Supervised learning for classification of clinic patients

Varvara Yakovleva

176984

Introduction

For the proper work of hospitals and other health facilities, it is important to know whether a person that made an appointment will show up or not. It is crucial because there could be people who need this help, but did not manage to make an appointment. Also, doctors waste their valuable work time. Fortunately, now there is a lot of available information to analyse and technologies, such as machine learning techniques, that could help to solve the problem. I want to find optimal algorithm to predict someone to no-show an appointment or at least to find characteristics that affect it most.

To do this, I will use a data set from kaggle.com that consists of 110527 medical appointments of the public healthcare of the capital city of Espirito Santo State - Vitoria - Brazil and 14 variables of each: Patient ID, Appointment ID, Gender, Scheduled Day, Appointment Day, Age, Neighbourhood, Scholarship, Hipertension, Diabetes, Alcoholism, Handcap, SMS received, No-show.

To select the best model, I will use Recall as a measure of performance because for the clinic it is more important to maximise True Positive and minimise False Negative. Model should show when a person does not come. The worst mistake is when a model tells that a person will come when he/she does not because it leads to money loss for the clinic.

Methodology

To begin with, I would like to analyse the data. Among the variables I chose Gender, Quantity of days between Scheduled Day and Appointment Day, Age, Neighbourhood, Scholarship, Hipertension, Diabetes, Alcoholism, Handcap, SMS received, No-show as the most considerable ones. Patient ID and Appointment ID are unique identification values that does not separate information. Out of Scheduled Day and Appointment Day I created a characteristic that could be more representative, quantity of days between these dates. I thought that maybe people forget about appointments. Other original variables I classified as relevant for the problem.

Let's consider each variable in more detail:

- Gender: 0 represents a female patient, 1 — a male patient. It is a categorical variable.
- Days between Scheduled Day and Appointment Day: It is a quantitative variable.
- Age: tells the age of a patient. It is a quantitative variable. The maximum age is 115 years.
- Neighbourhood: there are 81 neighbourhoods where patients live. It is a categorical variable.

- Scholarship: 1 tells that a patient has a Scholarship, 0 — does not. It is a categorical variable.

- Hipertension: 1 tells that a patient has Hipertension, 0 — does not. It is a categorical variable.

- Diabetes: 1 tells that a patient has Diabetes, 0 — does not. It is a categorical variable.

- Alcoholism: 1 tells that a patient has Alcoholism, 0 — does not. It is a categorical variable.

- Handcap: the total amount of handcaps a person presents. It is a quantitative variable. The minimum value is 0. The maximum value is 4.

- SMS received: 1 tells that a patient received SMS from the clinic, 0 — did not. It is a categorical variable.

- No-show: 1 tells that a patient did not come, 0 — came. It is a categorical variable. The variable that we want to predict.

To make a prediction, Decision Tree, Random Forest and K-Nearest Neighbours were used. As for other algorithms that I do not apply, SVM, for example, was too long to run because there is too much data for it.

Before implementing models, I divided data into train set (75% of all data) and test set (25%). Train set had 82895 observations. Test set had 27632 observations.

Decision Tree

I chose Decision Tree as one of the models applied because it tells how much each variable contributes to the classification. Then it could be possible to reduce the dimensionality by ignoring variables that do not contribute information. To apply Decision Tree, it is not necessary to scale the data because the algorithm does it itself.

I got Recall = 0.842. Then I compared feature importances to eliminate variables with low ones.

```
Feature ranking:
1. Days between Scheduled Day Appointment Day (0.305846)
2. Age (0.282394)
3. Neighbourhood (0.272974)
4. Gender (0.049996)
5. Hipertension(0.025142)
6. Scholarship (0.019346)
7. SMS_received (0.016835)
8. Diabetes (0.011146)
9. Handcap (0.008636)
10. Alcoholism (0.007685)
```

I tried to run the model again with only Days between Scheduled Day and Appointment Day, Age and Neighbourhood as attributes. New Recall was 0.843>0.842. So, elimination increased the performance.

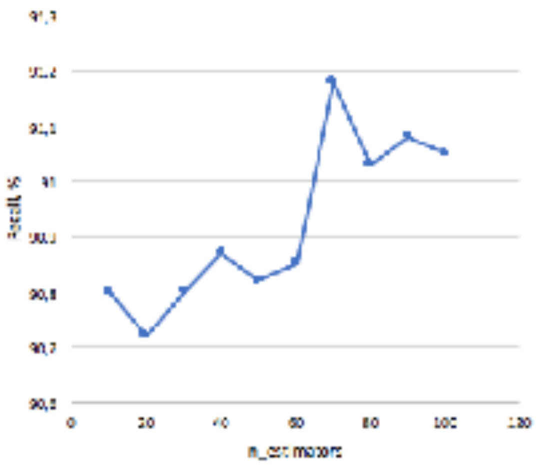
Random Forest

As we know, Random Forest helps a bit with the curse of dimensionality and increases diversity. That is why I decided to use it as well. To apply Decision Tree, it is not necessary to scale the data because the algorithm does it itself.

