Medical Appointment No Shows Prediction

https://github.com/f18bc/data1030project

Bochen Fu

December 5, 2022

1 Introduction

For health care facilities, receiving a patient's appointment request, then completing the appointment process, but the patient not showing up can be a frustrating issue and a potential cost. Therefore, it would be beneficial if facilities can predict whether the patient will show up or not in advance.

This report analyzes a dataset of medical appointment no-shows in Vitoria, capital of the state of Espírito Santo, Brazil. In 2013, Vitoria's government started collecting scheduling data in all municipal health care facilities, which includes medical appointment no-shows. The data use unique patient ids to ensure anonymity. This report uses its second version on Kaggle, with the help of Joni Hoppen and Aquarela Advanced Analytics under a CC BY-NC-SA 4.0 license, which contains 110,527 data in 2016.

Each row corresponds to a patient's medical appointment, and the attributes are PatientId, AppointmentID, Gender, ScheduledDay, AppointmentDay, Age, Neighbourhood, Scholarship, Hipertension, Diabetes, Alcoholism, Handcap, SMS_received, and No-show. There are two typos regarding the variables name: Hipertension corresponds to hypertension, and Handcap corresponds to handicap, which is the level of handicap. This analysis will fix those typos. Moreover, Scholarship corresponds to whether a patient was receiving benefits of the 'Bolsa Família' social welfare program from the Brazilian Government. Also, ScheduledDay corresponds to the date that a patient makes an appointment.

In our analysis, we will focus on medical no-shows, and use No-show as the target variable. Here, the goal is to predict if a patient will show up for an appointment, and it is a classification problem.

The dataset does not contain missing data. For individual variables, PatientId and AppointmentID are unique ids for patients and appointments for privacy. Male, Scholarship, Hipertension, Diabetes, Alcoholism, Handcap, SMS_received, and No-show are all categorical attributes, which represent if a patient belongs to one group or another. Neighbourhood indicates where an appointment took place in Vitoria. Age is a contiguous variable, ranging from -1 to 115. ScheduledDay and AppointmentDay represent the date and time when the scheduling and appointment happened. For privacy reasons, the times of AppointmentDay were all set to 00:00. ScheduledDay ranges from November 2015 to May 2016, and AppointmentDay ranges from April 2016 to May 2016. AppointmentDay does not contain a national holiday or Sunday.

Regarding previous work, Helmonds 2018 used several machine learning algorithms to find which one is the best for predicting medical no-shows. The research applied Logistic regression, Decision tree, Bagged decision tree, Random Forest, Gaussian Naive Bayes, and Neural network. The result showed that only Logistic regression did not offer meaningful results for medical no-shows based on F1 scores and AUC. Moreover, another research (PANDIRI 2019) on Kaggle aimed to predict which features affect medical no-shows. It used Decision tree, Random forest, and GridSearchCV, and concluded that Gender, Age, Neighbourhood, Scholarship, and Hypertension are top features that affect whether a patient with an appointment will show up or not.

2 EDA

First, we study the target variable, No-show.

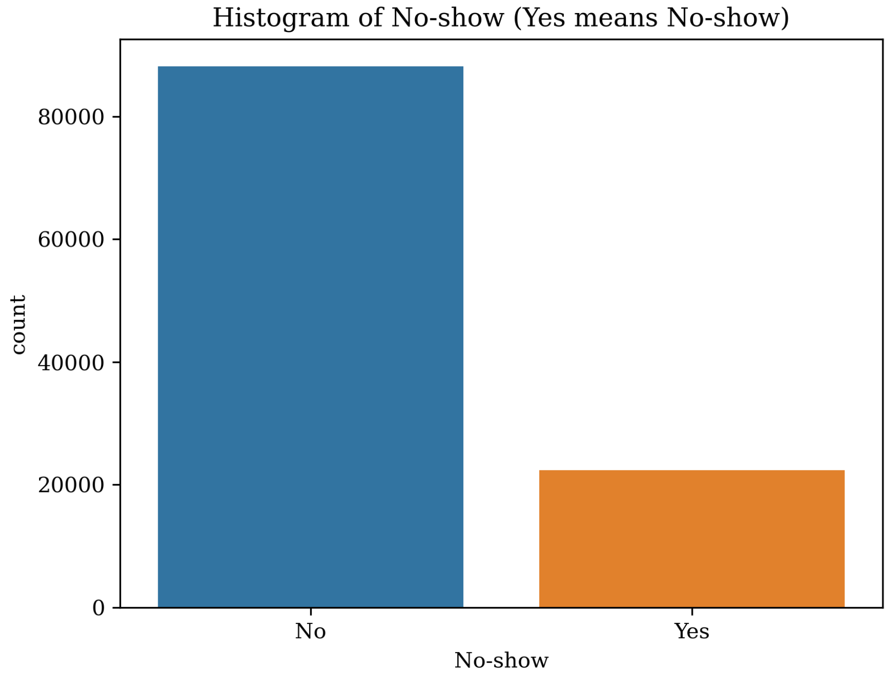

Figure 1: Histogram of No-show (Yes means No-show)

The above plot shows about 20.2% of the patients did not show up (Yes for No-show) for appointments, while 79.8% of the patients showed up (No for No-show). Therefore, patients without no-shows are much more populous.

Moreover, we study three variables that are important and could affect medical no-show.

