Report on “No-show to scheduled clinic appointment prediction”

by Gaukhar Makhmetova

DSC 789-001 - Spring 2021: Strategic Capstone Projects

Teacher: Zahra Sedighi Maman

05/18/2021

Introduction

The problems in the health industry are the ones of highest importance and resolving the issues that are present in hospitals around the globe is crucial for sustaining strong health infrastructure and boosting the quality of service provided to people. One of the most disruptive problems that healthcare industry faces nowadays is a high proportion of the appointments scheduled in clinic being cancelled because of patients not showing up for them. No-shows of patients have a significant impact on healthcare systems, including lower clinical efficiency, higher costs, limited access to care and lower health outcomes. Negative effects are also felt by patients who keep their appointments, such as dissatisfaction with long lead time and perception of the overall decrease in service quality. In recent research conducted by Harris & Samorani (2020), the no-show rate was stated to be 23% on average, this variability in patient's attendance makes it harder for clinics to provide crucially service for those in need in an effective rate. Research shows that hospitals try to resolve the issue by overbooking appointments slots, which means they schedule more than 1 person for the same time slot. However, it is hard to believe that this is the best solution that can be found to tackle this problem. Moreover, patient interventions are attempted to reduce no-show rates, with phone calls, short message service (SMS) texts, and emails being particularly impactful. However, in spite of all these initiatives, the no-show rates cited for USA ranged from 5% to as high as 55% (Lenzi, Ben, & Stein, 2019). Hospitals with high percentage of no-show rate should reallocate their resources in a right way. Therefore, the aim of this research is to identify main reasons behind patients' no-shows to appointments by evaluating trends between patients' information and no-show rate.

While conduction this research the question that will be targeted to be answered are following:

1. Is it possible to predict no-show to appointment by the use of machine learning techniques?
2. What are the important features in order to predict the no-show to appointment?
3. What is the importance of the features?
4. How does the presence of outliers change the prediction performance?
5. How do the different sampling techniques affect the prediction of no-show to appointments?

6. How do the different feature selection methods affect the prediction of no-show to appointments?
7. How does the prediction performance change for more complex analytical models?
8. What is most efficient strategy outline to increase the performance of no-show to appointments prediction?

Data Overview

The dataset represents healthcare information collected in Espirito Santo, Brazil. The dataset that was downloaded from Kaggle (https://www.kaggle.com) contains 72,607 cases observations and 16 variables in total (refer to the table below.

Variable	Definition
PatientId	Patient ID
AppointmentID	Appointment ID for patient
Age	Patient age
Gender	Patient sex
Month	Month of the appointment
AppointmentDay	Day of the appointment
Waiting_time..minute.	Time hospital waited for patient
Calling.time..hour.in.a.day.	The hour of calling to schedule an appointment
SMS_received	Status of receiving reminder message or call
Alcoholism	Status of being an alcoholic
Financial_aid	Status of having a financial support
Handicap	Status of being handicap
Hypertension	Status of having high blood pressure
Diabetes	Status of being diabetic
time_b_appointment..day.	Number of days between two consecutive scheduled appointments
Show.up	yes – showed up, no – no-show

Methodology

In order to create a predictive model for no-show to appointments, there were several steps undergone for data preparation and modelling.

Firstly, exploratory data analysis was conducted by downloading data from Moodle Capstone Project source provided by course leader and imported in Tableau. In order to understand all variables available, bar charts were plotted for categorical variable and line graphs

were projected for continuous variables. The data file, data description and Tableau variable analysis files are attached in the submission.

