E-COMMERCE COSTUMER SATISFACTION PREDICTING MODEL

First report for Machine Learning course project work
Academic year 2021/2022

Federico Cantarelli
Matricula #: 992964
Master in Management Engineering
Politecnico di Milano
federico.cantarelli@mail.polimi.it

Matteo Lorandi
Matricula #: 976145
Master in Management Engineering
Politecnico di Milano
matteo.lorandi@mail.polimi.it

Riccardo Righetti
Matricula #: 969167
Master in Management Engineering
Politecnico di Milano
riccardo.righetti@mail.polimi.it

Abstract

This is the first report for the project work of Machine Learning course held by professor Carlo Vercellis and professor Mauricio Soto Abel Gomez at Politecnico di Milano. This report aims to describe the steps in developing a model to predict customer satisfaction and to identify main drivers for customer satisfaction in order to gain competitive advantage. The dataset provided has 50,000 records and 19 features.

1. DATA OUTLOOK

Yojo.com is an e-commerce that rely on customer satisfaction to gain competitive advantage on the competitors. The objective of this project is to find a model to predict customer satisfaction based on both historical opinion of the customer and the characteristics of the products purchased.
The datasets are structured as follow:

- `model.csv` contains 50,000 records with the respective target variable;
- there are 19 features in the `model.csv` dataset, the last one is the target;
- `predictions.csv` contains 20,000 records without target variable and it will be relased for the next assignement.

The model selection process will be based on f1 score defined as follow:

$$f1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

#	Variable	Description
1	ID	Client ID
2	Gender	Client gender
3	Customer type	Client type: Premium or Standard
4	Age	Client age
5	Price	Order amount
6	New/Used	Used & New or used condition of the purchased product
7	Category	Product category
8	Product description accuracy	Level of satisfaction on product description
9	Manufacturer sustainability	Level of satisfaction on the manufacturing sustainability process
10	Packaging quality	Level of satisfaction on packaging
11	Additional options	Level of satisfaction on extra options
12	Reviews and ratings	Level of satisfaction on reviews and rating information
13	Integrity of packaging	Level of satisfaction on packaging state
14	Check-out procedure	Level of satisfaction on payment procedure
15	Relevance of related products	Level of satisfaction on related product suggestion
16	Costumer insurance	Level of satisfaction on insurance options
17	Shipping delay in days	Delay of shipping in days
18	Arrival delay in days	Arrival delay in days
19	Satisfaction	Target: Satisfied, Not Satisfied

Tab1: dataset structure

2. DATA CLEANING AND PREPROCESSING

As initial task, we analyzed the dataset, searching for possible anomalies and missing data. We discovered that there is a 7.736% of missing data for the costumers' age feature. Comparing these missing values with missingno package with other features, we discovered that there are no recognizable patterns in NULL values as we can see in figure 1. So, these are probably people who preferred to not specify their age during the survey. We decided to look more deeply into it trying to understand if and how to fill those missing data.

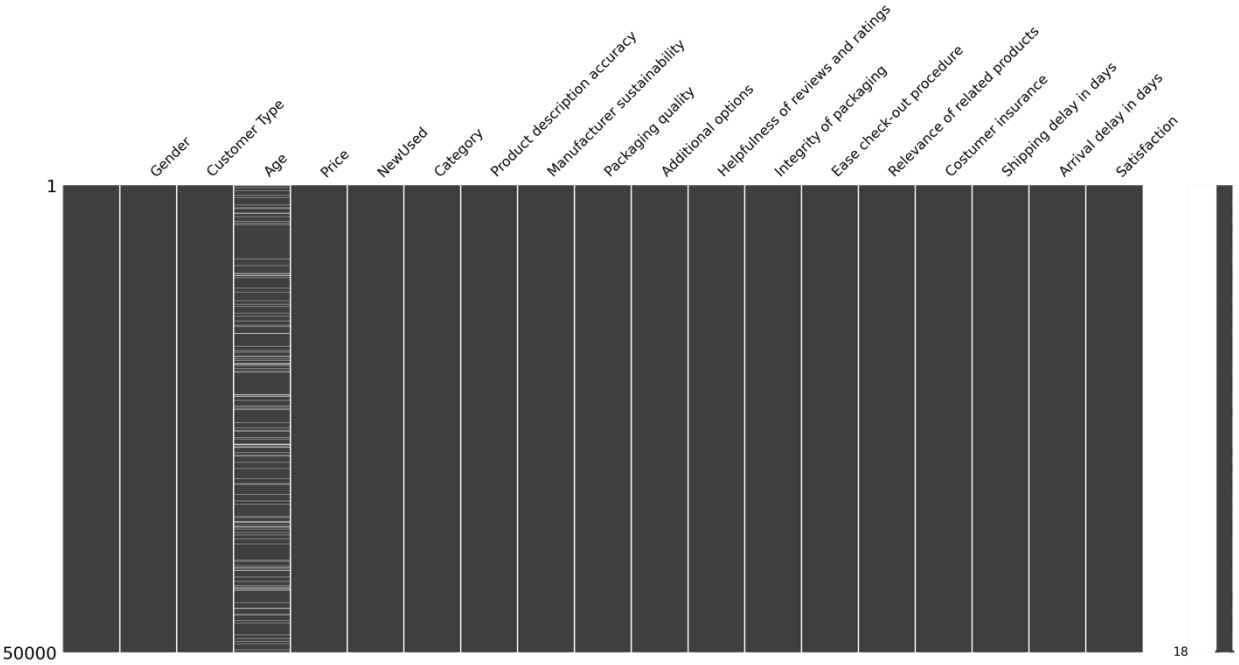


Figure 1: NULL matrix

Customer ID is unique by definition and so it has no correlation with target and the other features, thus it is useless in order to build a binary classification model. So, we decided to drop the ID column, even with the aim of reducing the computational effort needed to analyse the dataset. We found two different customers (one male and one female) with the same ID, undermining the relevance of these records, so we decided to eliminate these two customers to avoid any kind of problem (Id: `037U619y`).
Since age is a customer related features, there are only three features related to customers that could help us to fill NULL values: id, gender, age and customer type.
We filled the missing values to keep all the records rather than eliminating those customers. To fill the missing values in the dataset we used the mean age of the customer. There is a difference between premium and not premium customers average ages, so we can fill each missing value with the mean of the costumer's cluster, without modifying the average age by gender and customer type.

