Hospital Visits

•••

Team 3 CopyPaste

Gina-Maria Tanja Färber 77211826704
Muhammad Raza 77211990025
Ligia Vergara 77212002271

Content

1. Problem and Goal Definition
2. Data Understanding
   a. Data Quality
   b. Exploratory Data Analysis
3. Data Cleaning and Basic Preprocessing
4. Advanced Preprocessing
5. Model Building and Fine Tuning
6. Model Evaluation
7. Business Recommendations
8. Resources and References

Problem Overview

The dataset contains a list of patients with different attributes that indicate the attendance or absence from a specific appointment at the associate hospital.

Objective

Develop and train a machine learning model that predicts if a patient will miss a future appointment.
With this prediction we are aiming to facilitate intervention strategies to reduce no-show events or effectively re assign the available appointment dates

Data Understanding

Target

df. head → 5 first records of the data

	PatientId	AppointmentID	Sex	ScheduledDate	AppointmentDate	Age	Community	SocialWelfare	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No-show
0	4.738527e+13	5387604	F	2016-02-24T07:53:17Z	2016-05-13T00:00:00Z	NaN	RESISTÊNCIA	no	no	no	no	no	no	No
1	6.557495e+13	5655266	M	2016-05-03T16:29:14Z	2016-05-12T00:00:00Z	4.0	NaN	NaN	NaN	no	no	no	no	No
2	1.265473e+11	5745855	F	2016-05-30T12:54:18Z	2016-05-30T00:00:00Z	19.0	JARDIM DA PENHA	no	no	no	no	no	no	No
3	2.681769e+13	5700247	F	2016-05-16T09:15:51Z	2016-05-16T00:00:00Z	55.0	JESUS DE NAZARETH	no	yes	no	no	no	no	No
4	7.813565e+13	5656211	F	2016-05-04T07:46:23Z	2016-05-04T00:00:00Z	0.0	ITARARÉ	NaN	no	no	no	no	no	No

- Appointment level granularity and contains detail of each appointment and patient.
- It has 14 columns of which one will be our target variable: No-show.
- Information about appointment date, the patients' health and location details. A column also shows if a patient received an SMS before their appointment.

```
Duplicate rows: 0
Duplicate appointments: 0
```

No duplicates rows or duplicate appointment

Only unique IDs

Data Understanding

Missing Values

	Missing Count	Percentage Missing
PatientId	0	0.00
AppointmentID	0	0.00
Sex	0	0.00
ScheduledDate	0	0.00
AppointmentDate	0	0.00
Age	8807	9.96
Community	10713	12.12
SocialWelfare	12519	14.16
Hipertension	8021	9.07
Diabetes	0	0.00
Alcoholism	14889	16.84
Handcap	0	0.00
SMS_received	0	0.00
No-show	0	0.00

Age, Community, SocialWelfare, Hipertension, and Alcoholism have significant null values.

No NULLs in the target variable

Data type checks

```
PatientId          float64
AppointmentID        int64
Sex                 object
ScheduledDate       object
AppointmentDate     object
Age                float64
Community           object
SocialWelfare       object
Hipertension        object
Diabetes            object
Alcoholism          object
Handcap             object
SMS_received        object
No-show             object
dtype: object
```

- PatientID and Age are usually whole numbers (int64)
- Dates are timestamp not objects

Inconsistent Values

	Handcap	0
0	2	139
1	3	11
2	4	3
3	no	86626
4	yes	1642

```
ScheduledDate > AppointmentDate
False     88417
True          4
Name: count, dtype: int64
```

- Handcap is a binary attribute (yes/no)
- Appointment date should be after schedule date

Data Understanding

EDA

- Female to Male ratio is **65:35**
- **1 in 5 appointments** is missed on average (for both gender).
- There is an even distribution of appointments missed in the various age groups. This tends to change after the age of 70, where appointments are missed less.
  - older people cannot afford to miss appointments due to more serious health issues and they have more time.
- It doesn't seem like that a specific community is missing more appointments than others.

Data Preprocessing

New Feature

Derived Features: time between appointment

	PatientId	AppointmentID	Sex	ScheduledDate	AppointmentDate	Age	Community	SocialWelfare	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No-show	time_bw_schedule_appointment
0	4.738527e+13	5387604	F	2016-02-24	2016-05-13	NaN	RESISTÊNCIA	no	no	no	no	no	no	No	79.0
1	6.557495e+13	5655266	M	2016-05-03	2016-05-12	4.0	NaN	NaN	NaN	no	no	no	no	No	9.0
2	1.265473e+11	5745855	F	2016-05-30	2016-05-30	19.0	JARDIM DA PENHA	no	no	no	no	no	no	No	0.0
3	2.681769e+13	5700247	F	2016-05-16	2016-05-16	55.0	JESUS DE NAZARETH	no	yes	no	no	no	no	No	0.0
4	7.813565e+13	5656211	F	2016-05-04	2016-05-04	0.0	ITARARÉ	NaN	no	no	no	no	no	No	0.0