Secondly, for performing data preparation and modeling phases RStudio script file was created and function rm(list = ls()), graphics.off() and gc() were used to clear global environment and close all graphics. Functions str(),summary() and head() were used for data overview and exploring data formatting.

```
> str(noshow)
'data.frame':   72607 obs. of  16 variables:
 $ PatientId                  : num  22638656 22638656 52168938 52168938 64851211 ...
 $ AppointmentID              : int  5580835 5715081 5704816 5607220 5683383 5697532 5742958 5743266 5651939 5786272 ...
 $ Age                        : int  22 23 28 28 29 29 21 21 3 3 ...
 $ Gender                     : chr  "F" "F" "F" "F" ...
 $ AppointmentDay             : chr  "Tuesday" "Wednesday" "Monday" "Tuesday" ...
 $ Month                      : int  5 6 5 5 5 5 5 6 5 6 ...
 $ Calling_time..hour.in.a.day.: int  7 13 16 11 7 16 8 8 9 8 ...
 $ Waiting_time..minute.      : int  19 21 0 27 2 4 0 9 10 0 ...
 $ Financial_aid              : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Hypertension               : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Diabetes                   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Alcoholism                 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Handicap                   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ SMS_received               : int  1 1 0 0 0 0 0 1 0 0 ...
 $ time_b_appointment..day.   : int  0 36 0 1 0 4 0 9 0 26 ...
 $ Show.up                    : chr  "yes" "yes" "yes" "yes" ...
> summary (noshow)
   PatientId         AppointmentID         Age            Gender          AppointmentDay         Month
 Min.   :2.264e+07   Min.   :5030230   Min.   :  0.00   Length:72607       Length:72607       Min.   :4.000
 1st Qu.:4.170e+12   1st Qu.:5644562   1st Qu.: 18.00   Class :character   Class :character   1st Qu.:5.000
 Median :3.160e+13   Median :5682686   Median : 37.00   Mode  :character   Mode  :character   Median :5.000
 Mean   :1.475e+14   Mean   :5678657   Mean   : 37.25                                        Mean   :5.207
 3rd Qu.:9.390e+13   3rd Qu.:5726350   3rd Qu.: 55.00                                        3rd Qu.:5.000
 Max.   :1.000e+15   Max.   :5790481   Max.   :115.00                                        Max.   :6.000
 Calling_time..hour.in.a.day. Waiting_time..minute. Financial_aid    Hypertension       Diabetes          Alcoholism
 Min.   : 6.00                Min.   : -6.000       Min.   :0.000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
 1st Qu.: 8.00                1st Qu.:  0.000       1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
 Median :10.00                Median :  4.000       Median :0.000   Median :0.0000   Median :0.00000   Median :0.00000
 Mean   :10.88                Mean   :  9.253       Mean   :0.104   Mean   :0.1973   Mean   :0.07315   Mean   :0.03462
 3rd Qu.:14.00                3rd Qu.: 14.000       3rd Qu.:0.000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
 Max.   :20.00                Max.   :179.000       Max.   :1.000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
    Handicap        SMS_received    time_b_appointment..day.   Show.up
 Min.   :0.00000   Min.   :0.0000   Min.   : 0.00            Length:72607
 1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.: 0.00            Class :character
 Median :0.00000   Median :0.0000   Median : 1.00            Mode  :character
 Mean   :0.02387   Mean   :0.3136   Mean   : 5.62
 3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.: 8.00
 Max.   :3.00000   Max.   :1.0000   Max.   :40.00
```

Following framework was used for data preparation and modelling phases:

1) Since there were categorical variables present in data set, fastDummies library was used to perform **onehot encoding** and dummy variables for AppointmentDay, Month Gender and Show.up were created. Onehot encoding is very effective tool for converting categorical variables into numbers (Alshaya, McCarren, & Al-Rasheed, 2019).
2) Gender_M, Show.up_no were deleted because they hold same information as alternative dummy colums created.

noshow1 <- dummy_cols(noshow, select_columns = c('Gender', 'Show.up', 'AppointmentDay', 'Month'), remove_selected_columns = TRUE)

3) PatientId, AppointmentID are irrelevant for the research, so they were deleted as well.

4) By visualizing AppoitmenttDay it was noticed that there were only 19 observations for Saturday and therefore it was decided to delete them dataset.

```
noshow1 = subset(noshow1, select= -c(Gender_M, Show.up_no, PatientId, AppointmentID, AppointmentDay_Saturday))
```

5) In the next step research was done to identify **zero and near zero variance predictors** by using caret library and nearZeroVar() function. By utilizing this function researchers are able to identify frequency ratio and make appropriate adjustment for their data preparation stage (Huang, Hanauer, 2014).

```
library(caret)
Near_Zero_Var <- nearZeroVar( noshow1,
                    freqCut = 95/5,
                    uniqueCut = 10,
                    saveMetrics = TRUE)
```

6) The data set was checked for any missing values by using function sum(is.na()) and none were detected.

7) The next step included detecting **outliers** and removing them by using **Cook's method** by use of function cooks.distance() (Parikh, Gupta, Wilson, 2010). All influential observations were removed from dataset, which accounted for 4246 observations.

```
mod <- lm(Show.up_yes ~ ., data=noshow1)
cooksd <- cooks.distance(mod)
```

8) The dataset was divided into train and test using **balanced split** in proportion of 80% for train and 20% for test. Balanced split is one of the most effective ways to divide data into train and test, it is low computationally expensive and not time consuming (AlMuhaideb, Alswailem, Alsubaie, 2019)

```
TrainIndex <- createDataPartition(noshow1$Show.up_yes, p = .8, list = FALSE, times = 1
```

9) Train dataset was imbalanced and it was decided to use **ADASYN** to balance dataset. ADASYN is a function used from library(smoteFamily) and it is used for syntetic sampling method (Alshaya, McCarren, Al-Rasheed, 2019).

```
noshow1_train_balanced <-ADAS(noshow1_train[,-12], noshow1_train[,12], K=5)
```

The target variable in balanced dataset became “class” which format after balancing was character and it was changed to numeric.

```
noshow1_train_balanced[["data"]]$class<-
as.numeric(noshow1_train_balanced[["data"]]$class)
```

10) In prior research papers different methods were used for feature selection phase and Lasso regression and Feature Importance had relatively high results. Therefore, for enhancing results of the model feature selection was done by using Feature selection by importance and Lasso regression.