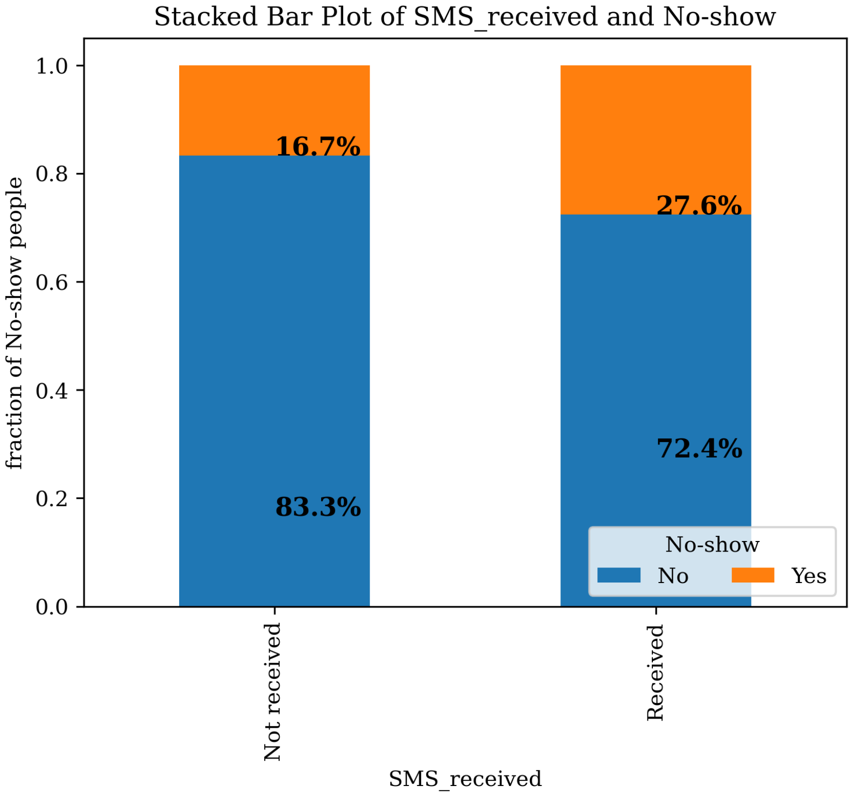

Figure 2: Stacked Bar Plot of SMS_received and No-show

Next, this analysis plots SMS_received and No-show, since it is interesting and unexpected in the way of being against intuition. From the above plot, we can conclude that 16.7% of patients did not receive SMS and did not show up, and

27.6% of patients (10.9% higher) received SMS and did not show up. It means that receiving SMS may increase the chance of patients not showing up for appointments.