3. DATA EXPLORATION

In the data exploration phase, we started by considering separately the categorical and numerical variables. For what concerns this second type of variables, we divided them in 2 other subgroups: Age, price, shipping delay and arrival delay as pure numerical variables, and it is possible to use the correlation test, while instead for the ordinal variables (which essentially are encoded categorical variables) we must rely on other tests. In order to increase the readability of the graphs, we also decided to keep the Satisfaction feature in all the data frame, to make data visualization simpler.
An important thing discovered is that the average age for premium costumer is higher compared to the one of not premium and this could give some insights for marketing communication purposes *e.g.* an hypothetical advertisement campaign should be conducted on specific channels related to the age of the different customers.
Since price, arrival delays in days and shipping delays days have an exponential-shaped distribution, we apply a log transformation in order to find some more interesting data pattern. We created three new variables: "log_price", "log_shipping_delays" and "log_arrival_delays"; we saw that the only one that actually turned into a normal was "log_price".

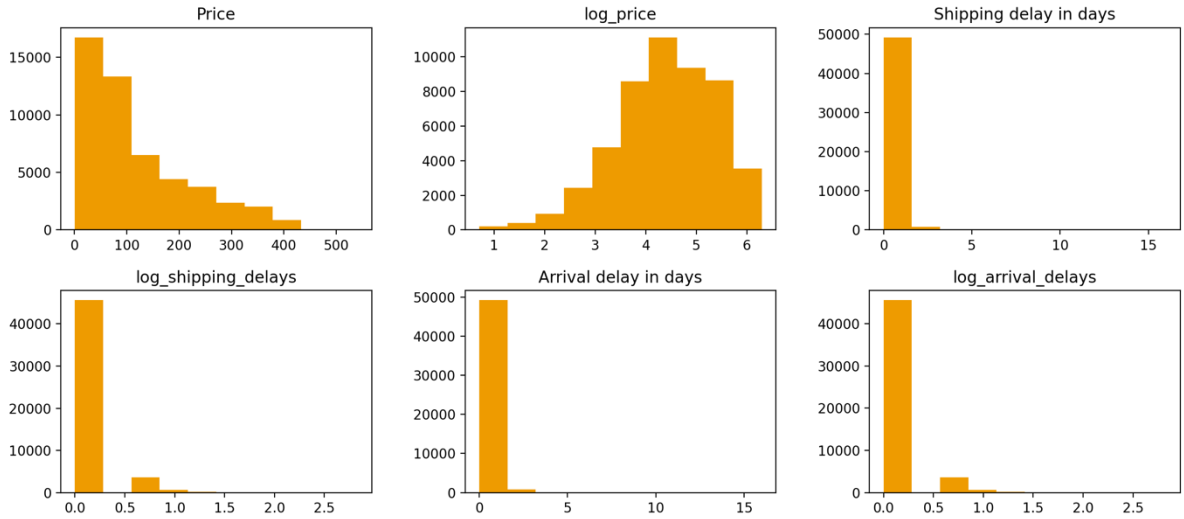


Figure 2: Log-transformation of numerical variables.

While performing a pairplot we discovered that the features "log_shipping_delays" and "log_arrival_delays" from a visual point of view have the same distribution, so we plotted the correlation matrix to confirm our feeling, discovering that those two variables have a correlation index of 0.88, thus confirming what we thought. For this reason, we decided to only keep "log_arrival_delays" and delete "log_shipping_delays" since it is basically redundant. Then we analyzed the other features, looking at their variance, with the goal of finding more possible redundant variables, but we found that there were not any features with a variance lower than 0.002 and so we decided to proceed without deleting any other feature.