Firstly features by importance were plotted by creating a GLM regression model and using varImp() function on the model and plotting variables by importance.

Feature Importance

```
control <- trainControl(method="repeatedcv", number=10, repeats=5)
model <- train(class ~., data=noshow1_train_balanced[["data"]], method="glm",
preProcess="scale", trControl=control) #Creating a model
importance <- varImp(model, scale=FALSE) # estimate variable importance
```

Only 80% of the most significant features remained for further analysis.

Lasso

```
x = model.matrix(class ~.,noshow1_train_balanced[["data"]]) # matrix of predictors
y = noshow1_train_balanced[["data"]]$class            # vector y values
lasso_model <- cv.glmnet(x, y, alpha=1,family = "binomial")
```

11) Final stage of modelling phase included choosing best performing model, which was GLM and Random Forest.

In order to create a model target variables in train and test datasets were turned into factors.

GLM

```
mylogit <- glm(class ~., data = noshow1_train_balanced[["data"]], family = "binomial")
summary(mylogit)
```

The confusion matrix was created in order to examine model performance.

```
probability <- predict(mylogit, newdata = noshow1_test, type = "response")
test_predict1 <- rep( 0, 17085)
test_predict1[probability >=0.5] = 2
test_predict1[probability < 0.5] = 1
test_predict1<- as.factor(test_predict1)
levels(noshow1_test$Show.up_yes) <- levels(test_predict1)
confusionMatrix(test_predict1, noshow1_test$Show.up_yes)
```

Random Forest:

```
random_forest <- randomForest(class ~., data = noshow1_train_balanced[["data"]], sampsize=.1*length(y), ntree=5000, maxnodes=24) #Created a model
predict_random_forest <- predict(random_forest, newdata = noshow1_test[-10]) #predicted values for test dataset
```

After leveling target variable and predicted values, built confusion matrix.

```
confusionMatrix(predict_random_forest,noshow1_test$Show.up_yes)
```

Decision Tree:

```
decision_tree <- rpart(class ~., data=noshow1_train_balanced[["data"]], method='class') #creating model
rpart.plot(decision_tree, extra = 110) #plotting decision tree
predict_decision_tree <-predict(decision_tree, noshow1_test, type = 'class') #predicting target value in test dataset
confusionMatrix(noshow1_test$Show.up_yes, predict_decision_tree) # confusion matrix
```

12) The model was evaluated by identifying Sensitivity, Specificity, Precision, Gmean, Accuracy, AUC and running time was measured for last two best performing models.

Results

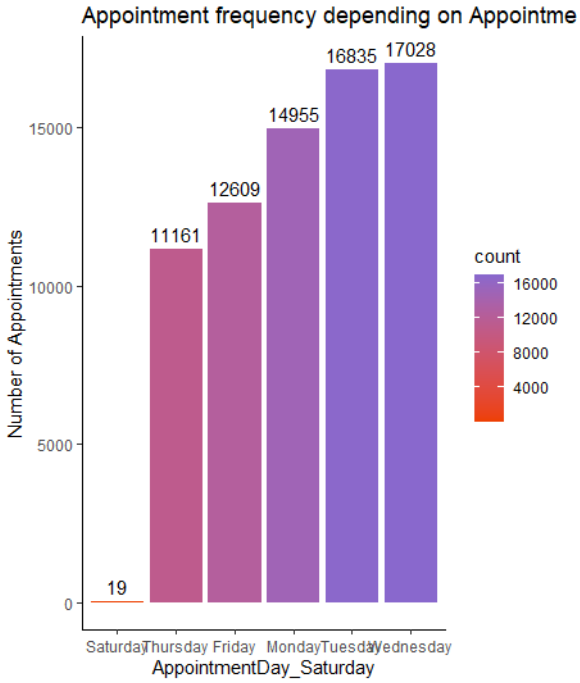
While completing each step of the methodology framework, there were a number of variations in choice of how to deal with different problems. The model was evaluated after each step undertaken and the results for different variations were compared according to model evaluation parameters.

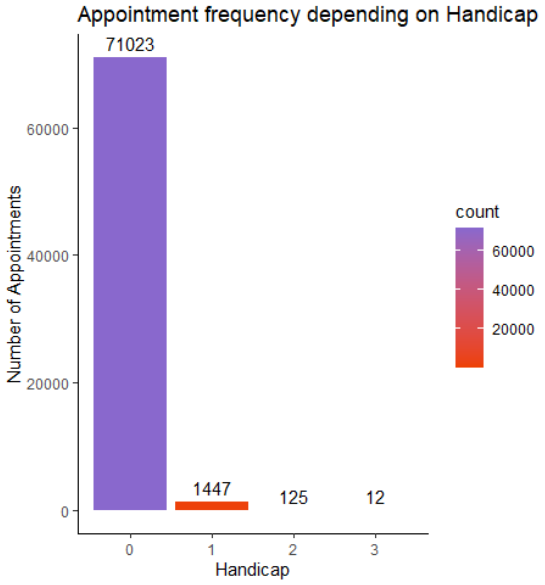
1. There were 3 near-zero variance predictors identified including Alcoholism Handicap and Month_4. Month_4 was decided to delete from Dataset because further in research it was eliminated in Feature selection phase.

