Data Science Project Hair Salon No-Show Protocol

Author(s): Daniel Vaks

Introduction

No-shows are a big problem for hair salons the same as airlines and medical facilities. Excessive no-shows increase costs and wait times for businesses and all other customers alike. A No-show prediction service would allow hair salons to select from a variety of treatment options at the time of the booking such as requiring a non-refundable deposit or scheduling the appointment at a different time, location or with a different service provider such that the potential no-show would have less business and customer experience impact.

Motivation

A small business may significantly benefit by applying simple machine learning techniques to solve daily troubles causing feasible costs. While the range of everyday troubles is wide, the problem of no-shows e.g. situations when a client does not come for an appointment is common for restaurants, hair and beauty salons, private dentist cabinets etc. Of course, each case would be unique so there shouldn't be a unique algorithm predicting no-shows for different types of small business at the same time. In this study, we aim to solve the no-show prediction problem for hair salon proposed on kaggle.com. A no-show prediction service would allow hair salons to select from a variety of treatment options at the time of the booking such as requiring a non-refundable deposit or scheduling the appointment at a different time, location or with a different service provider such that the potential no-show would have less business and customer experience impact.

Project aims

Apart from just predicting the no-shows with the highest score possible we will also concentrate on the explanation of the mechanisms causing a no-show. Therefore the aims are:

- Investigate the data
- Build a prediction model
- Identify the features that contribute the most
- Explain the mechanisms underlying the model

Methodology (Project design)

Data

The data contains *1952 cases and 22 features* containing information about the current and previous bookings if any. The task is to predict no show which is 1 if a client misses an appointment and 0 if a client shows up.The data was used with the permission of an actual hair salon in Toronto, Canada. It covers a time series from March to July of 2018. (according kaggle.com)

All the data was downloaded from Kaggle, no external data was added. Possible data source

that could enrich my study is the weather forecast for the day of service - to see if it has an effect

on no show prediction. I could not find a reliable source for this so I didn't add anything because the data set has not contained the actual date of appointment.

Features:

noshow -- Did the client no-show or execute an out-of-policy cancellation for this booking? (0 - no, 1 - yes) (dependent)-**outcome variable**

book_tod -- The booking time of day.

book_dow -- The booking day-of-week.

book_category -- The booked service category (COLOR or STYLE)

book_staff -- The staff member to provide the service.

last_category -- The client's last booked service category before the current booking or today whichever is greater.

last_staff -- The staff member who provided the client's last service before the current booking or today whichever is greater.

last_day_services --The number of services provided to the client on their last visit before the current booking or today whichever is greater.

last_receipt_tot -- The amount paid by the client on their last visit before the current booking or today whichever is greater.

last_dow -- The day-of-week of the client's last booking before before the current booking or today whichever is greater.

last_tod -- The time-of-day of the client's last booking before the current booking or today whichever is greater.

last_noshow -- Did the client no-show on their last booking before the current booking or today whichever is greater? (0 - no, 1 - yes)

last_prod_flag -- Did the client buy a retail product on their last booking before the current booking or today whichever is greater? (0 - no, 1 - yes)

last_cumrev -- The client's cumulative service revenue as of their last booking before the current booking or today whichever is greater.

last_cumbook -- The client's cumulative number of bookings as of their last booking before the current booking or today whichever is greater.

last_cumstyle -- The client's cumulative number of STYLE bookings as of their last booking before the current booking or today whichever is greater.

last_cumcolor -- The client's cumulative number of COLOR bookings as of their last booking before the current booking or today whichever is greater.

last_cumprod -- The client's cumulative number of bookings with retail product purchases as of their last booking before the current booking or today whichever is greater.

last_cumcancel --The client's cumulative number of appointment cancellations as of their last booking before the current booking or today whichever is greater.

last_cumnoshow -- The client's cumulative number of no-shows as of their last booking before the current booking or today whichever is greater.

recency -- The number of days since the client's last booking before the current booking or today whichever is greater.