Handling Missing Data - Part 1

Using PatientID data to extrapolate for the same patient with missing data as Patients ID are duplicated with different Appointments

Train-Test Split

Train-Test Split by Group to avoid data leakage

Data Preprocessing

Handling Missing Data - Complete info

Using PatientID data to extrapolate for the same patient with missing data

Simple Encoder with most frequent value for categorical attributes and median for numerical value “age” as it is skewed

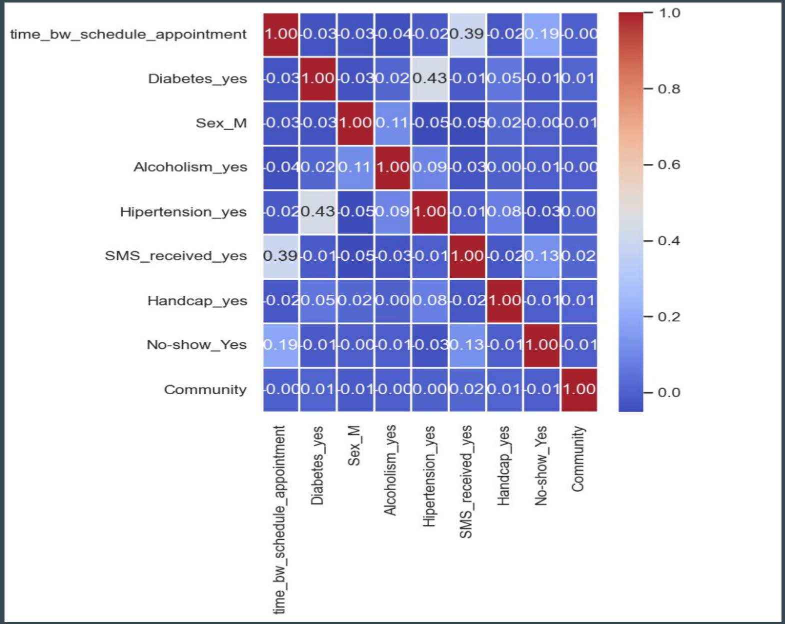
Handling Categorical Values

#	Column	Non-Null Count	Dtype
0	PatientId	69168 non-null	float64
1	AppointmentID	69168 non-null	int64
2	Sex	69168 non-null	object
3	ScheduledDate	69168 non-null	object
4	AppointmentDate	69168 non-null	object
5	Age	69168 non-null	float64
6	Community	69168 non-null	object
7	SocialWelfare	69168 non-null	object
8	Hipertension	69168 non-null	object
9	Diabetes	69168 non-null	object
10	Alcoholism	69168 non-null	object
11	Handcap	69168 non-null	object
12	SMS_received	69168 non-null	object
13	No-show	69168 non-null	object
14	time_bw_schedule_appointment	69168 non-null	float64

The majority of categorical values are binary: sex, SocialWelfare, Hipertension, Diabetes, Alcoholism, handcap, sms_received and they were handle with OHE
Community was also handle with OHE.
Categorical attributes does not have an ordinal relationship therefore no other encoder was needed.

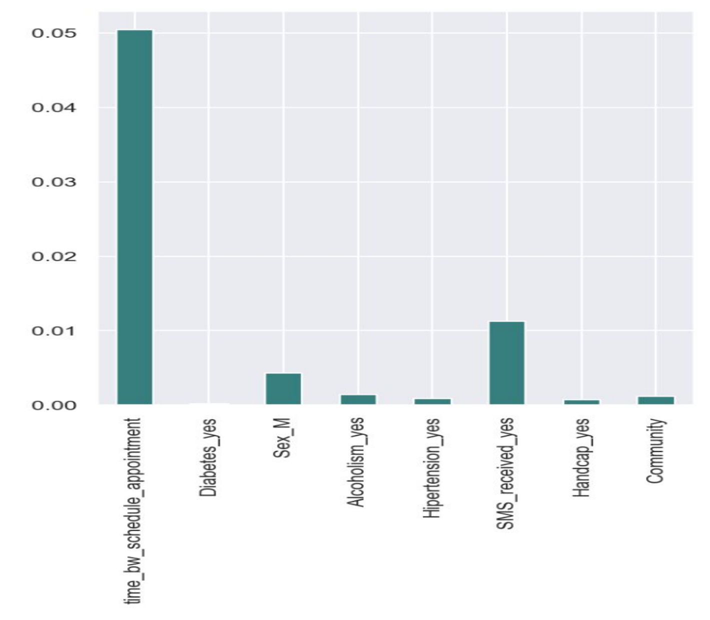
Data Preprocessing: Feature Selection

Correlation Matrix



- Variables do not show a high correlation with the No show targe
- Time_bw_schedule_appointment and SMS_received show correlation
- Hypertension and Diabetes show correlation

Information Gain



- Time bw_schedule appointment seems to be important for the model.