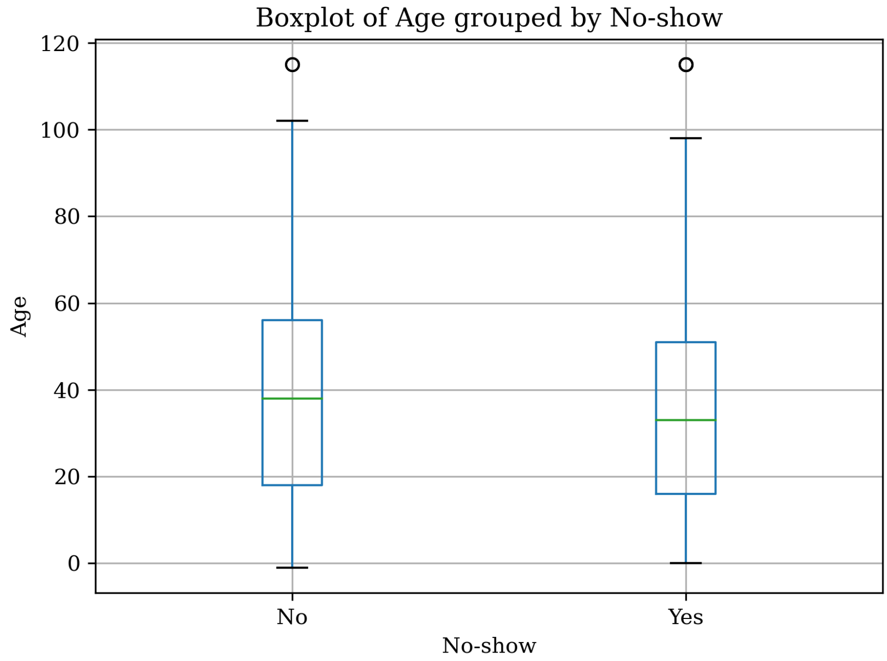

Figure 3: Boxplot of Age grouped by No-show

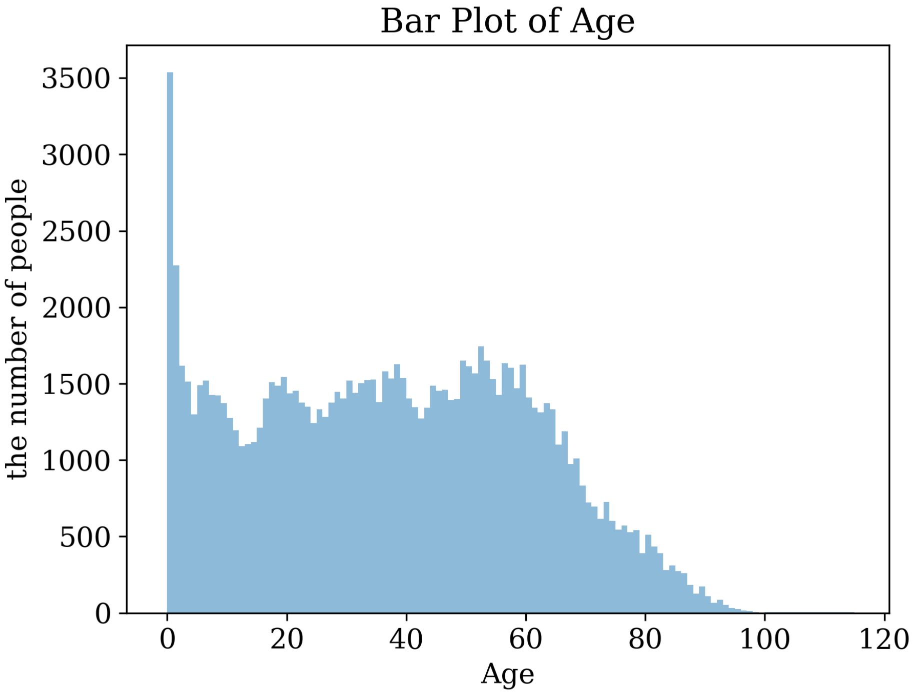

Figure 4: Bar Plot of Age

Also, we plot Age and no-shows, which affects medical no-shows. The first plot shows that the average age of patients who did not show up is lower than those who showed up. Also, it indicates an erroneous value of Age being negative (-1). The second plot shows that there are a lot of observations in Age 0 and 1. For those Age groups, other reasons, namely their patient's, are worth noticing, since usually patients of Age 0 or 1 do not decide on medical show-up. Also, there are 5 people with Age of 115 here, we consider it naturally normal here.

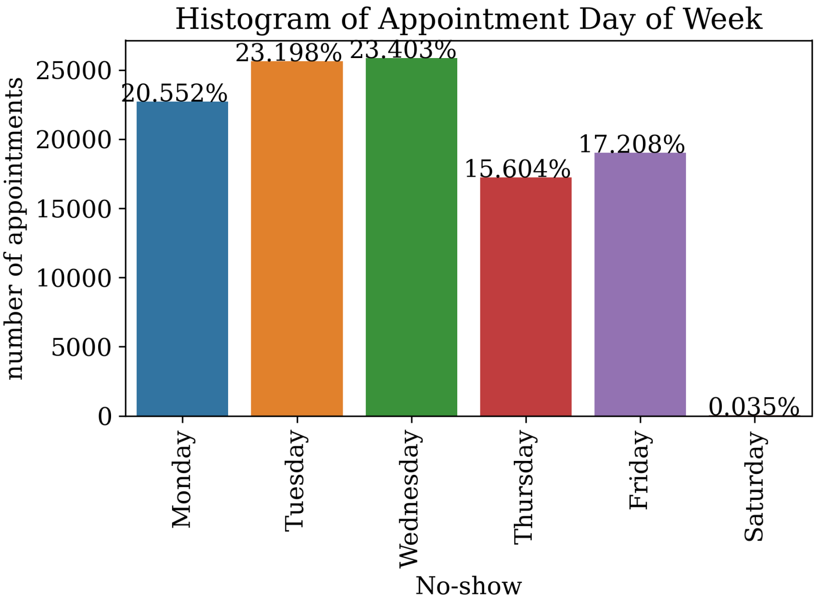

Figure 5: Histogram of Appointment Day of Week

Then, we check the day of week, since it may affect patient's willingness to show up for appointments. From the above pie chart, we can conclude Wednesday and Tuesday are the top 2 weekdays in our data, Saturday has the least percentage, which is only 0.035%. The reason is that oftentimes, health-care facilities do not accept appointments on weekends.

3 Method

3.1 Splitting Data Preprocessing

The dataset is iid. Here, since about 79% percent of the patients are unique, we do not consider the group structure. Also, it is not time series data, since we want to predict if a patient shows up or not.

For splitting data, we use the basic splitting method, since the data is iid, and it is big enough for us not to consider the randomness. We first choose a train data size with 60% of original data, then split the others in half, to make the validation and test data size both 20% percent, which follows the general rule of splitting a dataset that is not very large.

Regarding changes for variables, we change AppointmentDay to AppointmentDayofWeek, to easily see how the day of week affects the prediction. Here, we do not consider appointment time, since it is not disclosed due to privacy reasons. For Neighbourhood, we label each as a unique value. Also, for ScheduledDay, we delete the feature because here we focus more on AppointmentDay, and ScheduledDay barely affects it. For Age, we have one patient of -1 age. Since it makes no sense that a person has negative age, and it only has one data, we delete the row. Moreover, we delete PatientId and AppointmentID because they are only identifiers.

Regarding encoders, for Gender, Scholarship, Hypertension, Diabetes, Alcoholism, Handicap, SMS_received and Neighbourhood, this analysis uses OneHotEncoder since there are categorical variables. Neighbourhood has 81 unique values, and others have either 0 or 1. For Handicap, AppointmentDayofWeek, we use OrdinalEncoder since they have orders. Handicap has logical order, from no handicapped to serious handicapped, and AppointmentDayofWeek has chronological order. For Age, we use MinMaxScaler, since it is not normally distributed, and we know its max and min based on our experiences.

After we processed the feature, we have 110526 data and 11 columns, which means 10 features.

3.2 ML Pipeline

In this section, this report splits the data using 6-2-2 method as mentioned above and preprocesses the data, then calculates the test score. Here, it uses accuracy score as its evaluation metric since it is a classification problem, and we want to measure the ratio of the sum of true positive (TP) and true negatives (TN) out of all the predictions made, since we want the sum of TP and TN to be higher for better medical no-shows prediction to reduce the financial cost. It repeats this method 5 times for 5 different random states, and the function returns the 5 best models and their 5 test scores for each model, and collects 5 test sets. For non-deterministic ML methods, we calculate the standard deviation of test scores, which demonstrates how much uncertainty there is in calculations.