Missing values:

Some of the variables contain more than 900 missing values which is a sustainable number considering we have 1952 observations. I suggest to exclude them. There is also a categorical variable (daytime of the booking) that contains 235 missings, here I suggest to substitute missings with unknown value.

book_tod - empty value was replaced by new categorical value “unknown”.

last_category -empty value was replaced by new categorical value “unknown”.

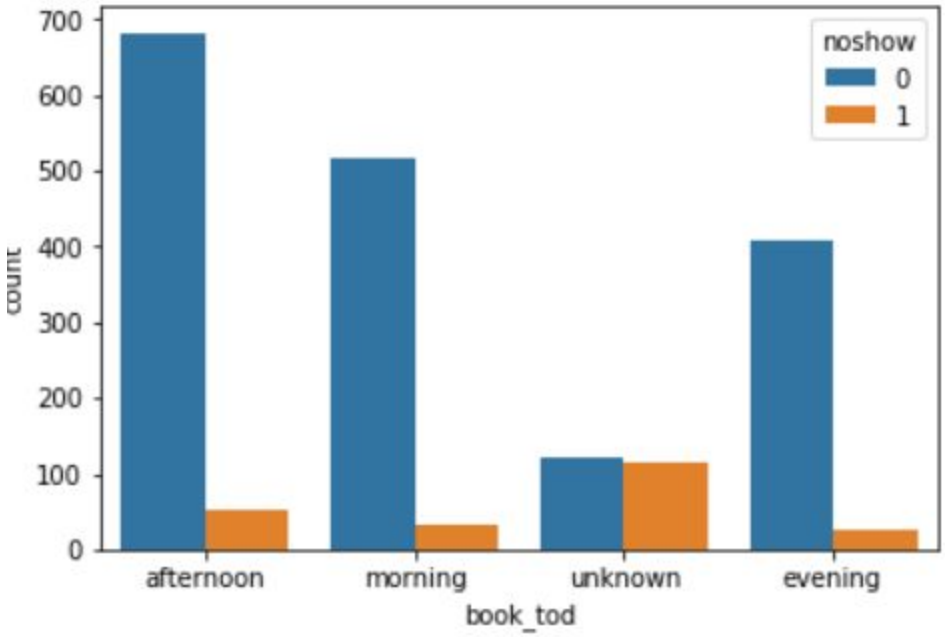
last_staff -empty value was replaced by new categorical value “unknown”.

last_dow -empty value was replaced by new categorical value “unknown”.

last_tod -empty value was replaced by new categorical value “unknown”.

As we have just filled book_tod with unknown values, let's check how many noshows happen when the day time of the appointment is unknown.

Well, almost half of the appointments with unknown time of the day end up with no-shows. Perhaps it would be an informative feature in the model.



Missing values:

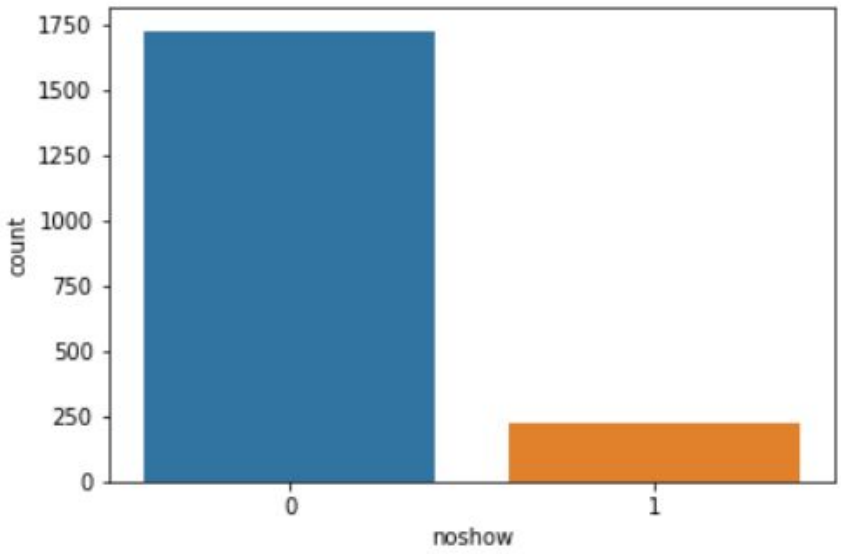
There are no missing values in the data set.