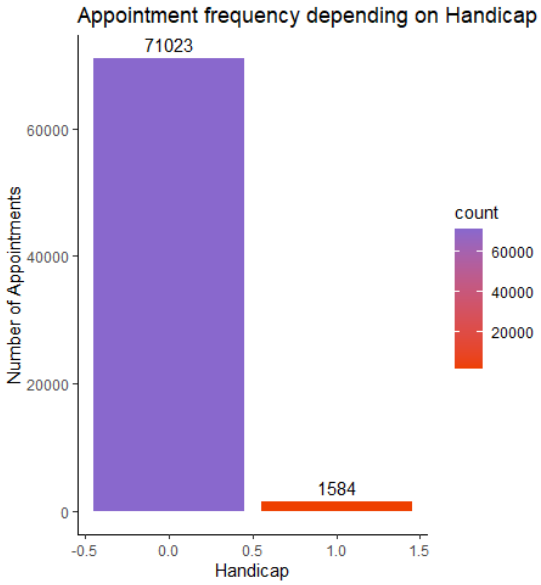
	freqRatio	percentUnique	zeroVar	nzv
Alcoholism	27.881066	0.002754555	FALSE	TRUE
Handicap	49.082930	0.005509111	FALSE	TRUE
Month_4	33.740191	0.002754555	FALSE	TRUE
Age	1.823838	0.139105045	FALSE	FALSE
Calling_time..hour.in.a.day.	1.170058	0.020659165	FALSE	FALSE

Data visualization showed that appointment frequency for Handicap levels 1,2 and 3 were extremely comparing to whole who did not handicap. It was decided to combine levels 1, 2and 3 together and see if that will enhance the frequency ratio. However, the count for observation with handicap levels 1,2 and 3 accounted for 1584 comparing 71004 of those without handicap.

Data visualization for Alcoholism showed that only 2214 observations had Alcoholism, comparing to 70093 of those who did not. Both zero variance predictors were kept in data set, to see if they will be shown as important predictors in feature selection step.


Appointment frequency depending on Appointme
Number of Appointments
15000
10000
5000
0
19
11161
12609
14955
16835
17028
count
16000
12000
8000
4000
Saturday
Thursday
Friday
Monday
Tuesday
Wednesday
AppointmentDay_Saturday


Appointment frequency depending on Handicap
Number of Appointments
60000
40000
20000
0
71023
1447
125
12
count
60000
40000
20000
0
1
2
3
Handicap


Appointment frequency depending on Handicap
Number of Appointments
60000
40000
20000
0
71023
1584
count
60000
40000
20000
-0.5
0.0
0.5
1.0
1.5
Handicap

1. The comparison analysis was made in order to evaluate *How does the presence of outliers change the prediction performance?*

Results of the Logistic regression before and after removing outliers while keeping Appointment day in levels from 1 to 5 were following:

Model	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM_before_Cook's	0.018	0.990	0.326	0.135	0.79	0.562
GLM_after_Cooks	0.020891	0.987135	0.305	0.144	0.781	0.547

*The GLM model performance was chosen to present how changes effect modelling stage due to its low computational power and time efficiency (Goffman, Harris, May, et.al., 2017). The previous research related to no-show to appointment used logistic regression in analysis and showed considerably high results (Lenzi, Ben, & Stein, 2019).

According to these results the model overfits training dataset before and after cooks Distance and AUC decreased after using this outlier removing method.

Same steps were undergone for the case where dummy variables were created for AppointmentDay. The number of outliers decreased to 3425 observations, which is 821 observations less comparing to previous version.

Here are the results of logistic regression:

Model	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM_before_Cook's	0.022	0.98	0.368	0.15	0.783	0.57
GLM_after_Cooks	0.040	0.985	0.305	0.2	0.781	0.6

Results did not show a lot of enhancement and overfitting problem still stands, but reduction of outliers is a valuable reason to choose second version of the code where dummy variables were created for AppointmentDay.

2. In the modelling phase, in the first step the answer for the following question was found: *How do the different sampling techniques affect the prediction of no-show to appointments?*

The dataset was divided into train and test using balanced split in proportion of 75% for train and 25% for test and by using K-fold validation. Here is the table that presents results for GLM for both cases. Results are slightly better while using K-fold cross validation, however the

computational power required for this method is higher comparing to balanced split. Therefore, balanced split was the method chosen for further modelling.

