Machine Learning Engineer Nanodegree

Capstone Proposal

Nathália Brunet
(nathaliabrunet@gmail.com)

August 20th, 2020

Customer Segmentation Report for Arvato Financial Solutions

Domain Background

Arvato is a business services company operating in different parts of the world, offering financial solutions in the most diverse forms, from payment processing to risk management activities, with development always focused on innovation. Arvato is owned by Bertelsmann, which is a media, services and education company operating in more than 50 countries and one of its priorities is to value solutions with an emphasis on creativity and entrepreneurship [1].

In this context, it is interesting to understand the customer segments that the company serves. For a company with a large customer base, the customer segmentation stage is very important. It aims to obtain more information (and really relevant information) from its customers, which can be used for a more personalized and targeted service, for example.

Problem Statement:

The goal is to perform a segmentation of customers, and with the knowledge obtained, identify whether a customer will respond to a specific marketing campaign or not. Thus, it is possible to increase the effectiveness of the campaign by directing it to an audience with a greater potential to " be reached" the campaign. In summary, the goal is to improve the performance of marketing campaigns.

Datasets and inputs:

The datasets were provided by Bertelsmann/Arvato in the context of the Udacity Machine Learning Engineer Nanodegree. It is a data that represents a real-life data science task.

The problem can also be found in the Kaggle challenge [2].

There are four data files associated with this project:

1. Udacity_AZDIAS_052018.csv: Demographics data for the general population of Germany; 891 211 persons (rows) x 366 features (columns).
2. Udacity_CUSTOMERS_052018.csv: Demographics data for customers of a mail-order company; 191 652 persons (rows) x 369 features (columns).
3. Udacity_MAILOUT_052018_TRAIN.csv: Demographics data for individuals who were targets of a marketing campaign; 42 982 persons (rows) x 367 (columns).
4. Udacity_MAILOUT_052018_TEST.csv: Demographics data for individuals who were targets of a marketing campaign; 42 833 persons (rows) x 366 (columns).

For more information about the columns depicted in the files, it is possible refer to two Excel spreadsheets the are provided too:

1. DIAS information levels - attributes2017.xlsx: is a top-level list of attributes and descriptions, organized by informational category
2. DIAS Attributes - Values 2017.xlsx: is a detailed mapping of data values for each feature in alphabetical order

---

Solution Statement:

The solution will be divided into two stages.

1. The first step is done to identify if the costumer are a potential customer or not. Through unsupervised learning, to segment customers into significant groups, identifying clusters / segments of the population in general that best correspond to the base of interest.

2. With the first stage completed, and the segments of the customers identified, supervised learning will be approached, with the objective of forecasting the individuals most likely to become customers. In this stage could be test some techniques that are good supervised learning, which may include Logistic Regression, XGBoost and LightGBM models, for example.

- Logistic Regression: it is a statistical method, in which the probability of the occurrence of an event is calculated to classify two classes, being one of the most used machine learning algorithms for this type of problem. It can be used for several types of classification problems, such as spam detection, disease prediction (such as diabetes and cancer), and one that I believe is very similar to the objective of our problem, quality classification problem and interest of visitors who browse the sites [3]. For example, a company wants to know if the visitor to the site has a genuine interest in the products offered or if they have entered the site for other reasons, with the classification of these visitors, you can do differentiated retargeting actions, present campaigns and personalized actions according to the profile or direct investments in an intelligent way, similar to our case, in which we want to classify whether it is a potential client or not.
- XGBoost: is a machine learning algorithm, based on a decision tree and that use a Gradient boosting structure. Is one of the most popular machine learning algorithm these days, and has been the most successful in Kaggle (an online

community of data scientists and machine learning practitioners that offer machine learning competitions). In addition, it has also been used, with sucessfully, in differents applications in the industry [4].

- LightGBM: is a fast gradient boosting framework, based on a decision tree algorithm. It is used for several types of machine learning problems, including classification problems. In [5], experiments showed that LightGBM can accelerates the training process of the conventional Gradient boosting decision tree by more than 20 times, and reaching almost the same accuracy.

As it is a proposal stage, for now there is no way to know which techniques will result in the best result. But during implementation, different approaches to evaluation will be tested.

Benchmark Model:

For this problem, the suggestions is Gradient Boosting Classifier of a similar dataset which has a performance of about 80%

Evaluation Metrics:

For the first stage of the solution (unsupervised methods), the amount of explained variance can be used as a metric. For the second stage (supervised methods), metrics such as precision, recall, ROC curve can be addressed. It is important to check how the data set classes are balanced.

Imbalanced class distribution in datasets occur when there are significantly less training examples in one class compared to another class. This means that the number of examples from the class of interest (minority in this case) is much less than the number of examples from the other class (majority). The class imbalance involves a series of difficulties in learning the model. When accuracy is used as a metric in datasets with imbalanced classes, we run the risk of obtaining misleading information about the model's performance [6].