This report utilizes 6 different machine learning models, which are Lasso (L1), Ridge (L2), Elastic Net, Random Forest, KNN, and XGBoost. The tuned parameters and values are listed as follows.

Model	Tuned Parameters	Tuned Values
Lasso (L1)	C	10 numbers spaced evenly on a log scale from -5 to 5 with base 10
Ridge (L2)	C	10 numbers spaced evenly on a log scale from -5 to 5 with base 10
Elastic Net	l1 ratio	0.1, 0.3, 0.5, 0.7, 0.9
Random Forest	max_depth max_features	1, 3, 10, 30, 100 0.25, 0.5, 0.75, 1.0
KNN	n_neighbors weights	1,2,5,10,50 uniform, distance
XGBoost	n_estimators max_depth	100, 200, 300, 400 1, 5, 10, 15, 20

4 Result

The baseline accuracy is 0.79807.

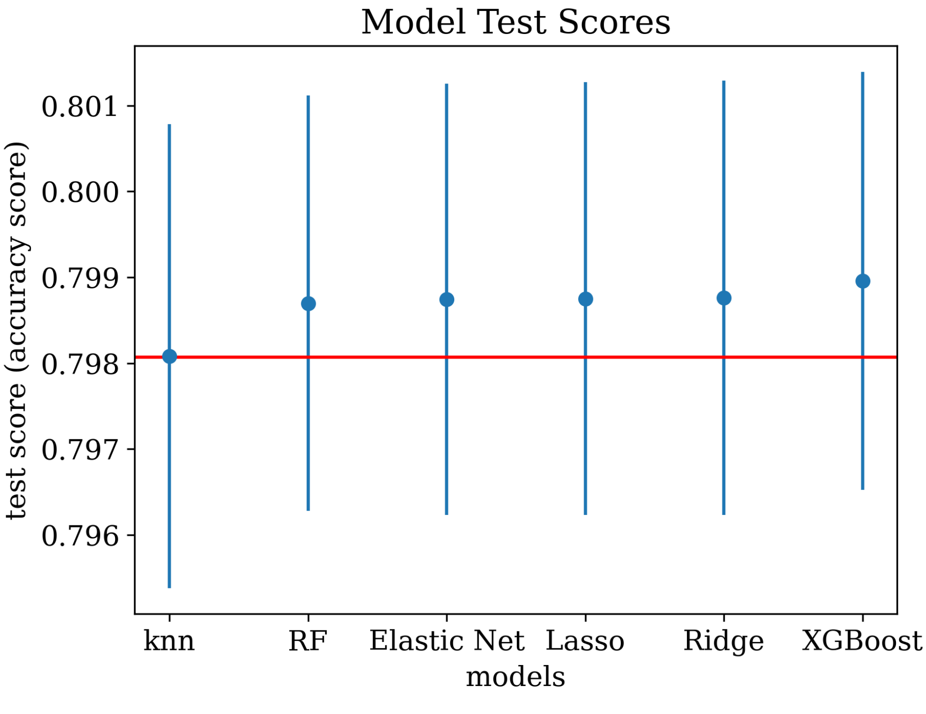

Figure 6: Model test scores

From the above plot, we can conclude that XGBoost has the best mean test score (0.79896). KNN has the worst test score(0.79808), and it is slightly better than the baseline accuracy. Random forest is the second worst, with a mean accuracy score of 0.79870. Lasso (L1), Ridge (L2), and Elastic Net have smiliar mean accuracy scores of 0.79874, 7.79875, and 0.79876. Also, Random forest has the smallest standard deviation (0.00242), and XGBoost has the second smallest one (0.00243). KNN has the largest standard deviation (0.00270). Also, Lasso (L1), Ridge (L2), and Elastic have similar standard deviations, which are 0.00251, 0.00252, and 0.00253.

Overall, this report chooses XGBoost as the one being most predictive, since it has the best test score, and second-best standard deviation, which is very close to the best one.

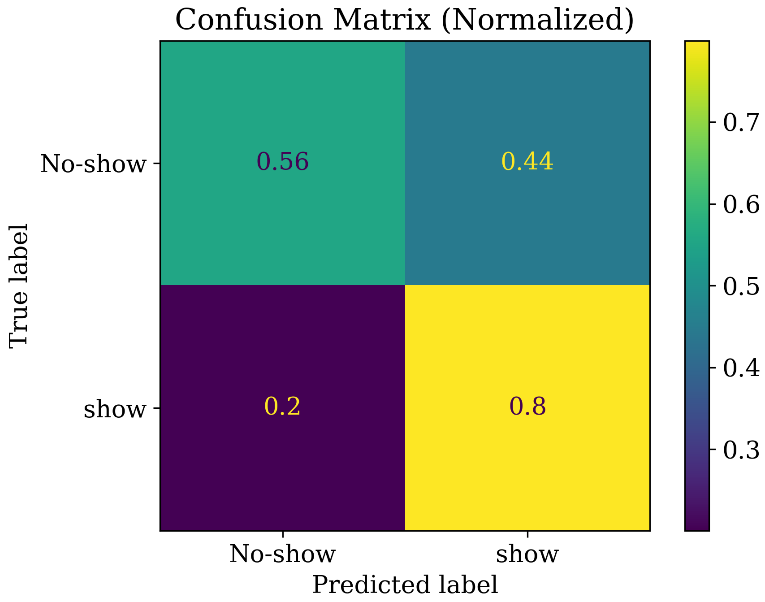

Figure 7: Confusion Matrix (Normalized)