Model	Train/test split	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	Balanced Split	0.040	0.985	0.305	0.2	0.781	0.6
GLM	K-fold cross-validation	0.043	0.983	0.351	0.2	0.821	0.6

3. Train dataset was imbalanced, and it was decided to use DBSMOTE, BLSMOTE and ADASYN to balance dataset and evaluate which balancing method will result in higher results.

   The following table presents results before and after balancing train dataset.

Model	Train/test split	Balance	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	K-fold cross-validation	DBSMOTE	0.575	0.691	0.282	0.631	0.671	0.583
GLM	Balanced Split	DBSMOTE	0.591	0.683	0.277	0.635	0.668	0.584

Results indicate that balancing dataset in terms of target variable Show.up_yes helped to tacked an issue of overfitting and Gmean increased considerably, but the accuracy of the model decreased by about 16% and AUC decreased by 2%.

Balanced SMOTE and ADASYN were also used in order to evaluate which method will lead to better results. ADASYN balancing method led to a higher results compering to DBSMOTE, while the time for running a code was considerably lower. BLSMOTE was applied to train dataset as well and according to results in the table below, it can be considered that model overfits, thereore the decision was made in favor of ADASYN.

Model	Balance	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	DBSMOTE	0.591	0.683	0.277	0.635	0.668	0.584
GLM	ADASYN	0.6832	0.6818	0.311	0.682	0.682	0.6111
GLM	BLSMOTE	0.222	0.9199	0.368	0.4519	0.798	0.608

**In the further analysis DBSMOTE was used as a balancing method. Results on final stage are presented for both balancing methods. Resetting some setting on later stages of research led to higher results for balancing train dataset with ADASYN.*

4. Answering research question: *What is the importance of the features?*

Feature selection by importance indicated that Calling_time..hour.in.a.day, Month_6, Month_4 and AppointmentDay_Wednesday were the features of least importance.

Secondly, Lasso regression was used for next step of feature selection and it automatically penalized variables that were not important for target variable.

Model	Feature Selection	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	Importance	0.572	0.695	0.288	0.631	0.673	0.585
GLM	Lasso	0.598	0.683	0.279	0.6287	0.668	0.586
GLM	Importance+Lasso	0.568	0.692	0.277	0.635	0.673	0.586

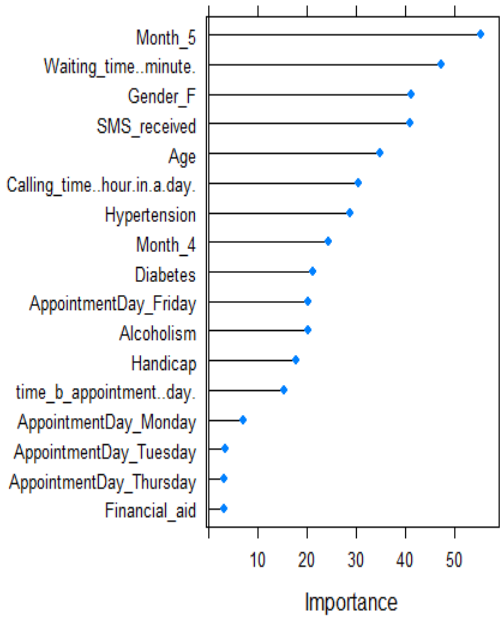
There were not enhancements of results after feature selection using Importance and Lasso. Therefore, it was chosen to run Elastic Net regression to see if it will improve the results of GLM model.

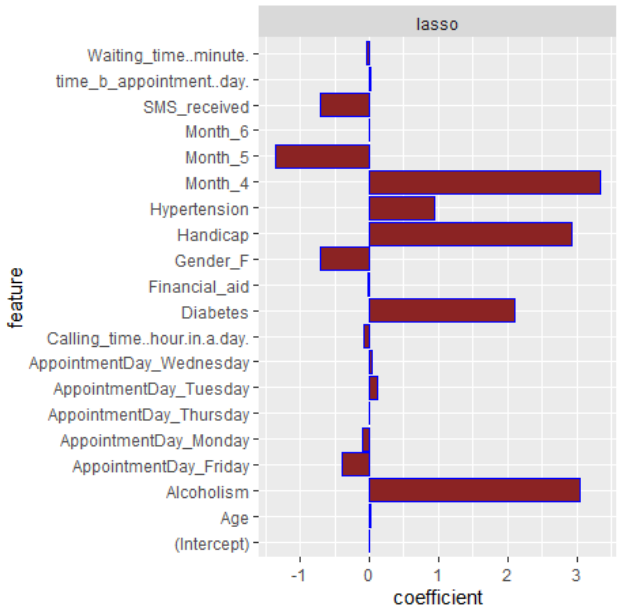
Model	Feature Selection	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	Elastic Net	0.569	0.694	0.282	0.628	0.673	0.583

Results did not change after using Elastic Net instead of Lasso and feature choice by importance. Therefore, it was decided to choose less computational power and less time-consuming way for feature selection, which is Lasso and feature selection by importance.

