FIRST LOOK, AND MODEL SELECTION

Having a first look at the train dataset, we can see that we have a clearly imbalanced data:

* 0: 38673
* 1: 12209

Imbalanced data affects the performance of many usual models used for classification, like Logistic Regression, Support Vector Machines… so we are going to be limited in the possible models that we can try. There are other ways to deal with imbalanced data, like collecting more data, changing the performance metric, resampling the data…but obviously these are not possible solutions for this task.

Some candidate models are: XGBoost, Random Forest and Light Gradient Boosting (Light GBM).

DATA PRE-PROCESSING

The first step is to check the presence of missing values. After doing that, we can see missing values in 3 different variables:

* Health Indicator
* Holding\_Policy\_Duration
* Holding\_Policy\_Type

But we have to point out that the presence of those missing values have different causes. In Health Indicator, the presence of missing values is completely random (many people is not comfortable giving information about their health problems), but in the other two variables is not random at all. Holding Policy variables only affect people who already have a policy with the company, so people who don´t, couldn´t answer those questions. The randomness in the missing values is very important, because it is going to determine the way in which we can deal with them.

There are many ways to deal with missing values in categorical variables. The method used in this case, is to impute this missing values with the most common value in the variable. Also a new variable was created: Holding\_policy, which takes a 1 if Yes and a 0 if No. That brings the possibility of including a variable with information about the actual policies of the customers that does not contain missing values.

After performing a heatmap of the correlation matrix of the variables we can see an obvious high correlation between two variables (Upper Age and Lower Age). To avoid possible problems due to multicolinearity, one of them should be dropped from the dataset. I decide to drop upper age, because it was also correlated with Reco\_Policy\_Premium.

LABEL ENCODING

The train dataset contains categorical and numerical variables, so the first ones have to be encoded in order to be included in the future model.

Label Encoder was used for variables in which order between levels was important: Health indicator, Holding\_Policy\_Duration and Holding\_Policy\_Type. That means, that the model was going to assume a hierarchical connection between levels.

That hierarchical connection Is not always what we want, and to avoid that I used One Hot Encoding (get dummies) for the rest of the variables.

MODEL

First of all, we have to remove the variables that are not going to be included in our model. The variables: ‘ID’ and ‘Region\_Code’ were removed from the model. The Id is obviously a feature that have to be removed and Region Code has 5638 levels, so it was also impossible to encode.

After that the train dataset was divided into train and validation test, in order to be able to have an estimation of the performance of our model. I decided to make typical split of 80-20 of the train set.

As the dataset was imbalanced, I calculated the weights of the target variable, in order to use them in the model.

After trying some models, Light GBM was the one with the best performance.

To get the best parameters for the model, a Randomized Search of them was performed.