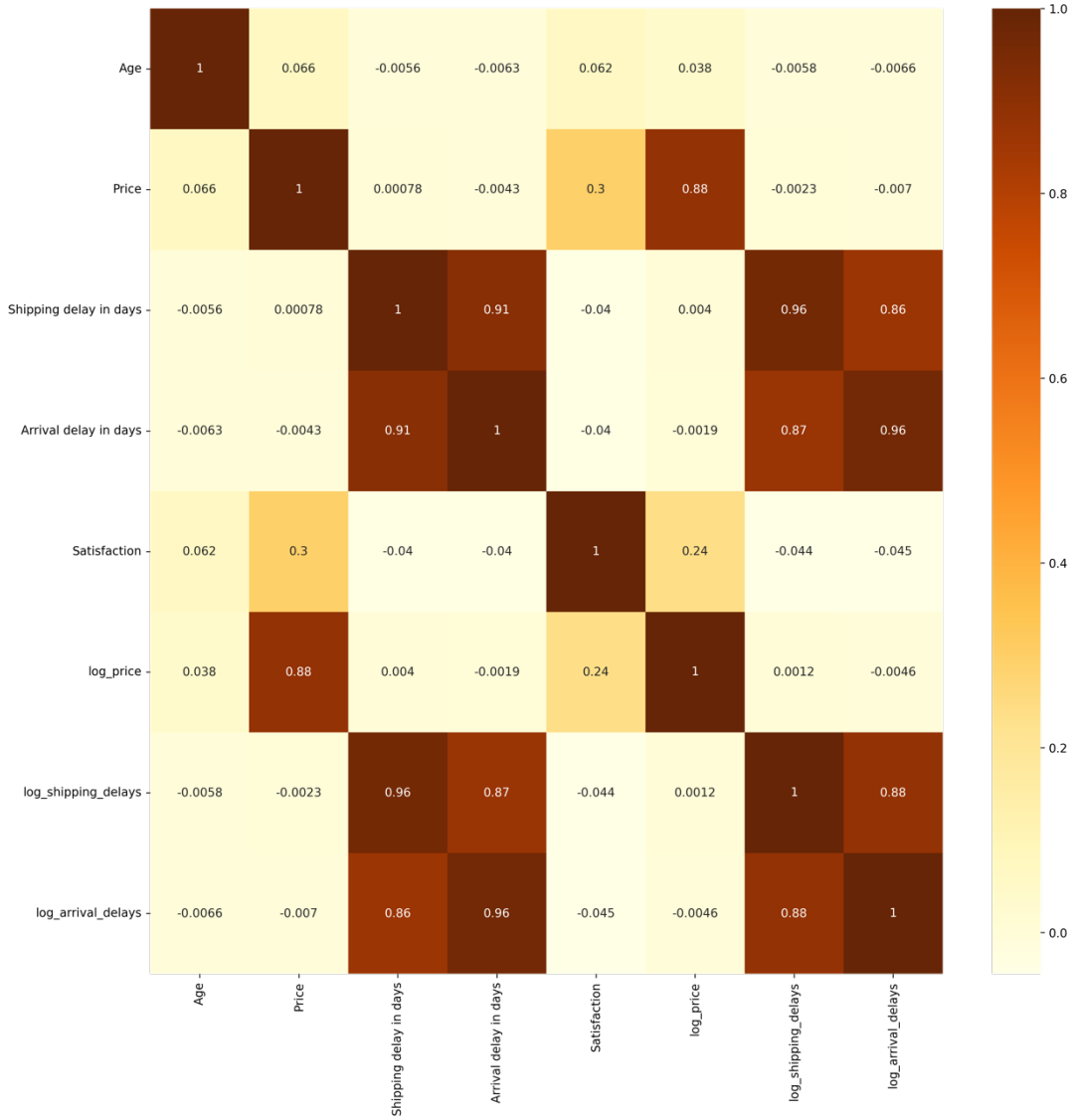


Figure 3: Correlation matrix for numerical values.

So, at this point numerical variables are age, log_price, log_arrival_delays (*cfr.* Features selection roadmap appendix).

For what concerns the categorical variables, it's meaningful to relate the target to the different variables. From an easy visual observation is possible to see that the item status new/used doesn't influence the satisfaction of the client at all, instead obviously premium costumer will be more satisfied than not premium. This shows that among the categorical variables the costumer type is the most discriminant one.
For a more precise feature selection, we decided to perform a chi-square test, in order to understand which are the most relevant features. The test is based on these hypotesis

- H_0: the feature is not relevant enough, feature and target are independent
- H_1: the feature is relevant, feature and target are not independent.

Then we defined the p-value as 0.02, in order to have a very reliable test. Finally, we analysed the results of the test on the "review" data frame and discovered that the test rejected H_0 for every feature, meaning that they are all relevant for our analysis.

Box and Whiskers plot was used to take an easy and clear look to the influence of the nine features on the satisfaction and we discovered that:

- Satisfied clients usually give a higher evaluation (obviously).
- Product description accuracy and manufacturer sustainability don't really influence the satisfaction.

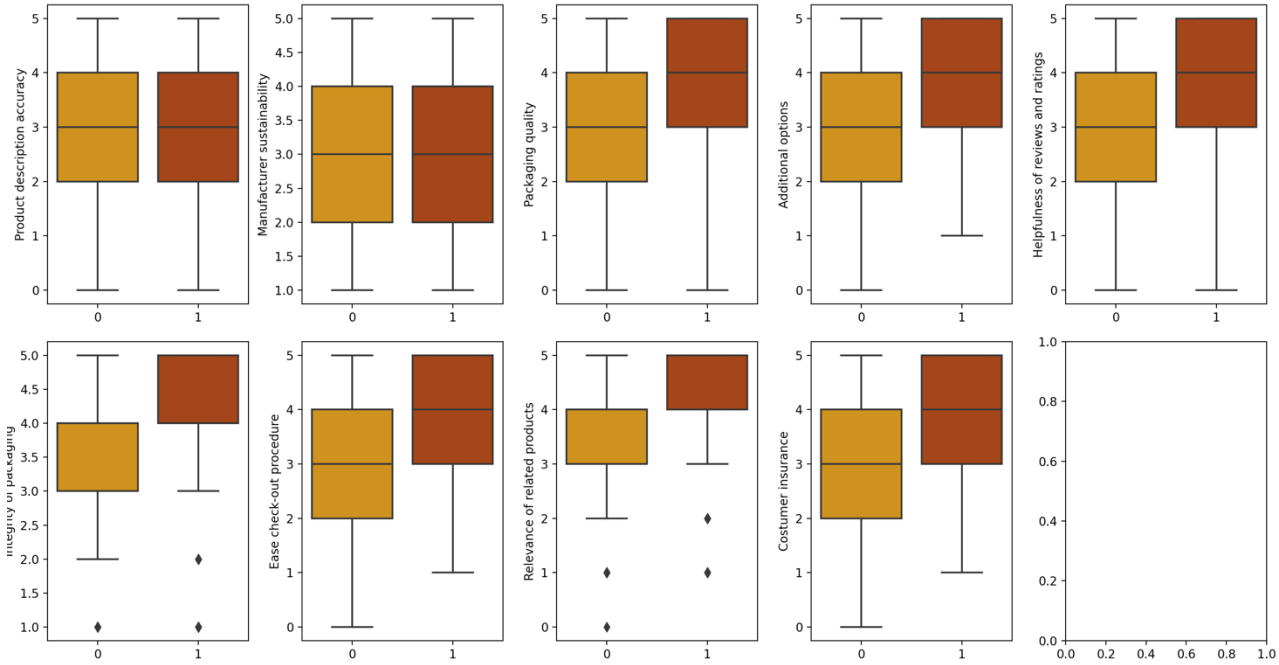


Figure 4: Reviews boxplot.

