algorithm is useful because it helps to avoid the problem of overfitting the model and it knows how to handle categorical values, of which we have several in our dataset (Synced, 2017).

Findings

The first algorithm applied to our machine learning pipeline was the logistic regression algorithm, which counter-intuitively is a classification algorithm. After tuning the associated hyper-parameters and using cross-validation, the best model was selected. The parameters for this model can be found in Appendix I. When we apply this model to the test set of data and compare the predicted value for NoShow to the actual value for NoShow, the model is correct 79.46% of the time.

The second algorithm applied to our machine learning pipeline was the random forest classification algorithm. To apply this model, it was not necessary to re-start the entire analysis. Rather, the first pipeline was cloned, with the last stage being updated to a different estimator. A new parameter grid was built for hyper-parameter tuning and cross-validation was used again. In this case, when we applied the cross-validated model to the test set of data and compared the predicted value for NoShow to the actual value for NoShow, the model was correct 79.74% of the time. So, in this case, the model built on the random forest algorithm was only slightly better than the logistic regression algorithm. Although, being above the 79.5% mark, makes the author feel better about claiming to his friends, classmates and colleagues that the model is 80% accurate…

Now that there is evidence that the predictive model is reasonably accurate, it is worthwhile to consider how the model can or should be applied. The model should serve some purpose, otherwise the entire exercise becomes purely academic. There are multiple different applications for this model, including the following:

Resource Planning – Assuming that the medical clinic retains all of the inputs required by the model, then it could be executed for each patient in advance of their next appointment. That would give the clinic a view each day of potential gaps in the schedule and staff levels could be adjusted accordingly. If 40 appointments constitute a full day's schedule, but 8 no-shows are predicted, then staffing levels could be cut for the day by up to 20%.

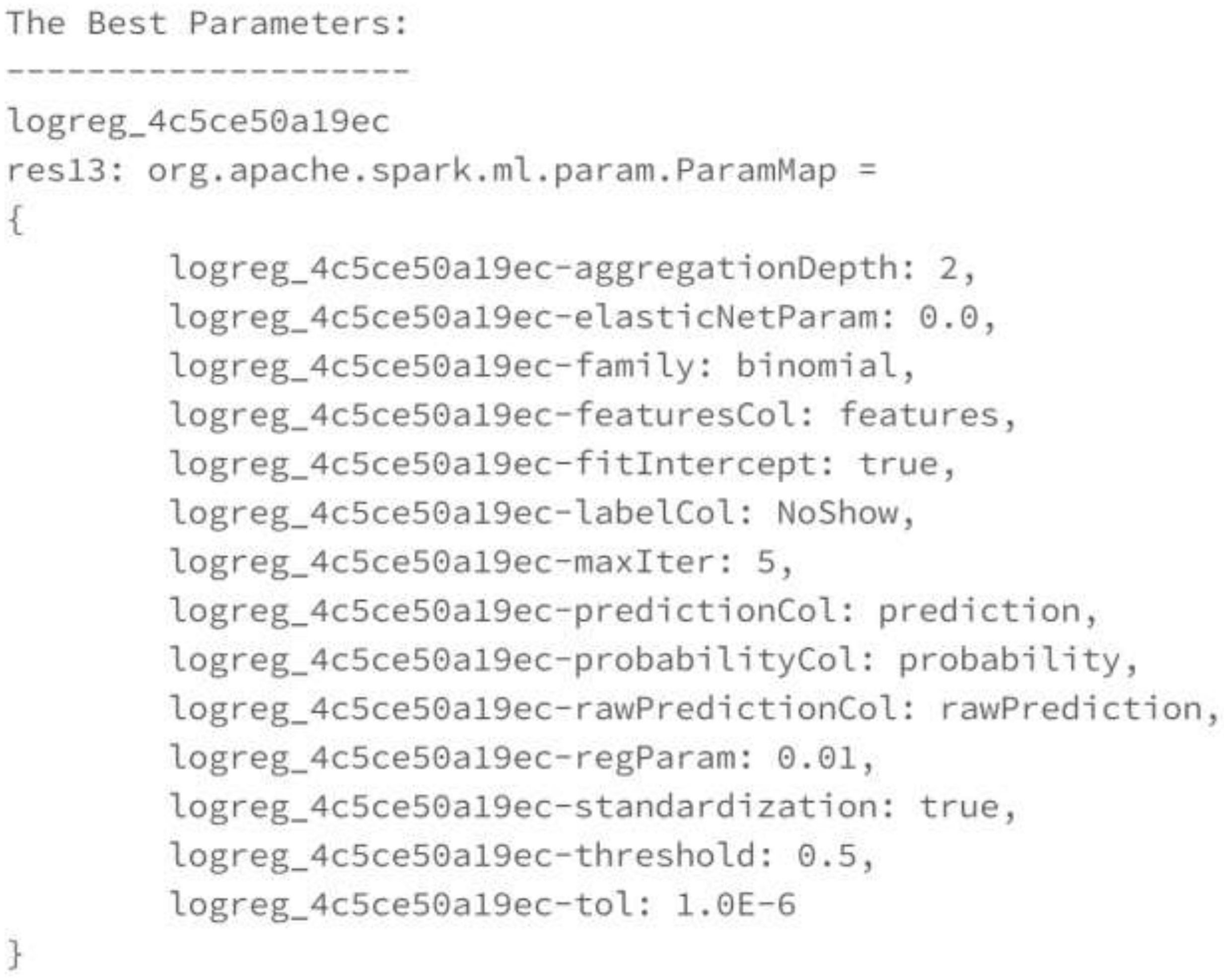
Waitlist Management – Another benefit of being able to predict no-show appointments is that it can help to reduce wait-times for the patients that do attend their appointments. Using the same example as above, if 20% of patients are expected to miss their appointment tomorrow, then up to 20% additional patients could be offered an earlier appointment. This eliminates the need to reduce staffing levels and allows other patients to be seen more expediently.