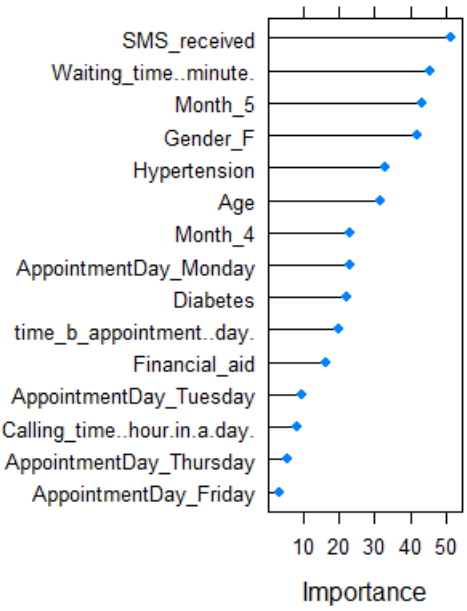
5. Answering research questions*: What are the important features in order to predict the no-show to appointment?*

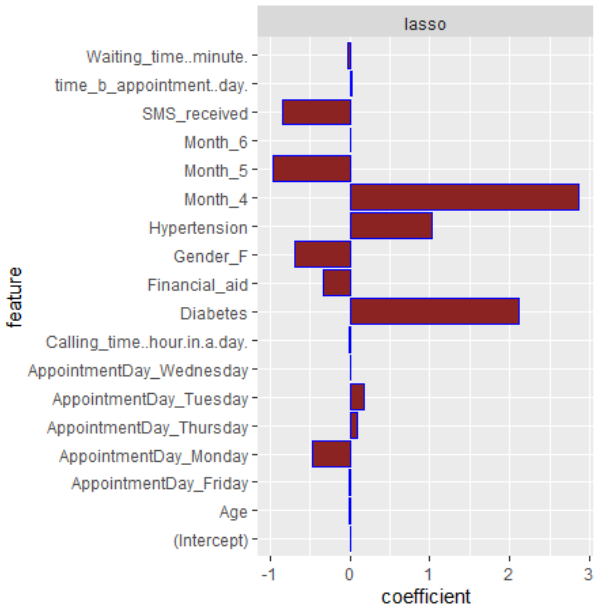
GLM variable importance





Reminder: Handicap and Alcoholism are near zero variance predictor, lets see if the variable will be removed from dataset.





Results of the model with and without Handicap and Alcoholism:

Model	Handicap+ Alcoholism	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	yes	0.568	0.692	0.277	0.635	0.673	0.586
GLM	no	0.578	0.694	0.284	0.633	0.674	0.5857

Presence or absence of Handicap and Alcoholism variables in the dataset does not change results of the regression, so it was decided to keep them in dataset for the analysis. However, it is important to note how different feature selection methods perform differently. Only a number of features that Lasso recognized as highly significant were similar in comparison with feature Importance method. Mostly they differed, however, they both did assign 0 as a coefficient for AppointmentDay_Wednesday and Month_6. In order to achieve a higher results, there were a number of combinations tested. Eventually, after feature selection following features were deleted from the dataset:
AppointmentDay_Thursday, AppointmentDay_Friday, AppointmentDay_Wednesday, Calling_time..hour.in.a.day., Age, Month_6.

The performance of GLM before and after removing following variables:

Model	Removed variables	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	no	0.568	0.692	0.277	0.635	0.673	0.586
GLM	yes	0.556	0.720	0.295	0.633	0.698	0.59

Evaluation parameters of the model increased slightly, but not considerably. However, it had more significant result for Random Forest later in the research.

6. Final stage of modelling phase included running GLM, Random Forest and Decision tree in order to identify the model that performs the best according to selected parameters. Decision tree was highlighted in a previous research related to no-show to appointment, however, was conducted on different dataset (Ferro, Brailsford, Bravo et.al., 2020)

Answering research question: *How does the prediction performance change for more complex analytical models?*

Model	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	0.568	0.692	0.277	0.635	0.673	0.586
Random Forest	0.106	0.958	0.835	0.319	0.810	0.594
Decision Tree	0.317	0.894	0.894	0.532	0.707	0.606

After receiving results above, the problem of overfitting for Random Forest and Decision Tree could clearly be seen. In order to resolve an issue of overfitting, the tuning parameters of the model were changed. Results below represent the case where the issue of overfitting was resolved for Random Forest in expense of accuracy lower by 8%. However, comparing results of GLM with ADASYN and Random Forest with BDSMOTE, it can be seen that even after resolving overfitting issue, results for GLM are higher.