The next step has been the data standardization, except (obviously) for the target column (satisfaction).
We proceeded with the principal component analysis in order to increase the readability of our dataset, but sadly discovered that the new components were not particularly relevant and had no semantic meaning. So, we decided to use the scaled selected-features dataset to perform classification algorithms without variance losses.

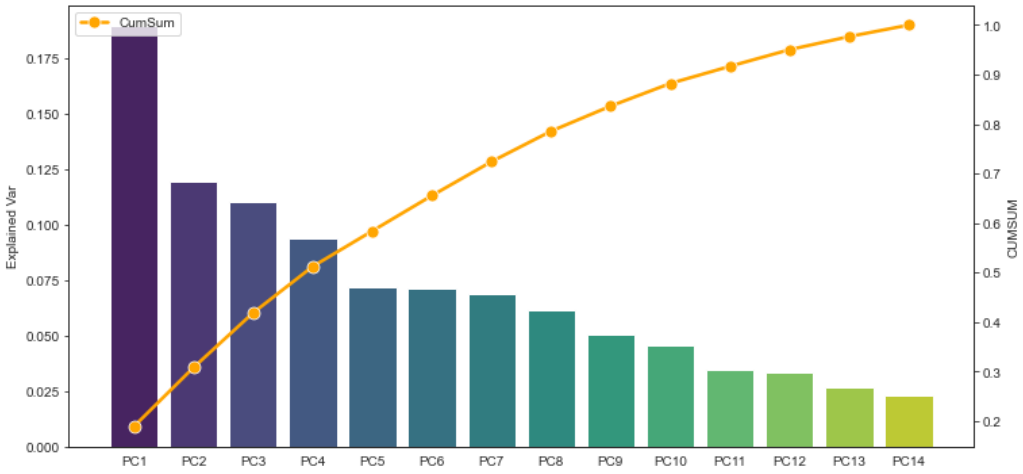


Figure 5: Cumulative explained variance.

Finally, we looked at the level of balance of the data and found that there are 8,994 more unsatisfied customers than satisfied and we decided to downsample the data by deleting the "extra" unsatisfied customers, making the dataset perfectly balanced. By doing this, there is an information loss, but on the other hand dataset is more coherent, thus increasing $f1$ score of our model.

5. MODEL SELECTION AND MODEL TUNING

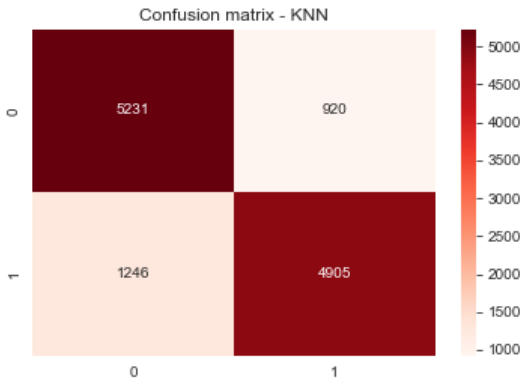
For the training phase, we firstly decided which indicators to use in order to measure the accuracy of our tests in order to have the most comprehensive yet easiest analysis possible. We defined a set of five metrics.

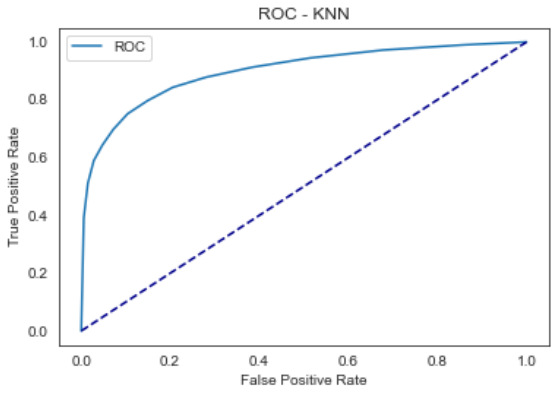
- Precision: the ratio between the number of true positives and the number of true positives + false positives;
- Recall: the ratio between true positives and true positives + false negatives:
- The F-1 score, defined as before in this report: it ranges from 0 to 1, where 0 means that the test is not reliable at all, and 1 meaning that both precision and recall are at the highest;
- The confusion matrix, to help take a clear visual look to the results of the tests;
- The ROC curve and its AUC, to see the performance of our models in the increasing classification levels.

For the model selection phase, we started by splitting training and test set (test set is 30% of the remaining observations after the previous phases), which leads us to have a 28,702 records training-set and 12,302 records test-set.

5.1 K-nearest Neighbors classifier

About the models, we started from the KNN, with the initial number of neighbours ranging from 10 to 500 with a step of 20 and we obtained that the best value is 10. Starting from this results we tried from changing the range to 8-15 with a step of 1 getting to the conclusion that the best number of neighbours is 13 obtaining a precision of 0.81 for the positives, while 0.84 for the negatives, a recall of 0.85 for the positives, and 0.80 for the negatives, an F-1 score of 0.83 for the positives and 0.82 for the negatives; then we plotted the confusion matrix in order to take a quick visual look at the results; finally we looked at the ROC curve, and obtained an AUC value of 0.90.





