Machine Learning - Project 2020 report

Predicting online purchasing intention

Florent HUYLENBROECK
Laurent BOSSART

May 24, 2020

Introduction

The goal of this project was to produce on the basis of various predictors the best model to predict if a session on an e-commerce website will lead to a purchase. This project took the form of a competition hosted on `https://www.kaggle.com`.
In groups of two, we had been given two dataset, one for training and one for testing, a short explaination of the different predictors and a way to measure our model's efficiency (the *F-1 score*).
We were able to submit five model per day.
This report will describe our reflexions on the subject and what submission we made.

The data sets

The predictors used in the two datasets are the following :

- *CategoryN* and *CategoryN_ Duration* with $N \in \{I, II, III\}$ represent the number of different pages belonging to a certain category visited by the user during that session and the time spent in that category.
  I = account management pages.
  II = website, communication and address information pages.
  III = product related page.

- *Bounce rate*, *Exit rate* and *Page value* are metrics provided by "Google Analytics" for each pages in e-commerce.
  Bounce rate is the number of single pages viewed by user (meaning the user exits the website on the same page he entered it, without navigating the site further).
  Exit rate tells from which page the users exit the website the most.
  Page value is the number of pages that a user visited before completing a transaction.

- *SpecialDay*, *Weekend* and *Month* all give information on the date when the session started.
  SpecialDay indicates the closeness of the site visiting time to a special day.
  Weekend tells if the session started during a saturday or a sunday. *Month* is the month of the visit date.

- *OS* and *Browser* are the exploitation system and the browser used by the user.

- *Region* is the geographic region where the user started his session.

- *TrafficType* is the traffic source from which the user entered the website.

- *VisitorType* indicates whether if the user is returning or new.

- *Transaction* indicates if a transaction has been completed. It is the value we will try to predict on our models.

Methodology

Brainstorming

Before anything else, we tried to think logically about the predictors. We ordered them from most to least important. We came up with a list that helped us build our first naive models.

Crossvalidation and useful functions

The first thing we did in *R* was to implement various function that would made our experimentation easier. We so implemented three functions :

- **submit_ prediticion**(*model*) that, given the parameter *model* being a anonymous function, returns a model, use it to predict our testing set's *Transaction* value and write that prediction next to the matching *Id*'s in a .csv file for submitting on Kaggle.
- **f1_ score**(*prediction*) that given a prediction over the training set, evaluates the $F1$-*score* of that prediction.
- **crossvalid**(*model, nrep, print*). This functions performs a 10-fold cross validation of the model *model* a number *nrep* of time over the training set and returns the mean $F1$-*score*. The argument *print* serves a debugging purpose.

Data pre-processing

We started our experiments by taking a deeper look a tthe data. We started analyzing every individual factors to see hwo it correlates to the *Transaction* value. We so generated one of the following plot for every predictor.

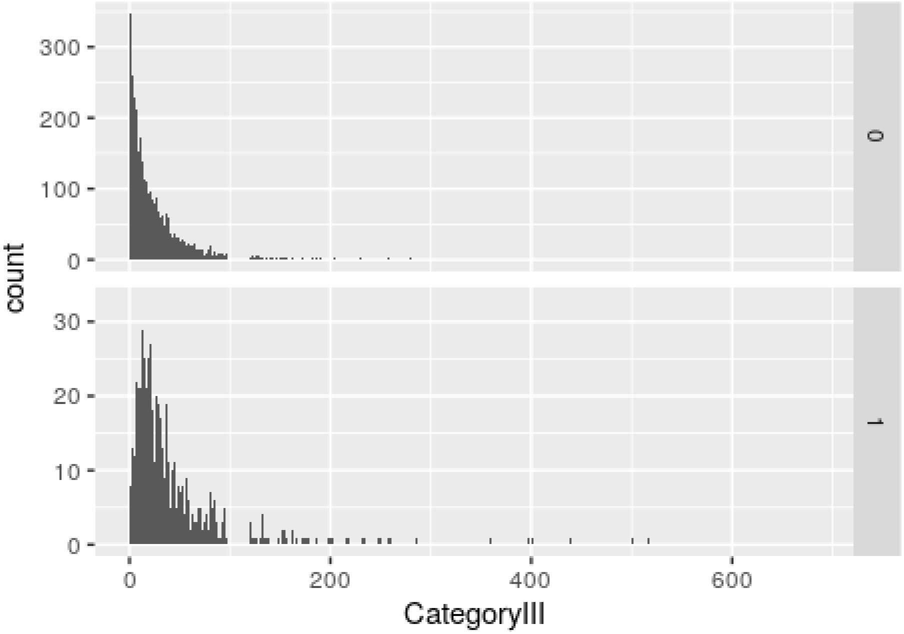

Figure 1: Plot generated to see CategoryIII impact on Transaction

This allowed us to get a clearer view of the predictor's impact on our models.
Using loops, we also computed, for every non-numeric variable, which values yielded the most completed transaction percentage. We then factored those predictor according to these results.
Finally we plotted the $Month$ predictor to see when were the users the most likely to complete a $Transaction$.

Linear regression

Our first models were built using linear regression. We tried to regress using our best predictors.
Then, since the number of predictors seemed managable, we tried to bruteforce every combination of predictors to see which would lead to the best model. This is not a good approach on itself, but we also kept track of the impact of every predictor on every model's $F1$-*score*. After two hours of computation, we obtained a good regression model and a third view on which predictor would be useful for further modelling.

Non-linear regressions

After trying linear regression, we tried to use higher order regression, limitting our higher order predictors to the 10 best predictors we found during our bruteforcing.
Since that resulted in a lowest number of predictor, we used the same bruteforcing function to yield a good non-linear regression using those predictors.