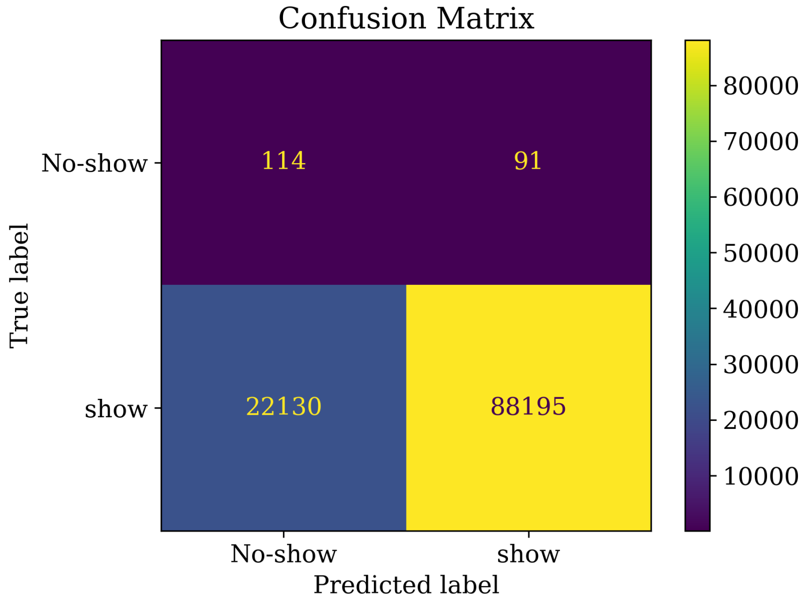

Figure 8: Confusion Matrix

The confusion matrix of 5 XGBoost test sets shows the TP is 114, FP is 91, FN is 22130, and TN is 88195. The normalized confusion matrix of 5 XGBoost test sets shows the TP is 0.56, FP is 0.44, FN is 0.2, and TN is 0.8. Here, true positive (TP) is an outcome where the model correctly predicts the positive class, which means a medical no-show, and true negative (TN) is an outcome where the model correctly predicts the negative class, which means a medical show-up. Therefore, we can conclude that it performs well in predicting medical show-ups, but not very well on medical no-shows.

Moreover, the f1 score is 0.02471, the f0.5 score is 0.01016, and the f2 score is 0.00639. Here, all f scores are so low because unnormalized TP and FP and low compared to TN and FN.

Therefore, we want to stick with accuracy score for measurement.
Next, we choose one XGBoost which performs best on both train and test dataset. It has an accuracy score of 0.79932 on train set, and 0.80186 on test score. Also, we study its global and local importance.