Patient Outreach – As described earlier in this report, there are many reasons why it is not desirable for a patient to miss their appointment in the first place. By giving the clinic advanced notice of a potential no-show, it provides an opportunity to pro-actively contact the patient and remediate the situation. If the no-show were to be related to a scheduling issue, then that could be easily resolved. If the patient were too sick to attend or had no transportation, then other accommodations could be made. If the patient had simply forgotten, then a reminder could be provided. By knowing that the appointment is at risk, the clinic has the opportunity to take action to prevent the no-show from occurring.

Appendices

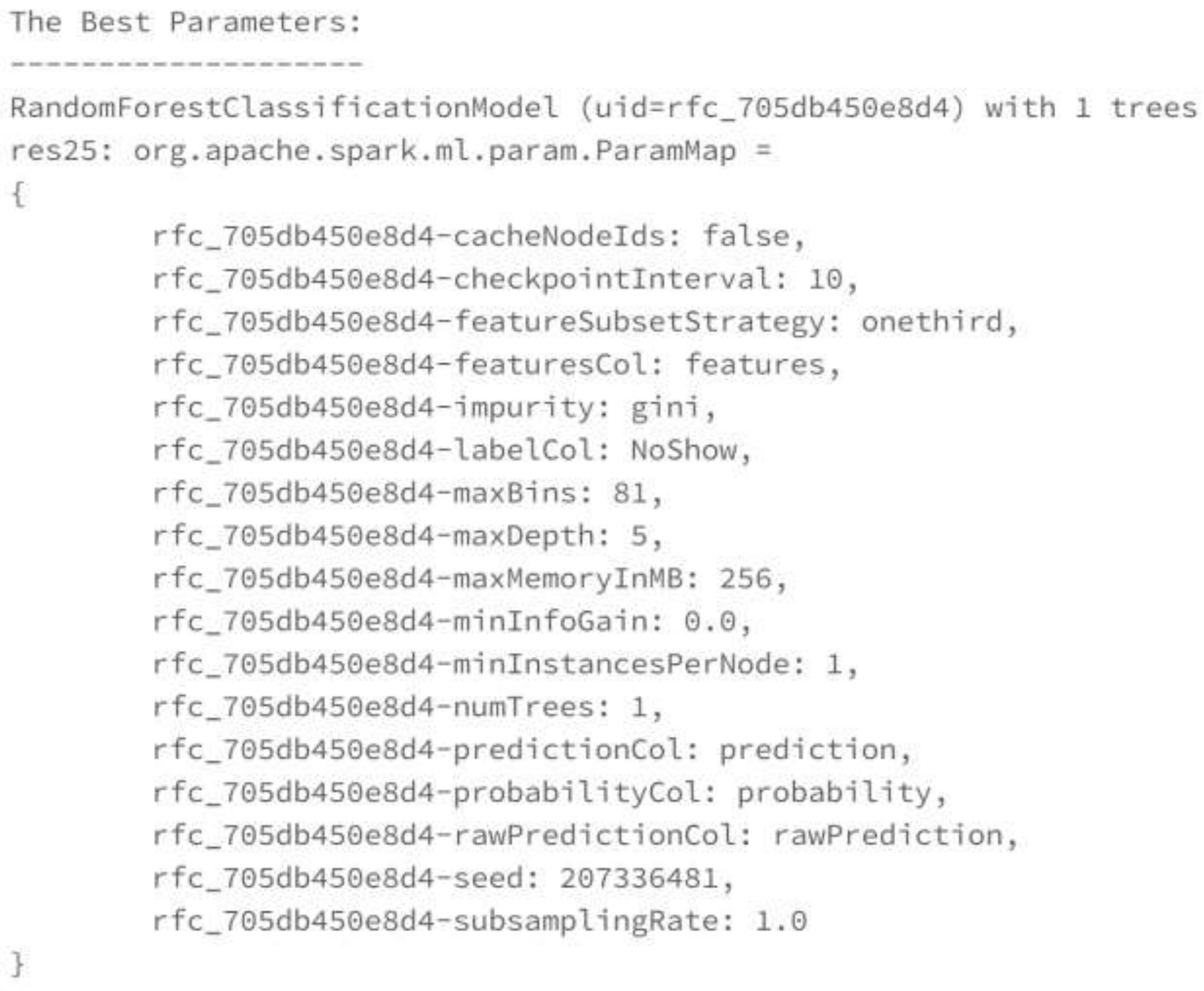
Appendix I - Model Selection

Logistic Regression



```
The Best Parameters:
--------------------
logreg_4c5ce50a19ec
res13: org.apache.spark.ml.param.ParamMap =
{
        logreg_4c5ce50a19ec-aggregationDepth: 2,
        logreg_4c5ce50a19ec-elasticNetParam: 0.0,
        logreg_4c5ce50a19ec-family: binomial,
        logreg_4c5ce50a19ec-featuresCol: features,
        logreg_4c5ce50a19ec-fitIntercept: true,
        logreg_4c5ce50a19ec-labelCol: NoShow,
        logreg_4c5ce50a19ec-maxIter: 5,
        logreg_4c5ce50a19ec-predictionCol: prediction,
        logreg_4c5ce50a19ec-probabilityCol: probability,
        logreg_4c5ce50a19ec-rawPredictionCol: rawPrediction,
        logreg_4c5ce50a19ec-regParam: 0.01,
        logreg_4c5ce50a19ec-standardization: true,
        logreg_4c5ce50a19ec-threshold: 0.5,
        logreg_4c5ce50a19ec-tol: 1.0E-6
}
```

Random Forest Classification



```
The Best Parameters:
--------------------
RandomForestClassificationModel (uid=rfc_705db450e8d4) with 1 trees