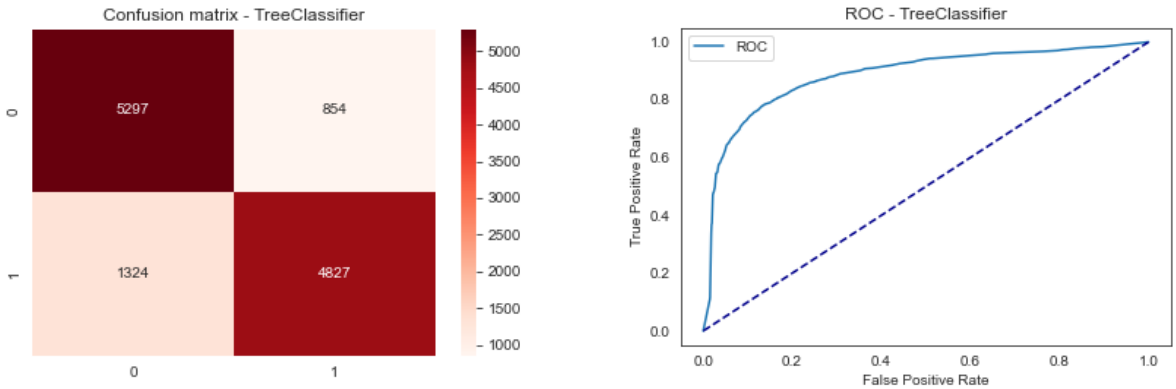
	Precision	Recall	F1-score
0	0.81	0.85	0.83
1	0.84	0.80	0.82
Accuracy			0.82
Macro avg	0.82	0.82	0.82
Weighted avg	0.82	0.82	0.82

Figure 6: KNN model results

5.2 Decision tree classifier

Then we tried the tree classification, using Gini index and Entropy index as criterions to measure the quality of the splits, max depth ranging from 5 to 12 with steps of 1, minimum number of samples to split an internal node ranging from 5 to 12 with steps of 1, and minimum number of samples to be a leaf node ranging from 5 to 12 with steps of 1.
We obtained as best criterion the Gini index, as optimal max depth 11, as minimum of samples leaf 7, and as minimum of samples split 9, in this situation the model is able to achieve a precision of 0.80 for the negatives and 0.85 for the positives,

a recall of 0.86 for the negatives and 0.79 for the positives, an F-1 score of 0.83 for the negatives and 0.82 for the positives, and an AUC value of 0.89.
Before computing all the different models, our first idea was to take into consideration the tree classification for making the prediction because it's the only model that is easily interpretable by an external user. After plotting it we obtained a very complex tree, almost impossible to interpretate and so for the prediction we decided to use the model with the highest AUC and f1-score.

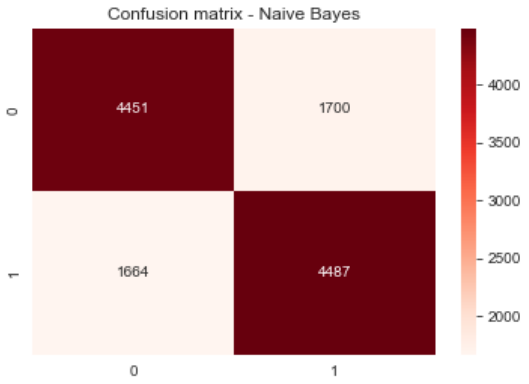


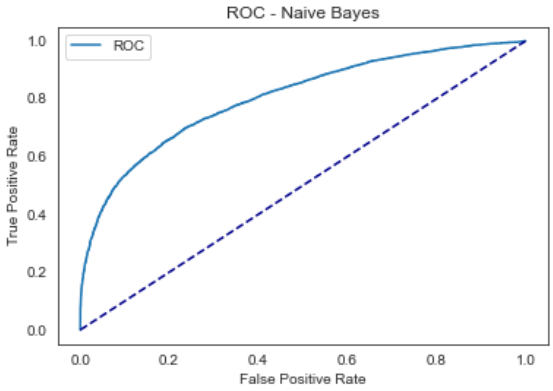
	Precision	Recall	F1-score
0	0.80	0.86	0.83
1	0.85	0.79	0.82
Accuracy			0.82
Macro avg	0.82	0.82	0.82
Weighted avg	0.82	0.82	0.82

Figure 7: Tree classifier model results

5.3 Naïve Bayes

We also performed the Gaussian Naïve bayes model obtaining a precision of 0.73 for both positives and negatives, a recall of 0.72 for positives and 0.73 for negatives, an F-1 score of 0.73 for both positives and negatives, and finally an AUC of 0.80, quite a bit lower than the other tests' results.



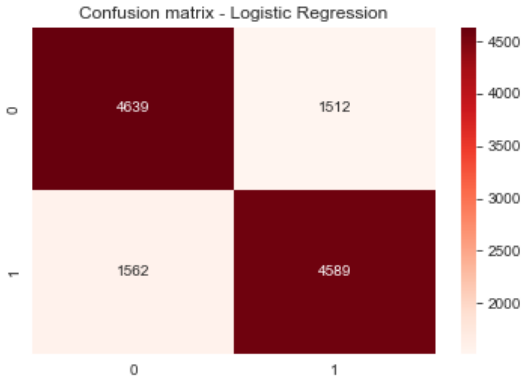


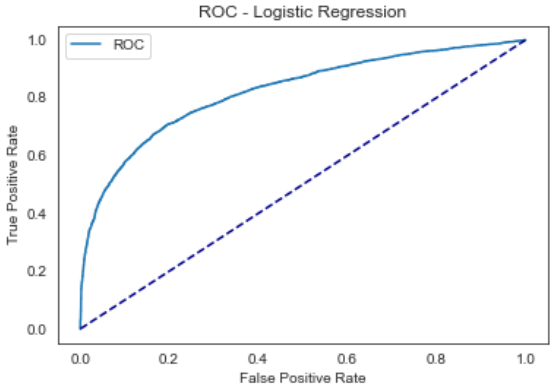
	Precision	Recall	F1-score
0	0.73	0.72	0.73
1	0.73	0.73	0.73
Accuracy			0.73
Macro avg	0.82	0.82	0.73
Weighted avg	0.82	0.82	0.73