The accuracy is the total number of correct predicts divided by the total number of predicts, so when we have imbalanced classes, it is possible that the accuracy is high because it is predicting a lot of the majority class. Suppose we want to solve the problem mentioned earlier, from a company that wants to know if the visitor to the site is really interested in the products offered or if entered the site for other reasons, dividing it between "good" users, who could buy or have the possibility of return to buy and "not good" users, those who would not be interested in the product and would not visit the site again. What can happen is that we have a much larger number of "not good" users than "good" users, so it can happen that the model just always responds "not good", leading to a high accuracy, so tells very little about the “good” class, that happens due to the class imbalance.

In our dataset, we can observe (Figure 1) that 98.76% (42.430) of the data refer to individuals who did not respond to the campaign, with only 1.24% (532) of the individuals who responded to the campaign, which means that the data training are very imbalanced, so

accuracy will not be an evaluation metric, because can gives inaccurate and misleading information about the classifier.

```
print('Count per class:\n', mailout_train['RESPONSE'].value_counts(normalize = False))

color='#00224e'
mailout_train['RESPONSE'].value_counts(normalize = True).iplot(kind='bar',
                                                               color= color,
                                                               title = 'Dataset imbalanced',
                                                               xTitle = 'RESPONSE',
                                                               yTitle = 'Percentual')
```

```
Count per class:
 0    42430
1      532
Name: RESPONSE, dtype: int64
```

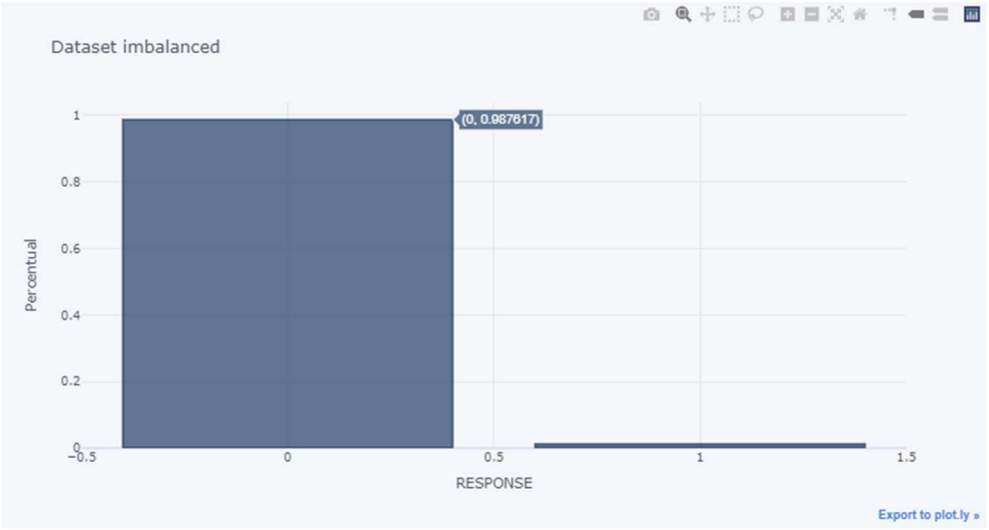

Figure 1 – Dataset Imbalanced

Confusion matrix, precision, recall and F1 Score are metrics that could be analyzed. Confusion matrix is a metric that could use when dealing with classification, this metric gives an interesting overview of how well a model is doing. More about each one [7]:

- Confusion Matrix: A breakdown of predictions into a table showing correct predictions (the diagonal) and the types of incorrect predictions made (what classes incorrect predictions were assigned).
- Precision: A measure of a classifiers exactness. Define how trustable is the result when the model answer that a point belongs to that class.
- Recall: A measure of a classifiers completeness. Expresses how well the model is able to detect that class.
- F1 Score (or F-score): A weighted average of precision and recall.

For better visualization purpose, many researchers adopt another performance measure, for example, the Receiving Operating Characteristic (ROC) curve. The idea is to estimate the performance of a binary classifier by varying the discrimination threshold [6].

Project Design:

The steps followed to implement the project are:

- Understanding the problem and first contact with the base: the first step is to understand the problem, what you want to solve, and the tools / data that will be

used. Again, it is important to check how the data set classes are balanced, it is a important point that will influence the solution of the project.

- Data cleaning: checking for missing data, incorrect values, etc.
- data visualization: creation of visualization to facilitate the identification of patterns, and understanding of the data.
- Resource engineering: Reduction of dimensions without losing much variance. One possible method for doing this is the PCA.
- Model selection: segmentation of data in clusters
- Model tuning: model optimization with hyper parameter adjustment
- Test and predict: test the selected model using the test data.

Links:

[1] www.bertelsmann.com

[2] https://www.kaggle.com/c/udacity-arvato-identify-customers

[3] https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python

[4] https://www.datageeks.com.br/xgboost/

[5] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree." Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 3149-3157.

[6] Ali, Aida & Shamsuddin, Siti Mariyam & Ralescu, Anca. (2015). Classification with class imbalance problem: A review. 7. 176-204.

[7] https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28