Model Building

Since it is a classification problem, we begin with a simple **Decision Tree**:

High interpretability and easy to understand

Keeping in mind it is a weak classifier...we see a low result for predicting no-shows

	precision	recall	f1-score	support
No	0.82	0.85	0.84	21351
Yes	0.32	0.27	0.29	5414
accuracy			0.73	26765
macro avg	0.57	0.56	0.56	26765
weighted avg	0.72	0.73	0.73	26765

Model Building: An advanced Tree Algorithm

Applying **Random Forest** to leverage the concept of bagging and combining the predictions of many trees

However, we only see **similar** results compared to the decision trees...

maybe sometimes less is indeed more?

	precision	recall	f1-score	support
No	0.82	0.90	0.85	21351
Yes	0.33	0.21	0.25	5414
accuracy			0.76	26765
macro avg	0.58	0.55	0.55	26765
weighted avg	0.72	0.76	0.73	26765

Boosting: Testing out alternative strategies

Since bagging did not provide more support, we try to see if boosting can get us there.

We sequentially train the 'weak' learners or trees and then build a final model that has 'learned' from the previous learners' mistakes

We must be careful of overfitting still...but here comes **XGBoost**! We predict no-shows far more accurately:

	precision	recall	f1-score	support
No	0.82	0.90	0.85	21351
Yes	0.33	0.21	0.25	5414
accuracy			0.76	26765
macro avg	0.58	0.55	0.55	26765
weighted avg	0.72	0.76	0.73	26765

	precision	recall	f1-score	support
0	0.92	0.53	0.67	21351
1	0.31	0.82	0.44	5414
accuracy			0.59	26765
macro avg	0.61	0.67	0.56	26765
weighted avg	0.80	0.59	0.63	26765

What are the trees not telling us?

We wanted to add variety to the underlying approach of our ML models and after some research we found that **Logistic Regression** can be a great complement to our previous approaches.

Linear clarity after ‘random’ walks through the forest gives us a fresh perspective:

	precision	recall	f1-score	support
0	0.92	0.53	0.67	21351
1	0.31	0.82	0.44	5414
accuracy			0.59	26765
macro avg	0.61	0.67	0.56	26765
weighted avg	0.80	0.59	0.63	26765

	precision	recall	f1-score	support
0	0.86	0.73	0.79	21351
1	0.33	0.53	0.41	5414
accuracy			0.69	26765
macro avg	0.60	0.63	0.60	26765
weighted avg	0.75	0.69	0.71	26765

May the best prediction win!

Instead of putting all our eggs in one basket, we see if a **(soft) voting classifier** helps us get even higher prediction power...and...we do!

We get the **highest F1 score** which is what we wanted to achieve.

	precision	recall	f1-score	support
0	0.86	0.74	0.80	21351
1	0.35	0.54	0.42	5414
accuracy			0.70	26765
macro avg	0.60	0.64	0.61	26765
weighted avg	0.76	0.70	0.72	26765

Jumping into unknown waters

How does our model do when we test on never seen before data? We put up a fight!

#	Team	Members	Score	Entries	Last	Join
1	Lolgarithm		0.62267	91	17h	
2	Numbers Nerds		0.62127	57	5h	
3	Power Rangers		0.61609	52	1d	
4	**CopyPaste**		$\boxed{0.61557}$	47	36s	
5	Data Detectives		0.60657	31	9d	
6	RPA_squad		0.59541	25	26m	
7	The Predictive Pioneers		0.57492	39	3d	
8	DataVoyagers		0.56499	27	4m	

🙂 Your Best Entry!
Your submission scored 0.60986, which is not an improvement of your previous score. Keep trying!

Business Recommendations

- The hospital could use our predictions to make a decision on managing patient appointments and resources. Whether we focus on the false positive or negatives depends:

- If the current resources are being underutilized, then it might make sense to have some overbookings and look at the false positives as it **expects** some patients to **not show up** but they **do show up**.

- If the hospital resources are already over utilised, then we would want to look at the false negatives more as they tell us which ones actually show up from the ones not expecting to show up. For the hospital, this would mean that they want to reduce the number of people who show up but expected not to show up. This would reduce the strain on resources.

What more could we do to help the hospital?

We could gather more data on:

- What are the patients coming in for - so we understand if patients coming in for certain reasons are more likely to miss their appointments. This could help the hospital manage resources better by specific department.
    - We gather this data during the appointment call to avoid target leakage.

Resources

ChatGPT: used for code snippets and troubleshooting errors.

Scikit-learn documentation: https://scikit-learn.org/stable/

Data Science Class Resources