Figure 8: Naïve Bayes classifier model results

5.4 Logistic Regression

Then, we tried the logistic regression, with the Float parameter C ranging from 1 to 10 with steps of 1, and a maximum number of iterations of 1000. Finally, using a C value of 10, we've been able to obtain 0.75 for Precision, recall and F-1 for both positives and negatives, with an AUC of 0.82.





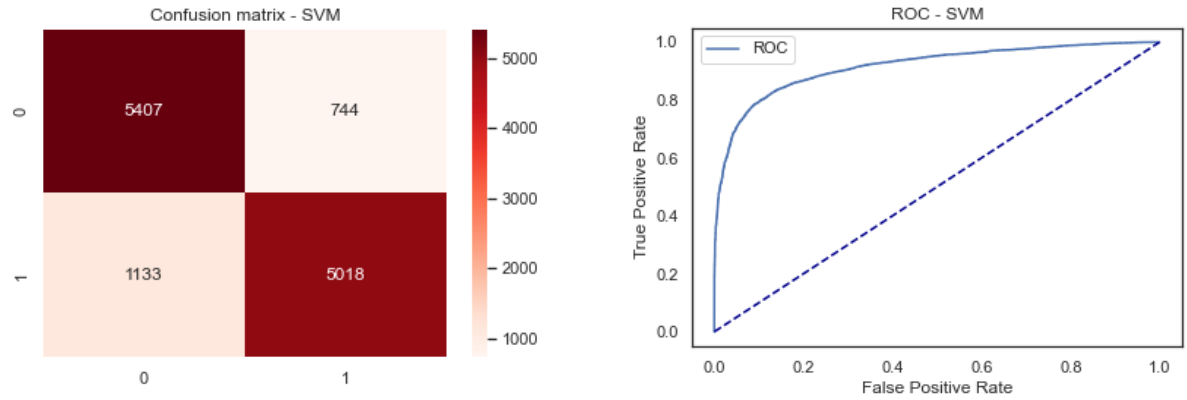
	Precision	Recall	F1-score
0	0.75	0.75	0.75
1	0.75	0.75	0.75
Accuracy			0.75
Macro avg	0.75	0.75	0.75
Weighted avg	0.75	0.75	0.75

Figure 9: Logistic regression model results

This model is less performing than the others, on the other hand it is explicative and the results are highly interpretable. We will discuss this in the last section.

5.5 Support Vector Machine

The next model has been the support vector machine. We made many tries with the purpose to find the optimal C value. This is particularly important because it strongly affects the misclassification level, in fact, having a lower C would allow us to have a large minimum margin of separation, while instead a large value would give us a hyperplane that is able to separate many more instances. Obviously, we would like to have both those characteristics, but since it's not possible, we've performed many tries in order to find the C value that most suits our dataset.
We first tried using linear and radial basis function kernels, with two possible C values: 0.1 and 100, obtaining 100 as optimal C value, and rbf as optimal kernel type, allowing us to obtain a precision of 0.83 for the negatives and 0.85 for the positives, a recall of 0.85 for the negatives and 0.82 for the positives, and an F-1 score of 0.84 for both.
Then we tried using 10 and 20 as possible C values, obtaining 10 as optimal, with a precision of 0.83 for the negatives and 0.87 for the positives, a recall of 0,88 for the negatives and 0.82 for the positives, a $f1$ score of 0.85 for the negatives and 0.84 for the positives.
Finally, we tried with 5 possible C values: 5, 6, 7, 8, 9 and, with C = 6, we obtained a precision of 0.83 for the negatives and 0.87 for the positives, a recall of 0.88 for the negatives and 0.82 for the positives, a $f1$ score of 0.85 for the negatives and 0.84 for the positives, and a very good AUC of 0,92.



	Precision	Recall	F1-score
0	0.83	0.88	0.85
1	0.87	0.82	0.84
Accuracy			0.85
Macro avg	0.85	0.85	0.85
Weighted avg	0.85	0.85	0.85

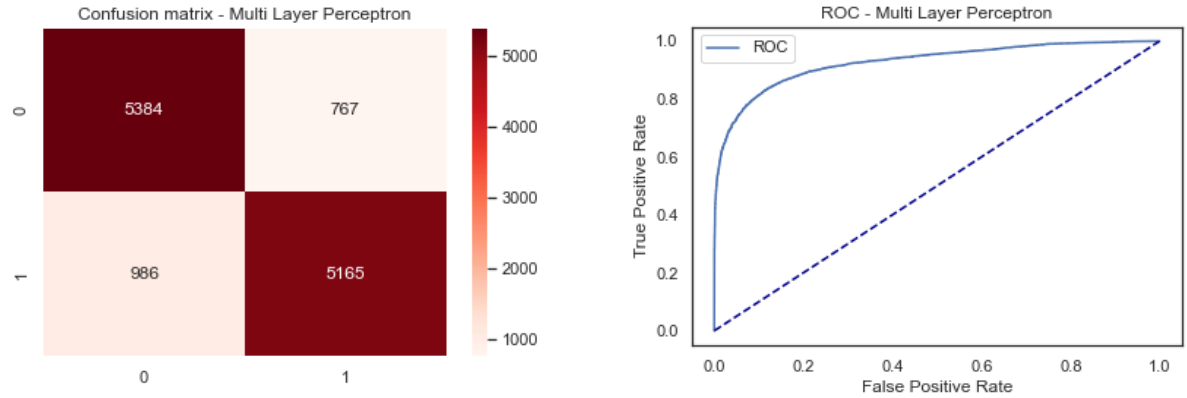
Figure 10: Support vector machine model results