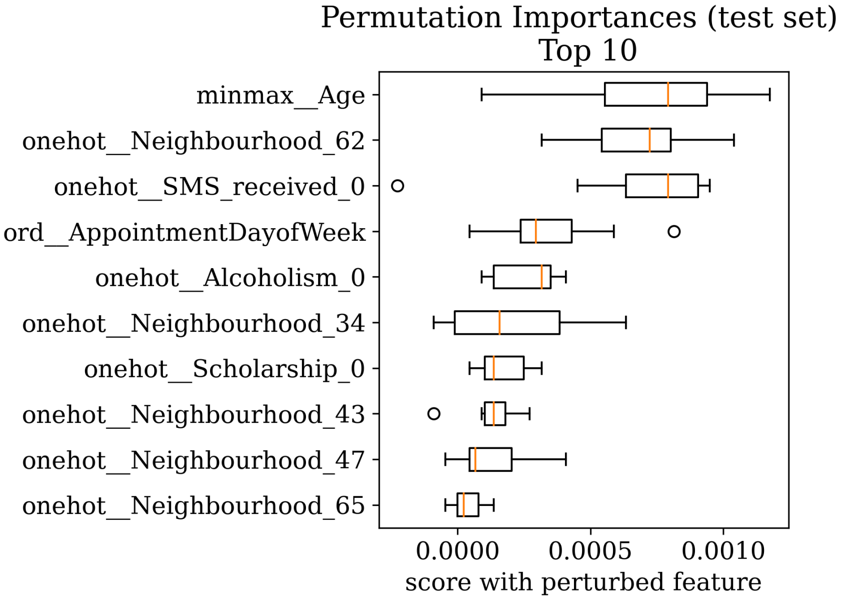

Figure 9: Permutation Importances (test set) Top 10

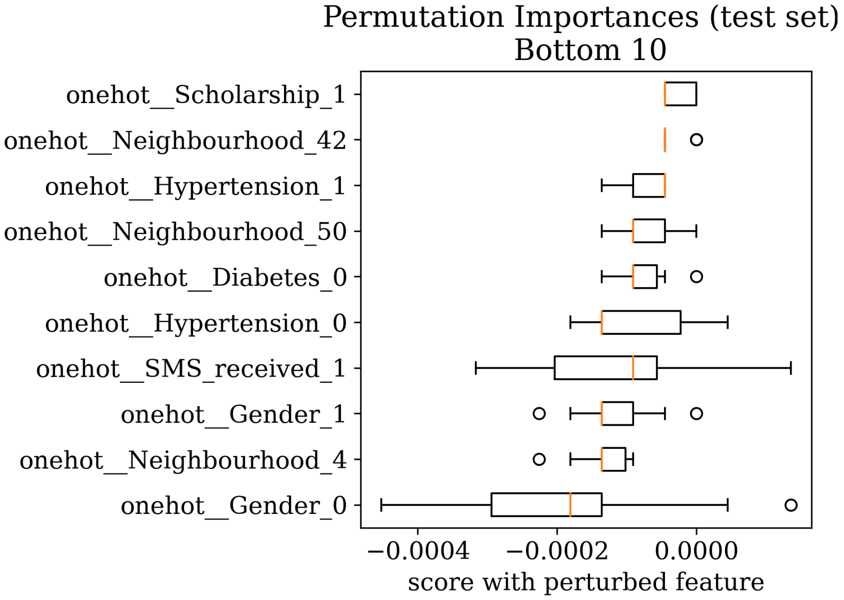

Figure 10: Permutation Importances (test set) Bottom 10

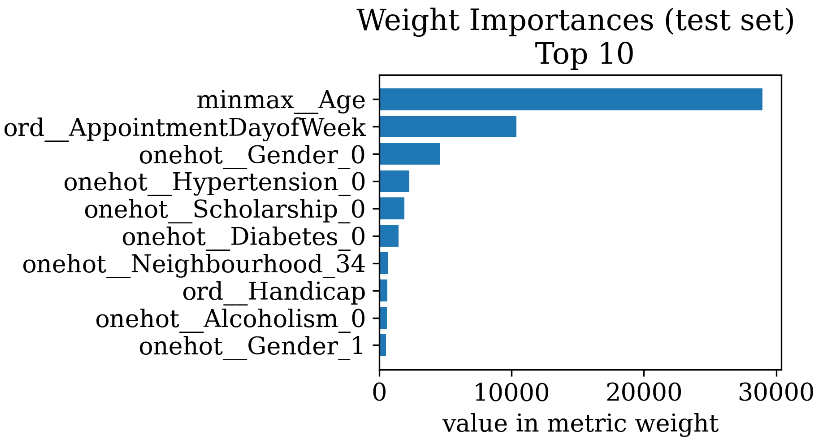

Figure 11: Weight Importances (test set) Top 10

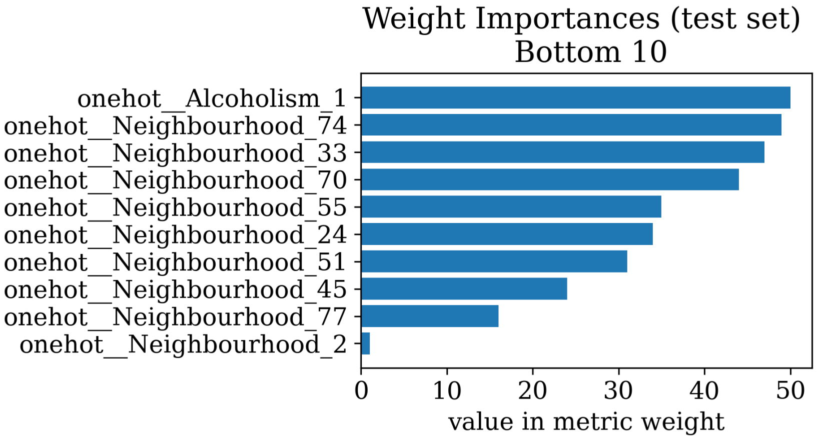

Figure 12: Weight Importances (test set) Bottom 10

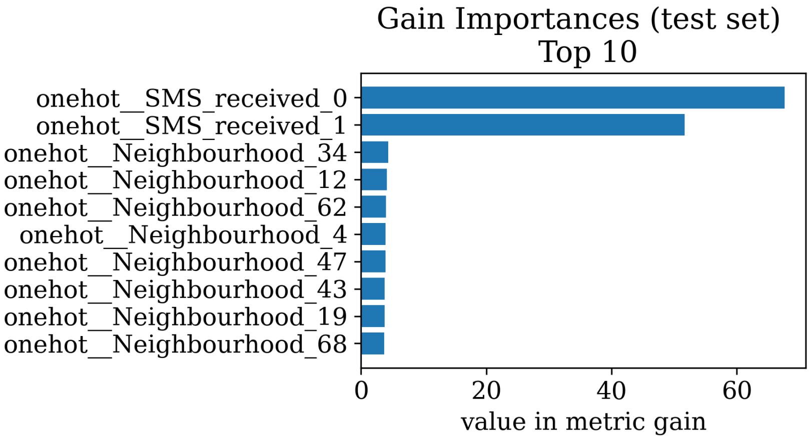

Figure 13: Gain Importances (test set) Top 10

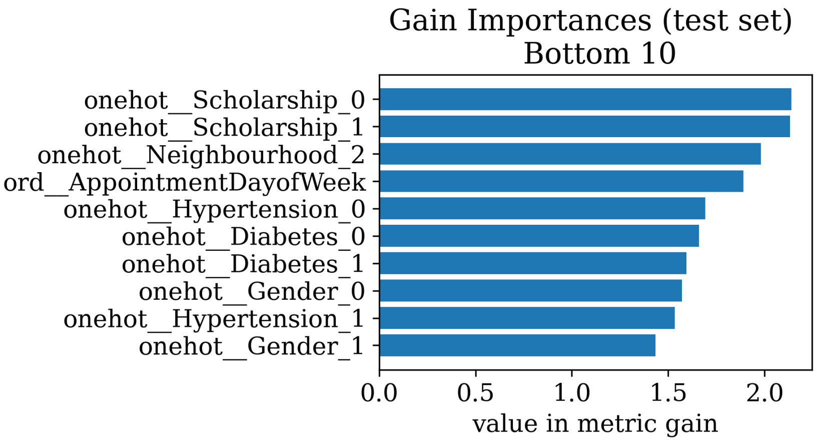

Figure 14: Gain Importances (test set) Bottom 10

SHAP Global Importances (test set) Top 10