res25: org.apache.spark.ml.param.ParamMap =
{
        rfc_705db450e8d4-cacheNodeIds: false,
        rfc_705db450e8d4-checkpointInterval: 10,
        rfc_705db450e8d4-featureSubsetStrategy: onethird,
        rfc_705db450e8d4-featuresCol: features,
        rfc_705db450e8d4-impurity: gini,
        rfc_705db450e8d4-labelCol: NoShow,
        rfc_705db450e8d4-maxBins: 81,
        rfc_705db450e8d4-maxDepth: 5,
        rfc_705db450e8d4-maxMemoryInMB: 256,
        rfc_705db450e8d4-minInfoGain: 0.0,
        rfc_705db450e8d4-minInstancesPerNode: 1,
        rfc_705db450e8d4-numTrees: 1,
        rfc_705db450e8d4-predictionCol: prediction,
        rfc_705db450e8d4-probabilityCol: probability,
        rfc_705db450e8d4-rawPredictionCol: rawPrediction,
        rfc_705db450e8d4-seed: 207336481,
        rfc_705db450e8d4-subsamplingRate: 1.0
}
```

Appendix II – Databricks Notebook

https://goo.gl/XYdcGp

Appendix III - References

Apache Software Foundation. (2018). *ML Pipelines*. Retrieved from Apache Spark: https://spark.apache.org/docs/latest/ml-pipeline.html

Hoppen, J. (2017). *Medical Appointment No Shows*. Retrieved from Kaggle: https://www.kaggle.com/joniarroba/noshowappointments

Jason Brownlee, P. (2016, April 1). *Logistic Regression for Machine Learning*. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/logistic-regression-for-machine-learning/

Synced. (2017, October 24). *How Random Forest Algorithm Works in Machine Learning*. Retrieved from Synced Review: https://syncedreview.com/2017/10/24/how-random-forest-algorithm-works-in-machine-learning/

Wikipedia - The Free Encyclopedia. (2018, March 28). *Feature (machine learning)*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Feature_(machine_learning)