5.6 Multi-layer Perceptron Model

Finally, we tried the multi-layer perceptron model, first, similarly to what we've done for the support vector machine, we have done many tries, in order to find the optimal values of hidden layer sizes and alpha. Finding the correct number and size of hidden layers is important because if we use a very little number, we risk of having a very vague approximation, so we started by trying with two layers of respectively 10 and 5 size, and then three layers of 100, 200 and 5 size. On the other hand, it is very important to find the correct alpha value, because, since it affects the regularization, having a high alpha would help us decrease the overfitting, creating a more linear decision boundary, while a lower value would reduce underfitting, but could create a very complicated decision boundary. We started with two values: 0,001 and 0,1.

The first hyperparameter search showed that the best solution is the one with alpha 0,1, and three layers, which obtained a precision of 0.83 for the negatives and 0.88 for the positives, a recall of 0,89 for the negatives and 0,82 for the positives, and an F-1 score of 0.86 for the negatives and 0.85 for the positives.

Those results led us to try higher alpha values (0.1 and 0.7) and different hidden layer sizes: (100, 20, 5), (3, 2, 3), (100), (3), (150, 35, 7). The hyperparameter search showed that the combination of an alpha of 0.7 and a hidden layer sizes of (150, 35, 7) obtaining a precision of 0.82 for the negatives and 0.90 for the positives, a recall of 0.91 for the negatives and 0.80 for the positives, and an $f1$ score of 0.86 for the negatives and 0.85 for the positives and an AUC of 0.93.



	Precision	Recall	F1-score
0	0.85	0.88	0.86
1	0.87	0.84	0.85
Accuracy			0.86
Macro avg	0.86	0.86	0.86
Weighted avg	0.86	0.86	0.86

Figure 11: Multi-layer perceptron model results

6. FINAL CONSIDERATION

Even if we discarded logistic regression model because of its low performances, we could draw some interesting conclusions for the analysis of satisfaction drivers.

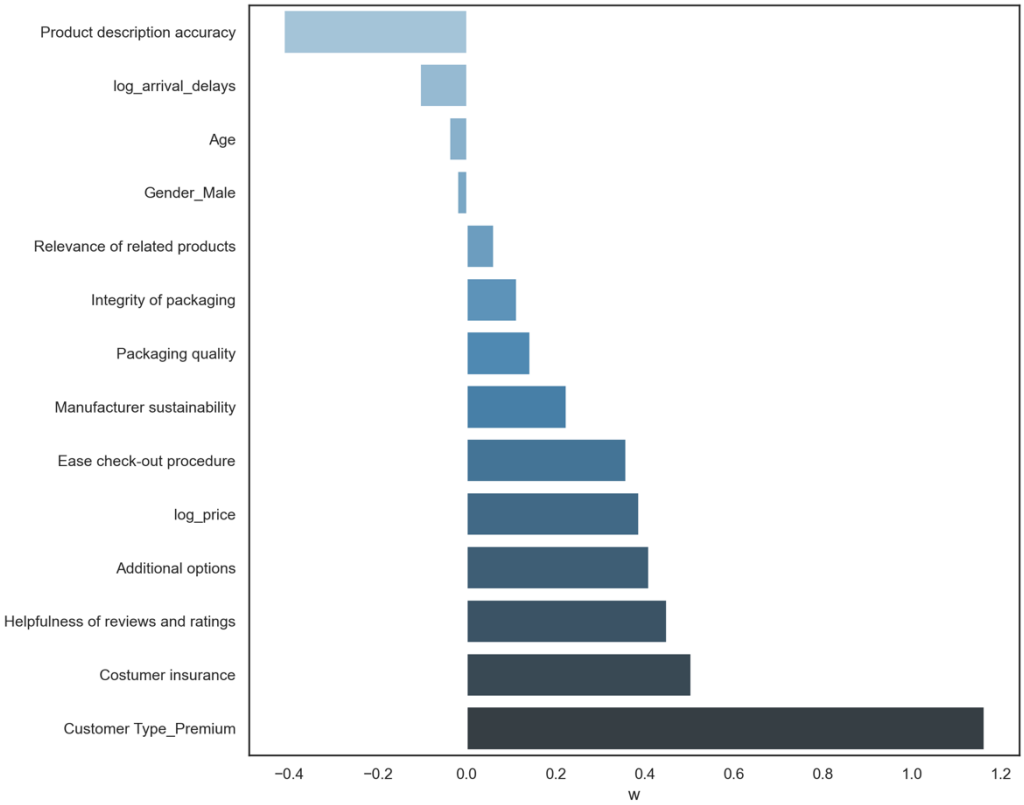
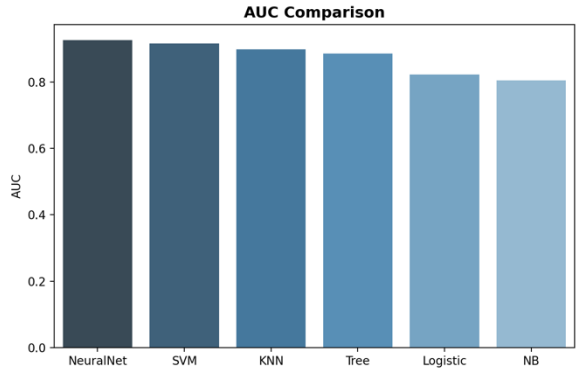