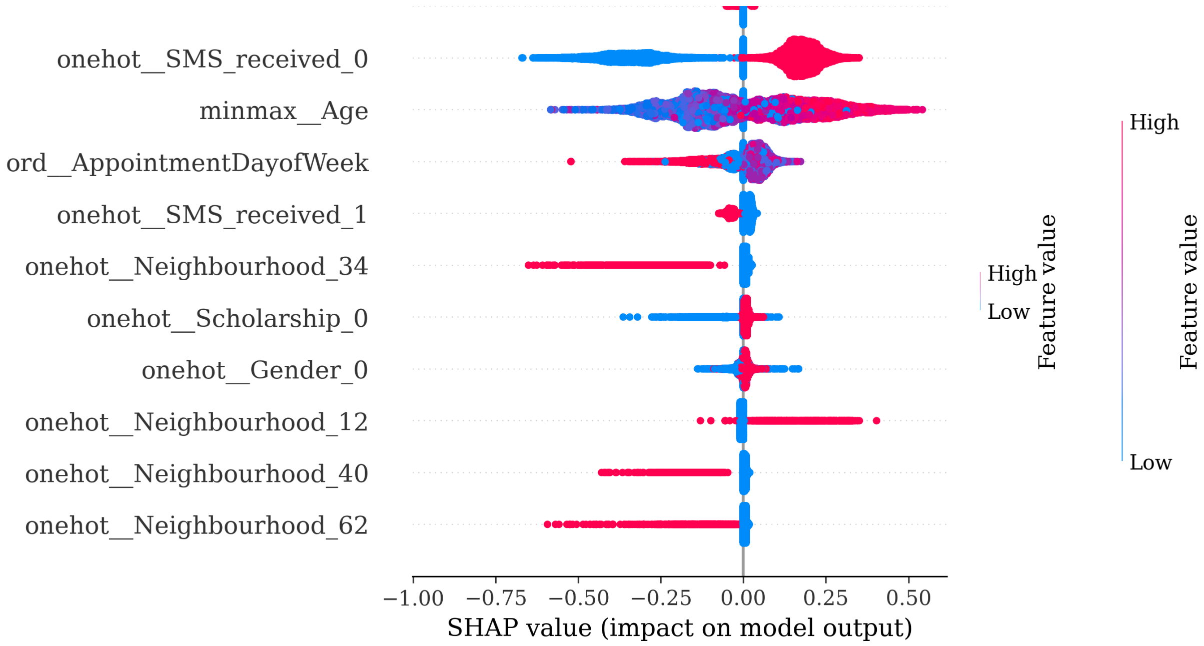

Figure 15: SHAP Global Importances (test set) Top 10

SHAP Global Importances (test set) Bottom 10

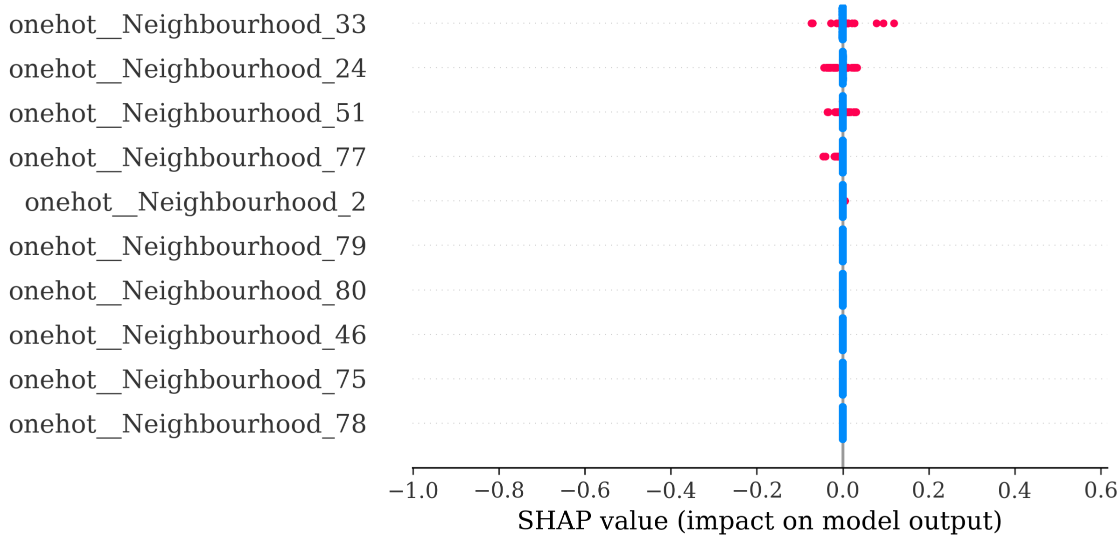

Figure 16: SHAP Global Importances (test set) Bottom 10

We calculate permutation, weight, gain, and SHAP as global importance. Here, we need to consider both weight and gain, since the dataset is slightly imbalanced for no-show-up patients. Here, some features frequently appearing on top 5 are onehot_SMS_received_0, minmax_Age, ord_AppointmentDayOfWeek, and the least frequent important one is onehot_Scholarship_1.