EDA:

Distribution of the dependent variable

In [3]:
```
sns.countplot(x='noshow', data=noshow)
```

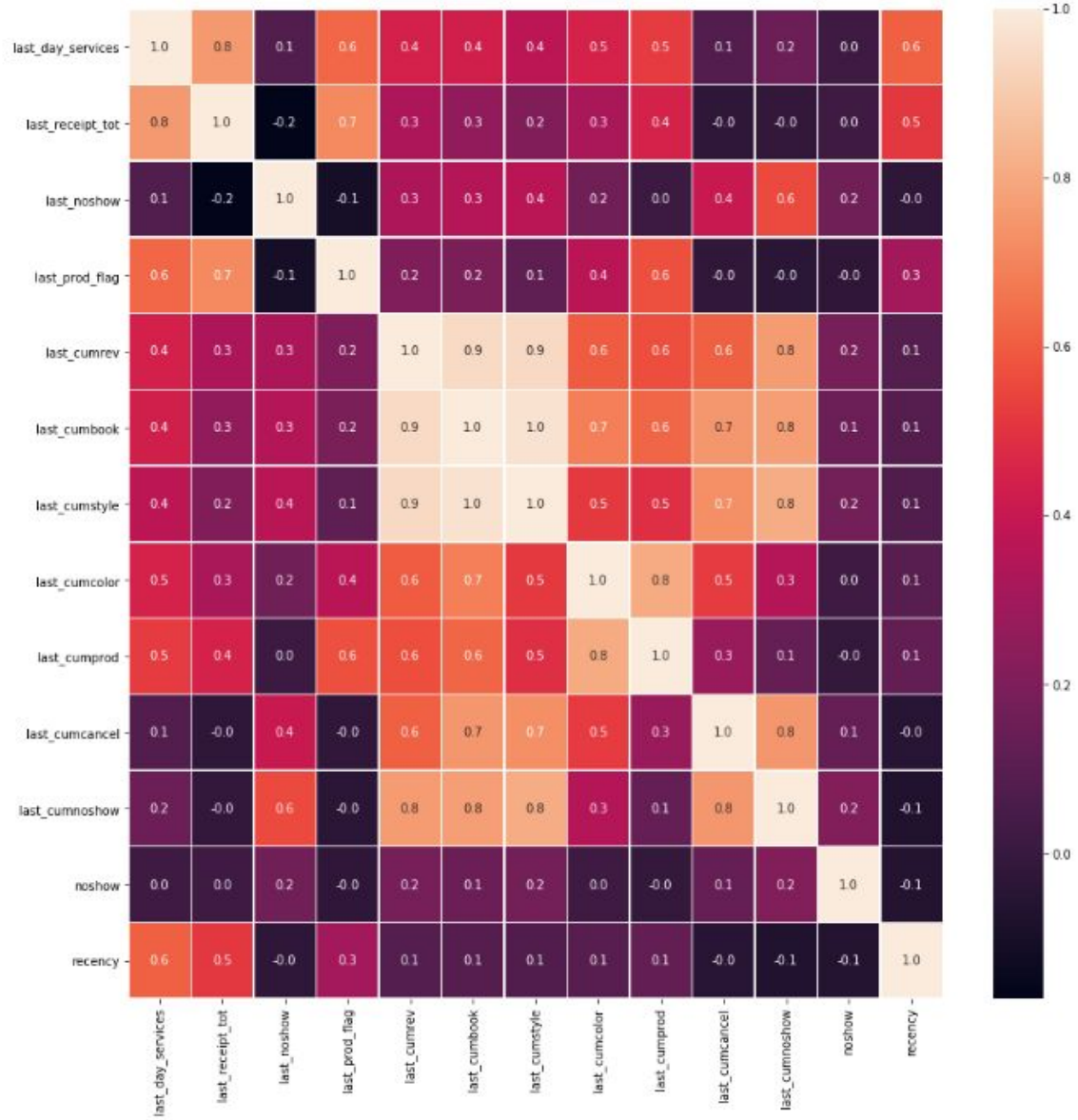
Out[3]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa33fceca90>
```



The data is imbalanced, there are more `0`s than `1`s in the dependent variable.

Correlation map