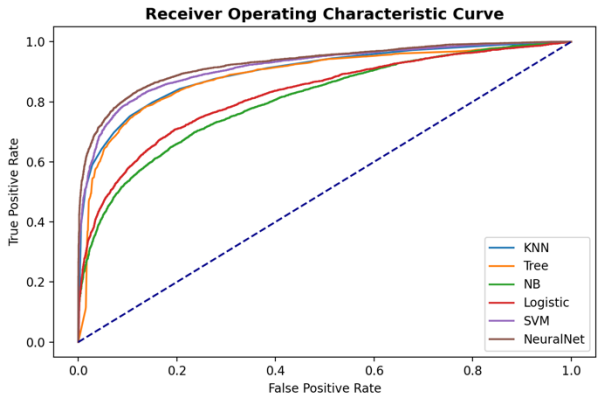
Figure 12: Coefficients of logistic regression

By plotting the coefficient of each feature obtained in the logistic regression, we discovered two interesting insights:

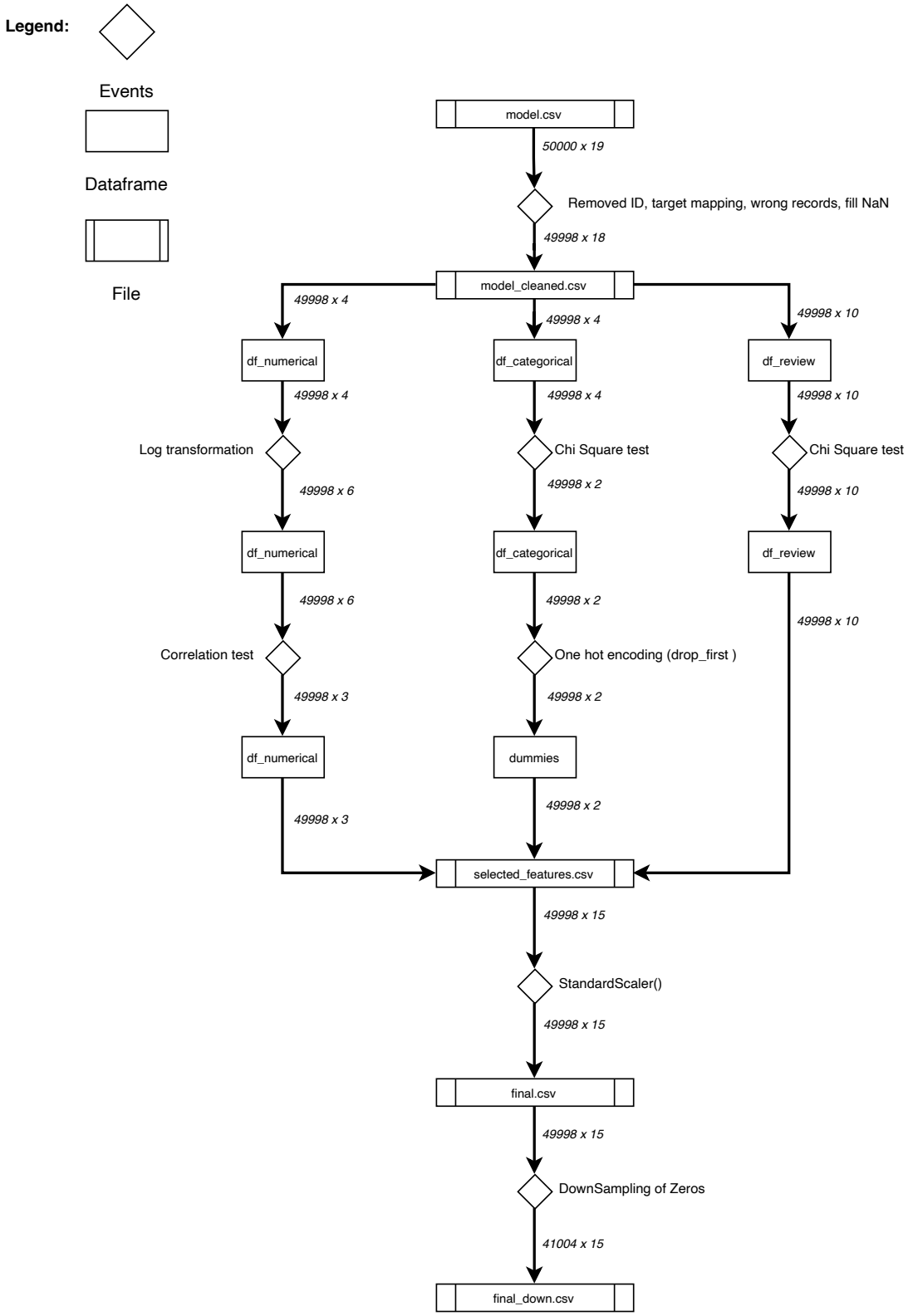
- The customer type very strongly affects the satisfaction, meaning that this service that they're offering is very effective and is a strength point for the company, which should leverage on it.
- The product description accuracy, is strangely negatively correlated to the satisfaction, meaning that the lower the product description accuracy is, the higher the satisfaction level will be, this is definitely metric that the company should revise, because this strange correlation could be the result of a non-clear explanation of the meaning of this item in the customer satisfaction survey.

In conclusion, in the images below we can see the ROC of each of the models presented, it is possible to clearly understand that the MLP (NeuralNet) and the SVM have the best performances and could be the most suitable models.





Appendix: Features Selection Roadmap



Note: in notebooks there are duplicate of target variable for data viz purposes. In this schema are reported NO duplicated.