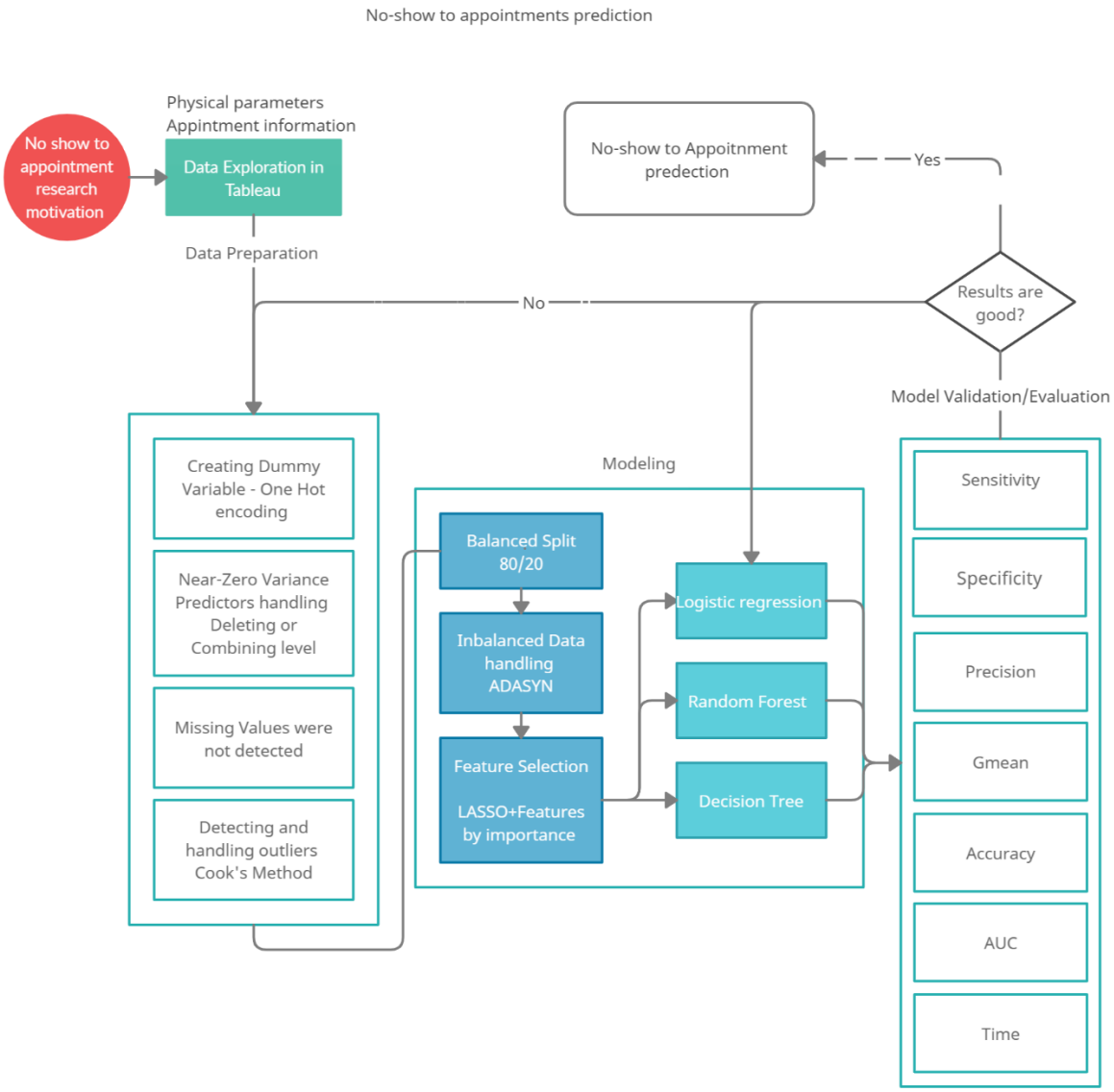
Model	Balancing Method	Sensitivity	Specificity	Precision	Gmean	Accuracy	AUC
GLM	DBSMOTE	0.568	0.692	0.277	0.635	0.673	0.586
Random Forest	DBSMOTE	0.508	0.774	0.514	0.627	0.728	0.6019
Decision Tree	DBSMOTE	0.317	0.894	0.894	0.532	0.708	0.606
GLM	ADASYN	0.6516	0.7279	0.335	0.6886	0.7146	0.621
Random Forest	ADASYN	0.643	0.692	0.9021	0.665	0.684	0.604
Decision Tree	ADASYN	0.301	0.904	0.904	0.5218	0.674	0.603

Moreover, the model run time was measured in order to evaluate GML regression and Random Forest running time. As it can be seen in the table below, the time GLM required was around 56 times lower compared to time that was required to run Random Forest. GLM with ADASYN balancing method gave the best results for predicting show ups for clinic Appointment.

Model	Elapsed (real time)
GLM	0.99
Random Forest	56.42

Overall, the methodology framework was adjusted in order to receive highest results possible.

The graph below summarizes all the phases of the methodology.