```
[4]: f, ax = plt.subplots(figsize = (15, 15))
     sns.heatmap(noshow.corr(), annot = True, linewidths = .5, fmt = '.1f', ax = ax)
     plt.show()
```



Highly correlated featrures are:

- **last_receipt_tot and last_day_services**
- **last_cumbook, last_cumstile, last_cumrev and last_cumnoshow**

Models

In this work I want to predict no show phenomena in a hair salon using the gathered data.
Due to this statement, I will use classification techniques .
After reading articles about good models to apply i chose to work with Gradient Boosting. This model is an ensemble of trees built one of the top of another. This model was chosen because it achieves the best accuracy and ROC-AUC scores among others I tried. The only drawback of this model is that the model is hard to interpret.
The other model I tried is the Random Forest classifier. Random forests create decision trees on randomly selected data samples, gets prediction from each tree and select the best solution by means of voting. It also provides a pretty good indicator of the feature importance. It was chosen because of essay implantation of the model , but the result was worse than Gradient Boosting
I will train the model only with numeric variables and dummy variables that I created from categorical variables.. I will train each model twice - without finetuning and after applying grid search for fine tuning.

Imbalanced data is a common problem in machine learning classification where there are a disproportionate ratio of observations in each class. Class imbalance can be found in many different areas including medical diagnosis, spam filtering, and fraud detection as we have seen in our data.
In my work I was trying to deal with problems by running regardless of the imbalance data and by applying oversampling.
Oversampling can be defined as adding more copies of the minority class. Oversampling can be a good choice when you don’t have a ton of data to work with like mine set of data.
We will use the resampling module from Scikit-Learn to randomly replicate samples from the minority class.

I wrote the function that will perform data split (default = 1/3), model evaluation with cross-validation if needed and print the ***Accuracy and ROC-AUC*** scores along with the feature importance of every run of one of the models i chose to run.The following functions will perform

models estimation with the default parameters and cross-validation on the whole dataset with 5 folds.
Also after that as part of fine tuning I performed grid search with cross-validation on the whole dataset to identify a proper number of estimators (number of trees built on top of each other) based on the maximization of accuracy score rather than roc_auc.

Deployment of your model

The model will be published and can be used by each haircut salon that wants to reduce no show phenomena.
The final user is an small business and not a customer so it can be given as a side program that can predict no shown using specific parameters.
From the hair salon owner's point of view the model can be used as a black box to predict and manage daily costs or he/she can gain some understanding of the mechanisms causing no shows by exploring different cases with LIME technique.

Conclusion

First, The topic was very interesting but much more complicated than I thought.There are so many variables ‘ models and techniques that affect prediction of no show phenomena that it is very hard to add them all.
A No-show prediction service would allow hair salons to select from a variety of treatment options at the time of the booking such as requiring a non-refundable deposit or scheduling the appointment at a different time, location or with a different service provider such that the potential no-show would have less business and customer experience impact.

Apart from just predicting the no-shows with the highest score possible I also tried to concentrate on the explanation of the mechanisms causing a no-show. Therefore I think I achieved my aims thay were:

- Investigate the data

- Build a prediction model
- Identify the features that contribute the most
- Explain the mechanisms underlying the model

Finally , the conclusion about the result of the work is that The Gradient Boosting model shows pretty good quality of the prediction but I a, sure that there are better models to fit the data or extend the data so the result could have been much more .

Model Report GradientBoostingClassifier after fine tuning
Test\Train split : 0.3333
Accuracy (Train) : 0.9508
Accuracy (Test) : 0.914
AUC Score (Train) : 0.949132
AUC Score (Test) : 0.830231

Model Report RandomForestClassifier after fine tuning
Test\Train split : 0.3333
Accuracy (Train) : 0.8885
Accuracy (Test) : 0.8786
AUC Score (Train) : 0.857657
AUC Score (Test) : 0.798320