Predicting if a costumer will start a dispute after a complain regarding financial services.

Data extracted from: Consumer Financial Protection Financial Bureau.

https://github.com/Thaleia18/Disputes_and_complains_regarding_financial_services

"Every complaint provides insight into problems that people are experiencing, helping us identify inappropriate practices…"
https://www.consumerfinance.gov/data-research/consumer-complaints/

After every complain consumers can start a dispute.

In this repository:

- Consumer data notebook
- Optimizing with grid search notebook
- is this overffiting notebook
- Insights notebook

Consumer data notebook

Exploratory analysis.

From the exploratory analysis I found a lot of findings:
A lot of null values from disputes:

- Products with no data about disputes
- Companies with no data about siputes
- Cases still in progress.

I didnt find any patter that indicated that a feature was a key to predict disputes.

I decided dont use date and locations features, since seems like all the dates and locations have the same rate of dispute vs non disputes. Also all these were categorical variables that just add more complexity to the models.
Most of my data is about non-disputes.

Featuring ingeneering

I used just the data with non null values for dispute. Create a model to predict if the customer will have the option to start a dispute is another interesteing issue. Now I will focus in: having the option to start a dispute, will a customer start a dispute??

I processd all the categorical variables, in some of them they had thousands of unique values so I grouped them in categories. Number of featur

Unbalanced class and setting simple models.

The classes for dispute are unbalanced, most of the data is for non-disputes. This is an issue for the models because I will reach high accuracy if the models are good predicting non-dispute, but from a business perspective is more interesting find when a dispute will start.

There are different approachs to unbalanced classes:
Resampling -- Up-sampling non-dsiputes -- Down sampling disputes But I dont want to create synthetic data.
Change performance metric from accuracy to AUROC.
Peanlizing algorithms using class weight.
Using tree based algorithms

I will use a combination of the last two options.
es for basic model= 90

Optimizing with grid search notebook

I removed some of the group by categories to add more features to the models, now I have 283 features. I used grid search to optimize the parameter values for each model.

is this overffiting notebook

MY main problem was that even when I have high accuracy F1 is smaller. The model is good to predict no disputes, but is failing to predict disputes.
I want to analize if there is overfitting. I used learning_curve to plot the accuracy curves for different sizes of training samples. All the models got a gap between the training score and validation score, with training score being higher, this means models are overfitting. The overfit is reduced when 20-25% of the data is used to training and the other is set to the validation set.

Insights notebook

I used eli5 to perform Permutacion Importance.
Permutacion importance returns the weight of each feature (weight is how the accuracy changes when the values in the feature shuffle).
When negative weights are got, that means that the accuracy improved with the shuffle values meaning that the feature is not important.
I want to check with features got positive weight for each models.
So I can eliminate the other features and explore features that I didnt include in the previous models.

Next steps (Currently working on that...)

Kept features with positive weight and create new models using features such sub_products, sub_issues and customer_narrative.
Change the metrics on the models to AUROC.