Decision tree

The last approach we took was to build a model based on decision tree using the *rpart* library.

Results and Discussion

Crossvalidation

Crossvalidation was a great tool to help us build strong models. Every model we built was automatically crossvalidated and our submissions were based on the $F1$-*score* given by our **crossvalidate** function.

Data pre-processing

Our plotting on the month variable allowed us to see that certain month were most likely to lead to a completed transaction than some others (November, October, September)

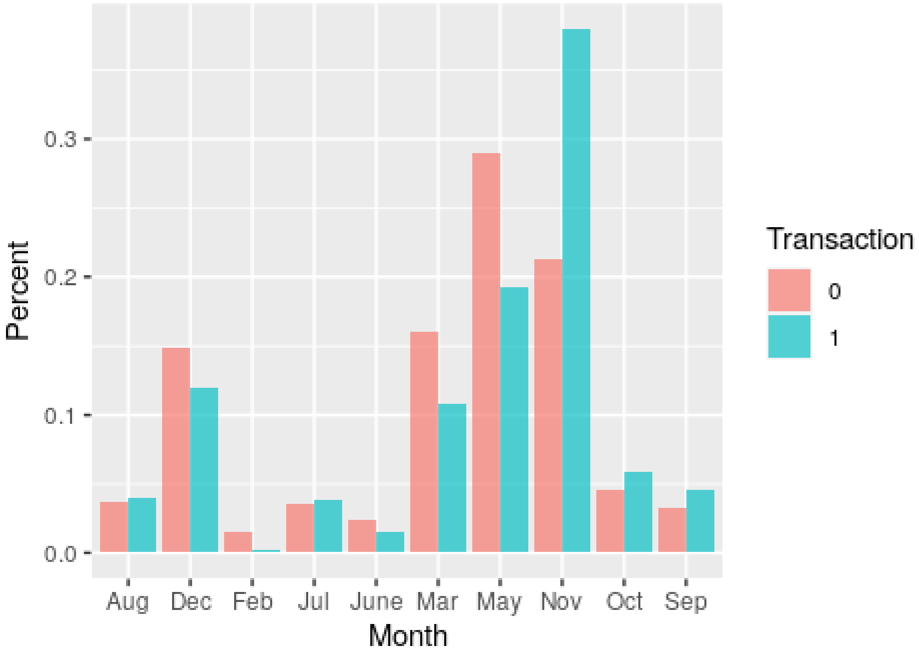

Figure 2: Percentage of transaction completed for each months

Linear regression

Our linear regression bruteforcing gave us decent first models. But most importantly, it told us that the most accuracy increasing predictors were $CategoryI$, $CategoryI\_Duration$, $CategoryII$, $CategoryII\_Duration$, $CategoryIII$, $CategoryIII\_Duration$, $Bounce\_rate$, $Exit\_Rate$, $Page\_value$ and $Month$. The exact result for this model got erased but it gave us our lowest score on the competition page.

Non-linear regressions

The results with higher order predictors were better than without. The best combination of predictors we found was :
$Transaction\,CategoryI + CategoryI^2 + CategoryI\_Duration + CategoryI\_Duration^2 + CategoryII^2 + CategoryII\_Duration + CategoryII\_Duration^2 + CategoryIII^2 + CategoryIII\_Duration^2 + Bounce\_Rate + Exit\_Rate + Exit\_Rate^2 + Page\_Value + Page\_Value^2 + Month$. Submitting this model yielded our second best score : 0.88278

Decision tree

Finally, we experimented a bit with decision trees. Using the *rpart* library and our factored data, we built the following model :

This model seemed accurate since it takes it's first decisions on factors that we already aknowledged as being important in the classification process. Submitting this model yielded our best score : 0.89568

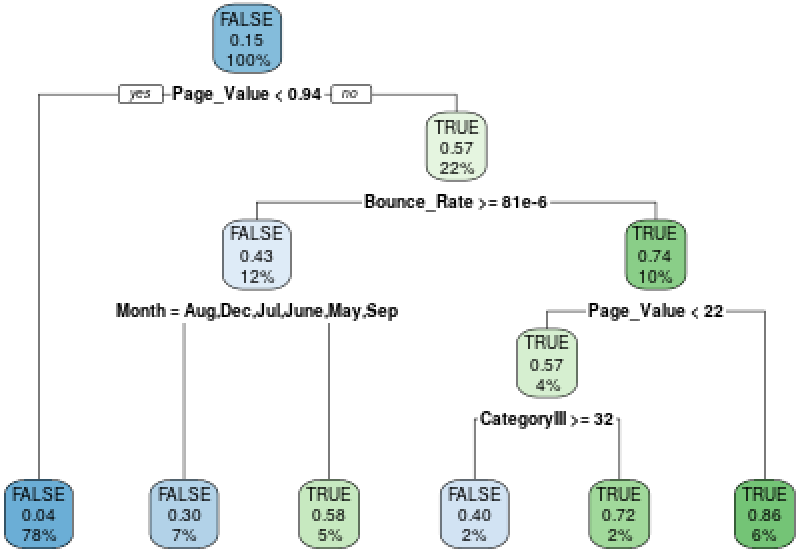

Figure 3: Our decision tree model

Conclusion

The best method we used was the decision tree. The model we built showed us that focusing on increasing the page value and the bounce rate was the best way to lead users to complete a transaction. Figure 2 also showed us that May, July, August, September and November were the months were an higher percentage of online shopper actually bought something online. Finally, a higher number of product related page visited by the users leads to more transaction. e-commerce website should empathize on making their product related pages apealing and easy to navigate in order to sell more.