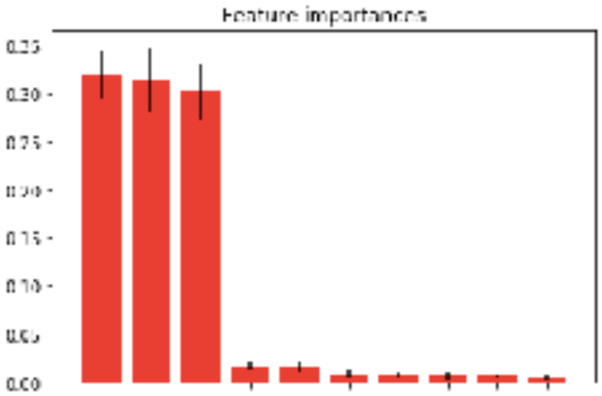
To apply Random Forest, we need to choose number of trees in the forest. I used cross-validation to do it. I was choosing between 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 trees. As long as recall is a measure of performance for me, I compared Recall to choose the optimal parameter.

```
Cross Validation Score for n_estimators =
70: 76.85%
[[20143  1950]
 [ 4331  1208]]
20143 1950
Recall: 91.17%
```

As you can see, n_estimators = 70 has the highest Recall, so I used it. Recall was 0.911. Then I compared feature importances to eliminate variables with low ones.



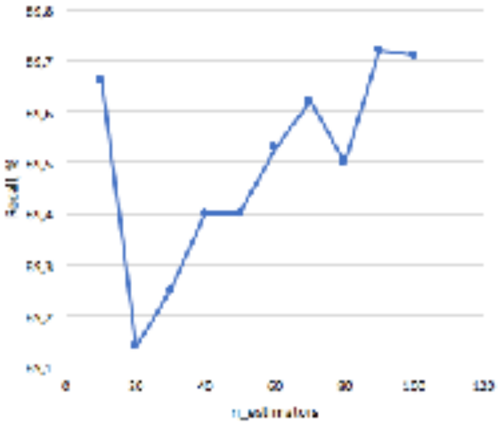
```
Feature ranking:
1. Days between Scheduled Day
Appointment Day (0.318428)
2. Age (0.314168)
3. Neighbourhood (0.301889)
4. Gender (0.017172)
5. SMS_received (0.014912)
6. Hipertension (0.008813)
7. Diabetes (0.006788)
8. Scholarship (0.006393)
9. Handcap (0.006300)
10. Alcoholism (0.005136)
```



Then I tried to run the model again with only Days between Scheduled Day and Appointment Day, Age and Neighbourhood as attributes. I used cross-validation:

```
Cross Validation Score for n_estimators =
90: 76.02%
[[19772  2188]
 [ 4254  1418]]
Recall: 90.04%
```

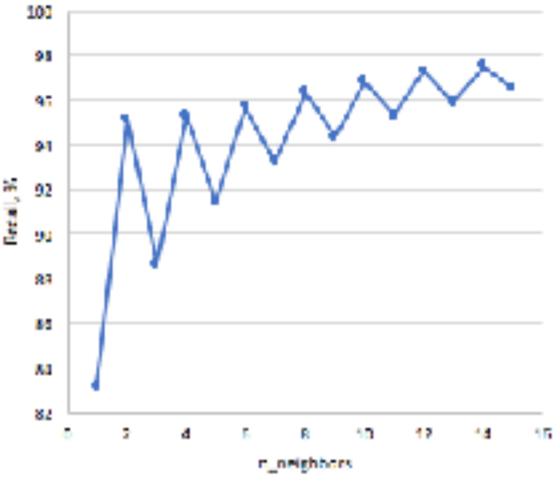
n_estimators = 90 has highest Recall, so I chose it. Finally, I got Recall = =0.912>0.911. So, elimination of irrelevant attributes increase the performance of the model.



K-Nearest Neighbours

K-Nearest Neighbours is a relatively simple algorithm, so I chose it to compare the results with the other algorithms. Before implementing the model I scaled data.

For this algorithm it was important to choose the quantity of neighbours properly. I used cross-validation for this purpose. I was choosing from numbers between 1 and 15. Again, I compared Recall to choose the optimal parameter.



```
Cross Validation Score for n_neighbors =
=14: 79.34%
[[21561   532]
 [ 5160   379]]
Recall: 97.59%
```

As you can see, n_neighbors = 14 has the highest Recall, so I used it.

Results

Three different classification algorithms were applied in order to find the superior model to know whether the person will show up for the doctor appointment or not. The performance measures with the test set were the following:

Recall of Decision Tree = 0.843

Recall of Random Forest = 0.912

Recall of K-Nearest Neighbours = 0.976

As we can see, the performance of K-Nearest Neighbours is the best among the models.

Conclusions

The objective of the project was to find the best classification model for a clinic to predict whether a patient will come to the appointment or not. The best classification model was K-Nearest Neighbours with n_neighbors = 14 because it classified people who do not come better than the other models.

Moreover, it is concluded by Random Forest that the great part of the performance of a model comes from Days between Scheduled Day and Appointment Day, Age and Neighbourhood as attributes. So, if a clinic wants to quickly decide whether a person will come or not, it can use a rule of thumb with these three characteristics.

As future work, I suggest trying different (more advanced) algorithms, such as building a classifier out of divers algorithms. Also, I guess it could be helpful to think of other attributes that could be important for output, for example, level of education or a information about whether a person has children and their age.

Bibliography

https://www.kaggle.com/joniarroba/noshowappointments