Discussion

Answering research questions: *Is it possible to predict no-show to appointment by the use of machine learning techniques? What are the important features in order to predict the no-show to appointment? What is most efficient strategy outline to increase the performance of no-show to appointments prediction?*

The exploratory data analysis showed that According to research papers done prior to this one, the dataset used for analysis in this paper did not have sufficient variables for no-show rate prediction. Prior research highlighted that lead time variable that was described as the time difference between making an appointment and actual appointment was a variable of significant importance for no-show prediction. Since the dataset used for this research did not include information on neither of these two variables, the prediction model lacked accuracy. Moreover, there was no available information about location of the clinics that provided data for this research, which would have been helpful in identifying no-show rate, since extra information about average salary could have been added into research. Taking all these factors into account by modifying the model and trying different methods on all stages of data preparation and modelling the accuracy of 71,5% was achieved by applying methods outlined in methodology section of this research. It can clearly be seen in the results section that by using Balanced Split for dividing dataset into train and test, ADASYN balancing method, ranking variables by importance and Lasso for feature selection led to the best result it was possible to achieve withing this research.

The most important features that might assist in identifying if a patient will show up for a scheduled appointment are SMS received, waiting time, gender, and months. However, it is important to notice that the month data in used dataset is not full, it only presents period for 3 month including April, May, and June. Therefore, it is insufficient variable, and it would be beneficial to collect data around the whole year to understand if month variable is as significant as the model shows now. There are a number of suggestions that can be made according to results received from this research. Firstly, sending notifications to patients about upcoming appoint is important and failing to notify them before their timeslot in a schedule might lead to lower operation efficiency in the hospital. Secondly, the decision to overbook time slots for different patients is not the most effective strategy that can be used to prevent high no-show rate and decrease costs. It would be beneficial to introduce a penalty for patients that are late for the appointment or patients who do not show up. Moreover, according to the data exploration analysis, hospital personal awaited for patient who did not show up 15 minutes on average, while they still had to wait for 8 minutes on average for patients who did show up for appointment. This waiting time for hospitals leads to inefficiency and dissatisfaction with services from people who show up for appointments in time. Hospitals should resolve this issue by implementing new

policy rules on defining the waiting time and penalty for the wasted time. It is scientifically proven that people with monetary incentives have higher probability of making an action, comparing to those who do not fear monetary penalty at all.

Overall, investing money into services that would remind a patient of an appointment, which could be an automated service or an extra employee who would be able to assist in calling to patients, would decrease no-show rate considerably as well as improving policies inside the hospital and introducing monetary penalties for late show up and no-show up cases.

For further research, I would recommend acquiring dataset with lead time information as well as neighborhood information on hospitals for more precise results. In addition, collecting mode data om occupancy of research participants might be helpful in identifying the show-up rate. In addition, since all the problems with inefficiency arise due to patients being late or not showing up at all, it would be interesting to change the direction of research and examine how the waiting time (of the hospital stuff) can be improved. I believe it can be done by making waiting time a target variable while using a dataset with features that identify the location of the hospital visited, the lead time, occupation, number of children and other variables that might potentially have a high effect on the result of no-show rate. Finally, the last suggestion I would give is exploring this problem only in hospitals with relatively high no-show rate and collect data by creating questionnaires and evaluate on how people are satisfied with their services, because after unpleasant experience people might make an appointment just in case for follow up and then find a hospital with higher quality service.

References

Alshaya, S., McCarren, A., & Al-Rasheed, A. (2019, December). Predicting No-show Medical Appointments Using Machine Learning. In *International Conference on Computing* (pp. 211-223). Springer, Cham.

Dantas, L. F., Hamacher, S., Oliveira, F. L. C., Barbosa, S. D., & Viegas, F. (2019). Predicting patient no-show behavior: a study in a bariatric clinic. *Obesity surgery*, 29(1), 40-47.

Ferro, D. B., Brailsford, S., Bravo, C., & Smith, H. (2020). Improving healthcare access management by predicting patient no-show behaviour. *Decision Support Systems*, 138, 113398.

Goffman, R. M., Harris, S. L., May, J. H., Milicevic, A. S., Monte, R. J., Myaskovsky, L., ... & Vargas, D. L. (2017). Modeling patient no-show history and predicting future outpatient appointment behavior in the veterans health administration. *Military medicine*, *182*(5-6), e1708-e1714.

Harris, S., & Samorani, M. (2020). On Selecting a Probabilistic Classifier for Appointment No-show Prediction. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3631887

Huang, Y., & Hanauer, D. A. (2014). Patient no-show predictive model development using multiple data sources for an effective overbooking approach. *Applied clinical informatics*, *5*(3), 836.

Lenzi, H., Ben, Â. J., & Stein, A. T. (2019). Development and validation of a patient no-show predictive model at a primary care setting in Southern Brazil. *Plos one*, 14(4), e0214869.

Parikh, A., Gupta, K., Wilson, A. C., Fields, K., Cosgrove, N. M., & Kostis, J. B. (2010). The effectiveness of outpatient appointment reminder systems in reducing no-show rates. *The American journal of medicine*, *123*(6), 542-548.