SHAP Local Importances (test set) near Index 0

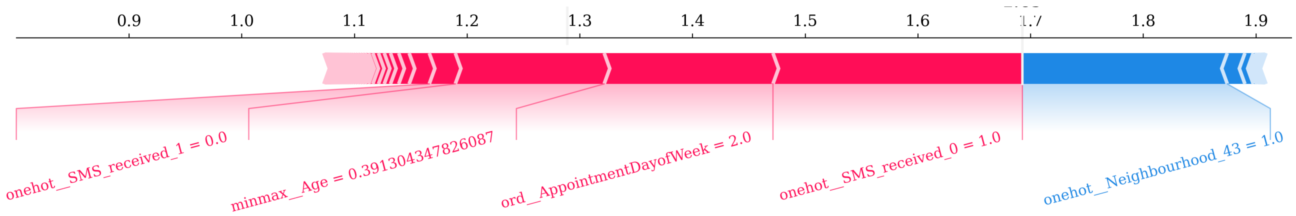

Figure 17: SHAP Local Importances (test set) near Index 0

SHAP Local Importances (test set) near Index 5000

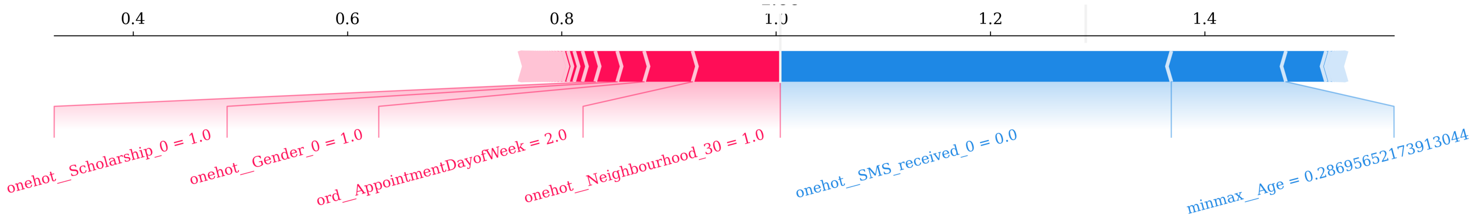

Figure 18: SHAP Local Importances (test set) near Index 5000

SHAP Local Importances (test set) near Index 15000

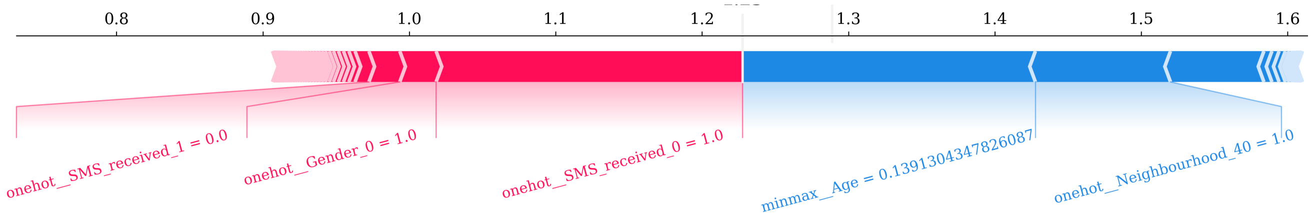

Figure 19: SHAP Local Importances (test set) near Index 15000

For local importance, we choose indices 0, 5000, and 15000, and we found a similar result: onehot_SMS_received_0, minmax_Age, ord_AppointmentDayOfWeek are the most important.

The foundation is important since it shows us the way to predict medical no-shows. For example, we can send more SMS notifications to make patients show up. Also, it is interesting because it shows age and AppointmentDayOfWeek can be important factors to consider. Especially for AppointmentDayOfWeek, as mentioned above, we know that there is no holiday or Sunday, and only a few observations on Saturday. It is worthwhile to study further how AppointmentDayOfWeek affects medical no-shows.

5 Outlook

This report does not use sophisticated Feature Engineering, and ignored correlation among categorical features, which can be used for better model performance and interpretability.

Feature Engineering could be a potential method for better model performance. For example, we can further categorize neighborhoods by finding some similar aspects. Since the global importance shows some neighborhoods are important, it is worth paying more attention to how neighborhoods affect medical no-shows.

On the other hand, a correlation among categorical features is possible, such as Diabetes and Hypertension. Research Midha et al. 2015 shows that the risk of developing hypertension is 1.5-2.0 times higher in diabetic patients in contrast to nondiabetic patients. Therefore, operations such as deleting one feature may change the model's performance.

Moreover, extra data could be beneficial, such as whether a patient shows up for an appointment at other medical facilities. The reasons behind a person having a medical no-show are sometimes complicated. Therefore, we need more data from more sources to improve the prediction.

References

Helmonds, Joep (June 2018). "Predicting no-shows in Brazilian primary care". LIACS. URL: https://theses.liacs.nl/pdf/2017-2018-HelmondsJoep.pdf.

Hoppen, Joni (2015). "Medical Appointment No Shows, version 5". URL: https://www.kaggle.com/datasets/joniarroba/noshowappointments.

Midha, Tanu et al. (2015). "Correlation between hypertension and hyperglycemia among young adults in India". In: *World Journal of Clinical Cases: WJCC* 3.2, p. 171.

PANDIRI, SAMRAT (May 2019). *Predict show/noshow - eda+visualization+model*. URL: https://www.kaggle.com/code/samratp/predict-show-noshow-eda-visualization-model.